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

D4.1 — State of the Art and initial technical requirements

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.

Deliverable Information

Workpackage	WP4
Workpakace Leader	(CTL)
Deliverable No.	D4.1
Deliverable Title	State of the Art and initial technical requirements
Lead Beneficiary	ISI
Type of Deliverable	Report
Dissemination Level	Public
Due Date	30/06/2023

Document Information

Delivery Date	31/07/2023
No. pages	186
Version | Status	1.1 | final
Deliverable Leader	All consortium partners
Internal Reviewer #1	Christos Strydis (EMC)
Internal Reviewer #2	Joppe W. Bos, Gareth T. Davies (NXP)

Quality Control

Approved by Internal Reviewer #1	28/07/2023
Approved by Internal Reviewer #2	28/07/2023
Approved by Workpackage Leader	29/07/2023
Approved by Quality Manager	31/07/2023
Approved by Project Coordinator	31/04/2023

List of Authors

Name(s)	Partner
Konstantina Karagianni, Vassilis Paliouras, Christos Tselios, Alexander El-Kady, Apostolos Fournaris	ISI
Francesco Regazzoni, Kostas Papagiannopoulos, Marco Brohet	UvA
Alberto Gutierrez-Torre, Ferral Agulló, Laia Tarrés, Josep Ll. Berral	BSC
Juan Carlos Perez Baun	ATOS
Paolo Palmieri, Buvana Ganesh	UCC
Joppe W. Bos, Gareth T. Davies	NXP
Gergely Acs, Balazs Pejo	BME
Peter Pollner	SEM
Alice Héliou, Vincent Thouvenot	TSG

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
22.02.2023	0.1	Apostolos Fournaris (ISI)	Created the document and the initial version of its content
30.03.2023	0.2	Apostolos Fournaris (ISI)	Updated document structure
15.05.2023	0.3	All Technical partners	Draft SoTA input provided
20.05.2023	0.4	Christos Tselios (ISI) & Apostolos Fournaris (ISI)	Section 7 initial input (section 7.1) provided
30.05.2023	0.5	All Technical partners	Final SoTA input provided
15.06.2023	0.6	Christos Tselios (ISI) & Apostolos Fournaris (ISI)	Generic User requirements, SECURED architecture components and descriptions provided
20.06.2023	0.7	Christos Strydis (EMC) & Joppe W. Bos (NXP)	1st Round of review (on Sections 1-6) provided
25.06.2023	0.8	Apostolos Fournaris (ISI) & Use-Case partners	adaptation of SECURED architecture to the use-cases input provided
06.07.2023	0.9	Apostolos Fournaris (ISI), Vassilis Paliuras (ISI), Konstantina Karagianni (ISI), Alexander El-Kady (ISI), Christos Tselios (ISI)	1st Review round comments addressed
24.07.2023	1.0	Apostolos Fournaris (ISI)	Final version of the deliverable send for internal review
28.07.2023	1.0	Christos Strydis (EMC) & Joppe W. Bos (NXP)	Final Round of review (on Sections 1-6) provided

Date	Ver.	Author(s)	Summary of main changes
29.07.2023	1.1	Apostolos Fournaris (ISI), Vassilis Paliuras (ISI), Konstantina Karagianni (ISI), Alexander El-Kady (ISI), Christos Tselios (ISI)	Final Review round comments addressed and deliverable Finalized
31.07.2023	1.1	Francesco Regazzoni (UvA)	Final Deliverable submitted to EU

Table of Contents

Acronyms

2PC Secure Two-Party Computation. 50, 51, 54

3PC Secure Three-Party Computation. 50

A-SS Additive Secret Sharing. 45, 50, 51, 53

AE Autoencoder. 72

AES Advanced Encryption Standard. 44, 58, 59

AI Artificial Intelligence. 91, 118

AIA Attribute inference attack. 96

ALU Arithmetic Logic Unit. 57

API Application Programming Interface. 98–100

ASIC Application-Specific Integrated Circuit. 54–57

AUC Area Under the Curve. 83

BFV Brakerski-Fan-Vercauteren Homomorphic Encryption Scheme. 47, 48, 50–54, 56, 57

BGV Brakerski-Gentry-Vaikuntanathan Homomorphic Encryption Scheme. 47, 48, 57, 58, 63

CI/CD continuous integration/continuous delivery. 111, 113

CJ Customer Journey. 103, 104

CKKS Cheon-Kim-Kim-Song Homomorphic Encryption Scheme. 47, 48, 52, 54, 57, 60, 63

CMNT Coron-Mandal-Naccache-Tibouchi Homomorphic Encryption Scheme. 47, 54

CNN Convolutional Neural Network. 19, 83, 87

COMPAS Correctional Offender Management Profiling for Alternative Sanctions. 35

CRBM Conditional Restricted Boltzmann Machines. 80

CRT Chinese Remainder Theorem. 49, 55, 58

CT Computed Tomography. 66

CTG Cardiotocogram. 65, 121

CX Customer Experience. 104–106

DGHV van Dijk-Gentry-Halevi-Vaikntanathan, Homomorphic Encryption Scheme. 47, 58

DL Deliverable Leader. 15, 16, 38

DL Deep Learning. 7, 14, 15, 33, 34, 36, 39–44, 48, 51, 52, 54, 60–62, 96, 111, 112, 119, 120, 124, 141, 144

DM Diffusion model. 71, 74

DMA Direct Memory Access. 58

DoA Description of Actions. 14, 15, 107, 111, 125

DP ϵ-Differential Privacy. 91, 92, 101

DTW Dynamic Time Warping. 78, 84

EC Equivalence Class. 90

ECG Electrocardiogram. 65, 67, 69, 70, 78, 119, 121

ECoG Electrocorticography. 68, 69

EEG Electroencephalography. 68–70, 76, 79

EHR Electronic Health Record. 17, 19, 26, 27, 99, 121, 123, 139, 140, 142, 143

EMG Electromyography. 68–70

EOG Electrooculography. 67, 69, 70

FECG Fetal Electrocardiogram. 67, 69, 70

FFT Fast Fourier Transform. 54, 56

FHE Fully Homomorphic Encryption. 46–60, 63, 124

FHEW FHEW Homomorphic Encryption Scheme. 47, 57

FI Fréchet Inception Score. 84

FL Federated Learning. 14–16, 32, 39, 40, 54, 61, 101, 111, 112, 120, 144

FLAIR Fluid attenuated inversion recovery. 65

FPGA Field Programmable Gate Array. 54–56, 109

fUS functional Ultra-Sound. 117, 118

GA Grant Agreement. 14, 15

GAN Generative Adversarial Network. 71–76, 79–81, 84, 87, 88

GC Garbled Circuit. 44, 45, 50–52

GDPR General Data Protection Regulation. 86, 89, 109, 112

GPU Graphical Processing Unit. 52, 54–56, 109

GSW Gentry-Sahai-Waters Homomorphic Encryption Scheme. 47, 48

GUI Graphical User Interface. 98, 99, 101

HbC Honest-but-Curious. 42, 46

HE Homomorphic Encryption. 7, 14–16, 26, 41, 43, 45–55, 58–63, 111, 112, 118, 119, 123, 124, 126, 141, 142, 144

HMM Hidden Markov Model. 100

HPC High Performance Computing. 117, 118

IoT Internet of Things. 98

IS Abstract and Inception Score. 84

KA Karatsuba Algorithm. 56

KLD Kullback–Leibler Divergence. 84

KLIEP Kullback–Leibler Importance Estimation Procedure. 37

LAN Local Area Network. 62

LFHE Leveled Fully Homomorphic Encryption. 46, 47

LPIPS Learned Perceptual Image Patch Similarity. 83

LWE Learning-with-Errors Homomorphic Encryption Scheme. 48, 54

MAC Message Authentication Code. 45, 52

MAE Mean Absolute Error. 83, 84

MEG Magnetoencephalography. 68, 69

MIA Membership Inference Attack. 96

ML Machine Learning. 7, 14, 15, 33–41, 46, 49–52, 54, 55, 58, 61, 62, 91, 95, 96, 111, 112, 119, 120, 124, 141, 144

MMD Maximum Mean Discrepancy. 84

MRI Magnetic Resonance Imaging. 65, 66, 73, 75, 76, 87, 117, 118, 121

MSE Mean Squared error. 83, 84

NN Neural Network. 41–44, 49–55

NTT Number Theoretic Transform. 51, 55, 56, 58

OT Oblivious Transfer. 43–45, 48, 51–53

OTe Oblivious Transfer Extension. 43, 45, 53

P-SPN Partial Substitution-Permutation Network. 59, 63

PAHE Packed Additively Homomorphic Encryption. 51, 52

PCC Pearson Correlation Coefficient. 84

PCI Peripheral Component Interconnection. 58

PE Processing Element. 58

PETs Privacy-Enhancing Technologies. 7, 41, 44, 51, 54, 61–63, 144

PHE Partially Homomorphic Encryption. 46

PPT Privacy-preserving technique. 89, 91–93, 98

PRD Percent Root Mean Square Difference. 84

PSNR Peak Signal-to-Noise Ratio. 83

RBM Restricted Boltzmann Machines. 80

RMSE Root Mean Squared Error. 84

RNN Recurrent Neural Networks. 19, 20

RNS Residue Number System. 50, 55, 56, 58

ROT Random Oblivious Transfer. 43, 45

RPU Ring Processing Unit. 57

SDLI Secure Deep Learning Inference. 41–46, 48, 49, 52–54

SDLT Secure Deep Learning Training. 41–43, 46, 54

SIMD Single-Instruction-Multiple-Data. 47–49, 51–53, 58

SMPC Secure Multi-Party Computation. 7, 9, 14–16, 26, 28, 41–45, 48–50, 52–54, 59–63, 107, 109, 111, 112, 118–120, 123, 124, 126, 141, 142, 144

SNP Single Nucleotide Polymorphism. 65, 70, 82, 85

SNV Single Nucleotide Variant. 80

SoTA State-of-the-Art. 7, 14–16, 39, 125, 144

SPN Substitution-Permutation Network. 59, 63

SSI Structural Similarity Index. 84

SSIM Structural Similarity. 83

SWHE Somewhat Homomorphic Encryption. 46

TFHE FHE-over-the-Torus Homomorphic Encryption Scheme. 47, 48, 50, 52

UJ User Journey. 103–107

UJM User Journey Mapping. 106, 107

UQI Universal Quality Index. 83

UX User Experience. 103, 106, 107

VAE Variational Autoencoder. 71–74, 79–81

VIF Visual Information fidelity. 83

WAN Wide Area Network. 62

WSI Whole Slide Image. 121

ZK Zero Knowledge. 59, 60, 62, 63

1 Executive Summary

This deliverable is meant to document the preliminary activities of SECURED project Task 4.1 that aims from the collection of State-of-the-Art, user requirements, technical requirements to provide a detailed and analytic description and analysis of the SECURED Architecture, its components, their interactions and the data that they are exchanging. The D4.1 provides reporting of the first 6 months of activities on T4.1 that include:

- Advanced Anonymization techniques and tools including the use of Federated Learning (FL) as Privacy Enabling Technologies;
- State-of-the-Art (SoTA) documentation and analysis of Secure Multi-Party Computation (SMPC) and Homomorphic Encryption (HE) schemes and their adaptation to Machine Learning (ML) and Deep Learning (DL);
- Synthetic Data Generation at large scale;
- Biasing and Unbiasing techniques and methods.

The results of this study are been evaluated in order to identify the most prominent schemes to adopt as a starting point for the research and development to be done in WP2, WP4. In addition to the SoTA we also document preliminary user requirements and descriptions of the use-cases in accordance to the SECURED solution and derive a preliminary SECURED Architecture. All the above eventually are processed in order to extract and document the SECURED technical requirements in the deliverable, enabling the design and implementation of the SECURED Infrastructure and Innohub solution as sketched in the project's Description of Actions (DoA). This Deliverable constitutes an intermediate report on the T4.1 activities and will eventually be updated in the final D4.2 deliverable of the task.

1.1 Related Documents

- Grant Agreement (GA) Project 101095717 - SECURED; Description of Actions (DoA) Annex 1

2 Introduction

This report comprises the project's Deliverable D4.1 "State of the Art and initial technical requirements" that is associated with the T4.1 "State-of-the-Art, Technical Specifications & Architecture Design". As described in the Grant Agreement (GA), this task initially aims to collect the state-of-the-art research that has been done in the various research areas explored through the SECURED project focusing on the two flows of the project, i.e. the data flow and the processing flow as described in the SECURED Description of Actions (DoA). The SECURED project is exploring research on how to perform anomymization in a way that will make deanonymization attempts hard to achieve and in parallel explores privacy-preserving data processing through privacy-enhancing technologies, chiefly Secure Multi-Party Computation (SMPC) and Homomorphic Encryption (HE). The SECURED project will also explore how to perform synthetic health data generation in an efficient yet privacy-preserving manner. It should be noted that the project's use case is the health domain so all data involved in the above activities are health data. In addition, given that the one of the popular type of data processing needed in today's health applications is related to Machine Learning (ML) or Deep Learning (DL) and also that privacy preservation across many entities (in view of ML/ DL) is done using Federated Learning (FL) schemes, in SECURED the performed research, as described in the project's DoA, is foused on ML/ DL) applications. With the above guidelines in mind, in this deliverable we report State-of-the-Art (SoTA) work of related ML/ DL/FL approaches that are linked to the solutions of the following areas:

- Stat-of-the-art schemes relative to Secure Multi-Party Computation (SMPC)/Homomorphic Encryption (HE), existing libraries and solutions including techniques of scaling up to many parties and complex computations;
- Advanced anonymization schemes for health data that can offer considerable anonymization resistance against privacy attacks (including de-anonymization attacks);
- Synthetic data generation solutions for health data.

Note that the above research areas are chosen after analyzing the project's DoA objectives and the research topics that we address in SECURED as well as the conceptual architecture of Figure 2 in the Project's GA Annex 1 (i.e the project's DoA).

The most promising solutions are identified and the existing state-of-the-art gaps are documented in order to fuel the research and development to be done in WP2, WP3 and WP4.

Apart from the SoTA work reported in the deliverable, a thorough description and analysis of the preliminary SECURED architecture Technical requirements is also provided, as those are extracted from the work done in T4.1 and also through discussions with the technical and use-case partners (as part of the T5.1 activities). More specifically, a preliminary architecture of the SECURED solutions (design around the SECURED Innohub concept) is provided, the main architectural components are identified and described while for each one of them technical functional and non-functional requirements are documented.

2.1 Structure of the Document

The document is divided as follows:

- Section 3 provides brief state-of-the-art on ML/DL techniques that can be applied on health data and then a similar analysis on FL techniques. The focus of the analysis is on ML/ DL/FL for medical images, electronic health records, health monitoring and genomics as those are related to the SECURED four use-cases and on possible inclusion of external (additional use-cases) partners through the SECURED open call activities. Special focus is given on the security and privacy of FL solutions, thus in this section a threat model is provided along with the state-of-the-art security and privacy concerns in FL. Additionally, in this section the most popular FL tools and libraries are briefly presented. Finally, in the section the concept of biased data in a regular ML/DL and FL setup are discussed and approaches on how to mitigate bias are described.

- Section 4 is focused on SMPC and HE solutions for health applications. Initially, a brief description of the existing SMPC approaches is provided including Garbled Circuits and Oblivious Transfer solutions, Secret Sharing schemes and Homomorphic Encryption schemes. Then the focus shifts towards the latest state-of-the-art HE solutions where the latest existing libraries are presented and how such libraries are further used to offer privacy preserving DL software libraries that support DL applications. This analysis includes HE-only software DL libraries, SMPC software DL libraries as well as hybrid DL libraries. Furthermore, in this section, the research that is performed on how to scale up the existing SMPC/HE schemes is reported. Such research can be based on hardware acceleration through GPUs, processor instruction set optimizations or dedicated custom hardware setups, as well as software acceleration through algorithmic means. In addition to the above, we also document attempts to scale up SMPC/HE schemes under FL settings. The section concludes with a critical view on the existing trends on the topic and insights on how to use them to enhance health privacy.
- Section 5 provides a thorough documentation of the state-of-the-art research on synthetic data generation for specific types of Health Data (determined from the SECURED various use-cases). Initially, the data types and data formats that are relevant to the project, to be synthetically generated are analyzed in detail. Then, the various synthetic data generation techniques for the identified data are reported and described in detail. The existing libraries and tools to be used for the above functionality are also reported and finally, an evaluation of the tools and techniques is made. Also, the next steps in the relevant research are reported based on the latest research trends on the topic.
- Section 6 provides a state-of-the-art on health data anonymization documentation. This includes the latest advanced anonymization techniques as well as the existing tailored techniques in the relevant literature that are used in order to de-anonymize the anonymized data. After providing the above information, the existing tools of anonymization and de-anomymization are presented and a comparison of those techniques/tools is made.
- Section 7 documents the procedure and the results of extracting the SECURED Architecture Technical requirements. Initial, the methodology that was followed for the above process is described i.e the user journey approach and the process mapping and then the adaptation of the SECURED Architecture is provided, leading to the preliminary SECURED architecture and an short description of each component of this architecture. After the preliminary architectural components are identified and their core functionality specified, the user journey/process mapping methodology is applied to each use case partner (i.e each one of the four use-cases) using a two stage procedure: a) bilateral teleconferences between technical and use-case partners to derive draft user requirements and identify draft processes to be mapped as well and b) the 1st end user workshop where the full process map methodology has been applied individually to each use-case using the already collected inputs from stage a. The outcomes of these endeavours are provided in subsections 7.3.1 and 7.3.2 where generic (applicable to all use-cases) preliminary user requirements are provided and preliminary processes are described for each use-case including their association with the preliminary SECURED architecture and its components. Eventually, for each use-case all the preliminary SECURED architecture components that the use-cases will utilize is documented. Having collected all the above information and combining them logically with the SoTA Gaps extracted from Sections 3 to 6 as well as with the common practises on existing integration technologies of the IT market, an analytic presentation of technical requirements (functional and non functional) for each preliminary SECURED architecture component is provided. Note, that since D4.1 is an intermediate deliverable for T4.1 (the final deliverable of the task, i.e. D4.2, is due on M18) it is expected that some information of Section 7 will be updated as the activities in T4.1 progress towards M18.
- Section 8 provides a conclusion of the deliverable

3 Machine Learning in Health Applications

Machine learning, with its ability to learn from data, is expected to revolutionize healthcare by providing solutions for diagnosis, treatment, and support [1]. In particular, machine learning can be applied to automate clinical tasks, provide clinical support, and expand clinical capacities. Clinical task automation involves automating tasks performed by clinicians, such as medical image evaluation and routine processes. Clinical support aims to optimize clinical decision-making and practice by integrating different healthcare records and improving communication and coordination. Expanding clinical capacities involves screening, diagnosis, and treatment innovations. The main application opportunities of machine learning in healthcare are as follows:

- **Improving prognosis:** Prognosis in clinical practice involves predicting the expected development of a disease, including the progression of symptoms, potential complications, ability to perform daily activities, and likelihood of survival. ML models can utilize multimodal patient data, such as phenotypic information, genomic data, proteomic data, pathology test results, and medical images, to facilitate disease prognosis [2, 3, 4]. ML models have been extensively developed for the identification and classification of various types of cancers in order to identify their prognosis.
- **Improving diagnosis:** Machine learning can aid physicians by offering second opinions. ML learning algorithms can analyze medical images, such as X-rays or MRI scans, and use pattern recognition to identify specific diseases. This can assist professionals in making quicker and more accurate diagnoses, ultimately improving patient well-being. For example, machine learning has been applied to tasks such as diagnosing diabetic retinopathy, detecting metastases from breast pathology, and phenotyping from observational data [5]. ML models can also be used to extract clinical features from Electronic Health Record data for facilitating the diagnosis process [6, 7, 8].
- **Developing new treatments/drug discovery/clinical trials:** Deep learning models can accelerate drug discovery and the development of new treatments. Since the process of preparing high-quality medical reports can be tedious and time consuming, different ML-based natural language processing (NLP) techniques have been used for annotating clinical radiology reports [9, 10, 11]. Machine learning can also analyze data from clinical trial and uncover previously unknown side effects of drugs, which eventually enhances patient care. ML generally improves the safety and effectiveness of various medical procedures [12].
- **Reducing costs:** Healthcare organizations can leverage machine learning technologies to improve the efficiency of healthcare delivery, leading to cost savings. ML can also be used to optimize resources and reduce wastefulness in the healthcare system, for example, by learning how to schedule appointments efficiently [9, 10, 11].
- **Improving care:** Machine learning can enhance the quality of patient care by proactively monitoring patients and detecting anomalies. These systems can provide alerts to medical devices or process electronic health records when there are changes in a patient's condition, ensuring timely and appropriate care [5]. ML techniques have been developed for real-time health monitoring such as human activity recognition with application to remote monitoring of patients using wearable devices [13].

Although machine learning applications in healthcare have already been gaining momentum, their full potential is still being realized. As we collect ever-growing clinical data sets, machine learning will become increasingly important to benefit from such data and deliver efficient and effective care. However, the application of machine learning in healthcare still has many *challenges* that need to be addressed before deployment [14].

- **Causality and interpretability:** One of the key challenges in healthcare is the need to answer causal questions. Many important healthcare problems require algorithms that can answer "what if?" questions about the outcomes of specific treatments (e.g., asking a question about what will happen if a doctor prescribed treatment A instead of treatment B) [15, 14]. Classical machine learning algorithms are not designed to handle such causal questions, and addressing this challenge requires reasoning about and

learning from data through the lens of causal models. Causal reasoning allows us to estimate the causal effect of certain variables on the target output, which can help identify factors that have a direct impact on patient outcomes or specific disease conditions [16, 17]. By considering causal relationships, we can make more informed decisions and better enforce fairness in predictions. For example, by understanding the causal relationships between sensitive attributes (such as race or gender) and the target output, we can assess whether the predictions are influenced by unfair biases. Causal inference methods have the potential to enhance the interpretability, fairness, and trustworthiness of ML models in healthcare.

- **Class imbalance and bias:** Most life-threating health conditions are naturally rare and diagnosed once in many (thousands to millions) patients which yields very imbalanced datasets. If classes are imbalanced in the training data, then the model's outcomes will also be biased to certain categories. Prediction biases in healthcare will have profound societal consequences and must, therefore, be mitigated.

- **Limited data:** The size of datasets used for training ML/DL models is not up to the required scale in general. Small and labeled datasets for specific tasks are usually available, but often result in algorithms that tend to underperform on new data [5]. In these cases, techniques for heavy data augmentation have been shown to be effective at helping algorithms generalize, but the distribution of transformed data often diverges from the underlying actual distribution of the training data which is usually unknown [18]. Similarly, large but unlabeled datasets are also easier to collect, but will require a shift towards improved semisupervised, unsupervised, or transfer learning techniques.

- **Data sparsity:** Another challenge is dealing with missing data. In healthcare, data is often incomplete or missing due to various reasons (e.g., unreported or very noisy samples), and this can also introduce biases and impact the performance of machine learning models. Missing values can negatively impact model performance.

- **Unreliable annotation:** Defining reliable outcomes is another important consideration in healthcare machine learning. Outcomes are used to create labels for supervised prediction tasks and to define cohorts in clustering tasks. However, clinicians like expert radiologists are rare professionals and hard to engage in secondary tasks like data annotation. As a result, less skilled personnel or ML/DL automated algorithms are usually employed for data labelling, which often leads to many problems such as coarse-grained labels, class imbalance, label leakage, and misspecification [18]. It is essential to create reliable outcomes from heterogeneous data sources and to understand the clinical relevance of these outcomes.

- **Distribution shifts:** In realistic healthcare settings, distribution shifts are common and can have a significant impact on the performance of machine learning (ML) models. For example, when ML models trained on images from one imaging center are deployed on images from different centers, the performance of the models tends to degrade. This is because the imaging data from different domains may have variations in acquisition protocols, equipment, and patient populations, leading to differences in the underlying distributions. Similarly, in predictive healthcare, ML models are typically developed using historical patient data, but they are then tested on new patients. This raises questions about the effectiveness and generalizability of the ML predictions. Distribution shifts can be addressed by domain adaptation and transfer learning [19].

- **Security and privacy:** The security and privacy problems of ML can be classified into three main categories: (1) confidentiality, (2) integrity, and (3) availability problems [18]. Confidentiality issues include model stealing and training-data extraction (e.g., membership inference [20] and data-reconstruction attacks [21, 22]). Training-data extraction leads to privacy problems which are both a sociological as well as a technical issue and must be addressed jointly from both perspectives. Integrity problems include model evasion (after deployment), model poisoning (during training) [18], and manipulating explanations both in training [23] and testing time [24]. Both attacks aim to manipulate model behavior on some samples after deployment. Finally, availability attacks include creating sponge examples that cause increased latency and resource consumption (e.g., cloud usage) in the deployment phase [25].

  The fact that ML models are neither secure nor robust hinders significantly their practical deployment in critical healthcare applications dealing with sensitive personal data. Indeed, the aforementioned problems

pose not only regulatory, ethical and legal challenges, related to privacy and data protection, but also technical ones. Perfectly eliminating these threats is a non-trivial, and sometimes impossible task without sacrificing model quality. However, ensuring the security of ML models and health data are paramount to building trust in these technologies in order to facilitate their widespread adoption in the industry.

In summary, machine learning holds great potential for healthcare, but careful consideration of the unique technical challenges and alignment with clinical needs is necessary for successful implementation.

3.1 Deep Learning

Historically, constructing a machine-learning system required domain expertise and human engineering to design feature extractors that transformed raw data into suitable representations from which a learning algorithm could detect patterns. In contrast, deep learning [26], which is a subfield of machine learning, is a form of representation learning in which a machine is fed with raw data and automatically develops its own representations needed for pattern recognition [27]. Deep learning has seen a dramatic resurgence in the past decade, largely driven by increases in computational power and the availability of massive new datasets. Deep-learning models can accept multiple data types as input, which makes them particularly appealing to process heterogeneous healthcare data. Deep learning techniques have had a significant impact on computer vision, natural language processing, and reinforcement learning. These results can be leveraged by various medical applications including medical imaging, processing EHRs, robotic-assisted surgery, genomics, and real-time health monitoring [28, 5, 29, 30, 31, 32].

3.1.1 Medical Imaging

The purpose of medical-image analysis is to assist clinicians and radiologists for the efficient diagnosis and prognosis of diseases. The prominent tasks in medical image analysis include detection, classification, segmentation, retrieval, reconstruction, and image registration.

Convolutional Neural Network [26] have been successfully applied in medical imaging for various diagnostic purposes. They have achieved remarkable accuracy comparable to physicians and have the potential to assist in clinical decision-making and improve patient outcomes.

CNNs trained on medical imagery, including radiology, pathology, dermatology, and ophthalmology, can aid physicians by providing second opinions and identifying concerning areas in images [5, 28]. CNNs have achieved human-level performance in object-classification tasks and demonstrated strong performance in transfer learning, where they leverage pre-training on unrelated datasets and then fine-tuned on medical images. These models have physician-level accuracy in diagnosing a range of conditions, including melanomas, diabetic retinopathy, cardiovascular risk, breast lesion detection, and spinal analysis [5, 30].

3.1.2 Electronic Health Record

In healthcare, natural language processing is mainly used in applications related to EHRs. EHRs are becoming increasingly prevalent and contain vast amounts of valuable data [5]. By parsing and organizing the data temporally and across patients, deep-learning models can answer high-level medical questions about relevant past medical history, identify current problem list, and recommend interventions [33, 34].

Most predictive models in EHRs have used supervised learning on structured data like lab results, diagnostic codes, and demographics [9, 10, 11]. Models that incorporate the temporal sequence of events in a patient's record have been used to predict future medical incidents. Large-scale Recurrent Neural Networks (RNN) are already demonstrating impressive predictive results by combining structured and unstructured data (e.g., clinical notes) in a semi-supervised manner [34]. These models outperform other techniques in tasks like mortality prediction, readmission prediction, length of stay estimation, and diagnosis prediction.

3.1.3 Robotic-assisted surgery

Robotic-assisted surgery can be enhanced using deep reinforcement learning, allowing robots to perform repetitive and time-sensitive surgical tasks with more adaptability, efficiency and precision. Computer vision models and reinforcement learning algorithms can enable robots to perceive surgical environments and learn from surgeons' motions, automating repetitive surgical tasks like suturing and knot-tying [35, 36]. These techniques are especially useful in fully autonomous robotic surgery or minimally invasive surgery, like laparoscopic surgery. Deep imitation learning [37], RNNs [38] [39], and trajectory transfer algorithms can automate teleoperated manipulation tasks in these procedures [40].

However, challenges exist in accurately localizing instrument positions and orientations in surgical scenes and collecting sufficient training data, especially for more general surgical tasks [5].

3.1.4 Real-time health monitoring

Real-time monitoring of critical patients plays a crucial role in their treatment process. There is growing interest in continuous health monitoring using wearable devices, IoT sensors, and smartphones. In this setup, health data is collected from the wearable device and smartphone and transmitted to the cloud for analysis using ML/DL techniques. The analyzed outcomes are then sent back to the device for appropriate actions or interventions.

One example of such a system architecture is presented in a framework described in [41]. The system integrates mobile and cloud technologies to monitor heart rate. Another study [13] provides a review of different ML techniques for human activity recognition, specifically focusing on remote monitoring of patients using wearable devices.

While sharing health data with cloud platforms for further analysis brings numerous benefits, it also raises important concerns regarding privacy and security.

3.1.5 Genomics

Deep learning has been adapted for genomics, allowing for improved analysis of various genomic measurements and benefiting biomedical applications, such as disease prediction, pathogenicity assessment, and biomarker analysis. One area where deep learning is valuable is in genome-wide association (GWA) studies. GWA studies aim to identify genetic mutations associated with specific traits. Deep learning algorithms can analyze large patient cohorts and latent confounders [42]. In the future, integrating external modalities like medical images or molecular phenotypes may further enhance GWA studies. Phenotype prediction from genetic data, including complex traits and disease risk, can also be improved with deep learning [43]. By integrating additional modalities like clinical history, and wearable device data into phenotype prediction too, deep learning models can enhance accuracy [44].

3.2 Federated Learning

Federated Learning [45] is a branch of Machine Learning, where multiple entities collaboratively train a joint model in such a way, that their potentially private and sensitive data never is never shared with other participants in the protocol. In Machine Learning, a single model is trained iteratively on a single dataset until convergence. Instead, in Federated Learning, in every training round (i.e., epoch) data owners (called clients) train a common model locally and share the corresponding model updates which are aggregated into a single global model for the following round. The process is aided by a trusted server (called aggregator) as illustrated in 1.

There are several angles Federated Learning [46] systems can be differentiated, such as Vertical or Horizontal, and Cross-silo or Cross-device. The first angle is concerned with the feature space of the underlying datasets, the second depends on the number of clients. More specifically, client datasets in Vertical Federated Learning have different feature space, while in Horizontal Federated Learning the datasets have the same feature space

across all clients. Concerning the participants, in Cross-silo Federated Learning there are typically a handful of reliable participants with significant computational power and sophisticated technical capabilities. On the other hand, in Cross-device Federated Learning there can be even millions of potential unreliable participants, with small datasets and limited computational resources. To tackle both ends of these spectrums Federated Learning has to be flexible. Its algorithmic model is presented below. Indeed, several possible modifications are possible, such as the model selection (e.g., convolutional neural networks, recurrent neural networks, etc.), the model initialization (e.g., random, pre-trained, etc.), the convergence criteria (e.g., fixed rounds, accuracy-based, etc.), the broadcasting method (e.g., entire model, only first layers, etc.), and so on. Below we highlighted three aspects: the client selection method, the training mechanism, and the aggregation technique.

1. The aggregator server initializes the model, i.e., determination of the hyperparameters.
2. In order to converge the model the following are necessary:
   (a) The aggregator broadcasts the model to *some* clients.
   (b) Those clients *train* that model on their local dataset and share the result with the aggregator.
   (c) The aggregator *aggregates* the received model updates into the new global model.
3. The final model is broadcasted to all participating clients.

3.2.1 Client Selection Methods

Concerning client selection, in Cross-silo Federated Learning all participants are selected as there are only a few of them and they are all reliable. In contrast, only a fraction of clients is selected in Cross-device Federated Learning to battle the emerging communication bottleneck. The server can either randomly pick the desired number of participants from a pool of devices (the de-facto standard mechanism) or use some algorithm for client selection. In the following we highlight a few strategies, for a more comprehensive list we refer to [47].

- Federated Client Selection [48]: FedCS solves a client selection problem with resource constraints, i.e., it select as many clients as possible within a specified deadline to accelerate performance improvement.

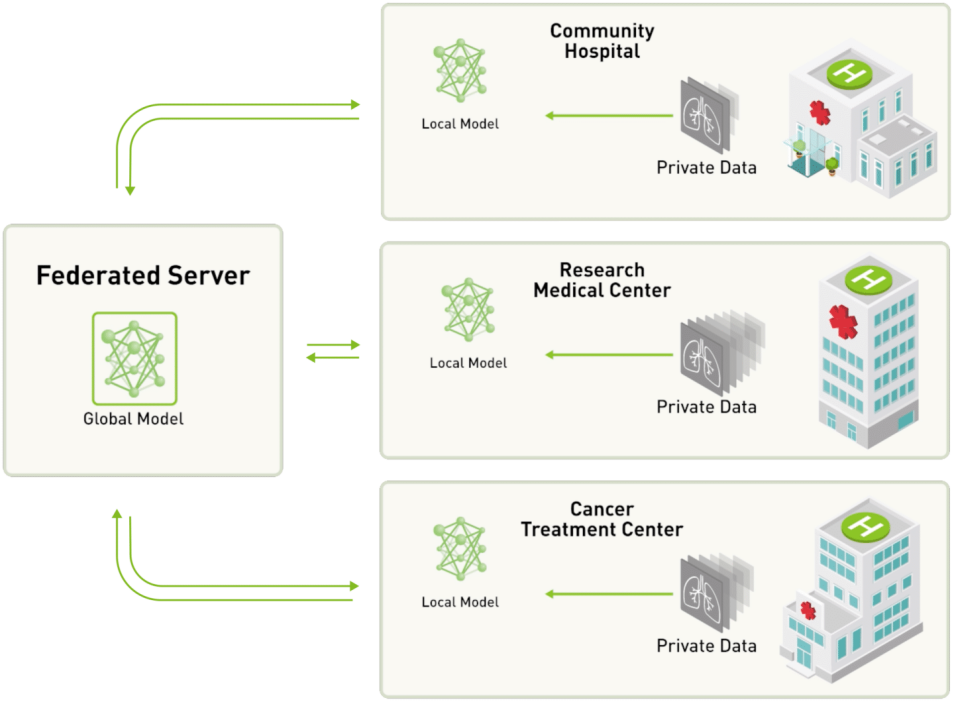

Figure 1 – Illustration of Federated Learning in the Health Domain.

- Power of Choice [49]: PoC is a selection method based on local loss to tackle the trade-off between convergence speed and solution bias.
- Oort [50]: Oort prioritizes the selection of those clients who improve the model's accuracy the most in order to enhance the convergence speed.
- Online Client sElection and bAndwidth allocatioN [51]: OCEAN formulates a stochastic optimization problem for joint client selection and bandwidth allocation under long-term client energy constraints to achieve long-term performance guarantee.
- Federal Optimization via LB [52]: FOLB performs client selection based on the correlation between local update and global update to optimize the expected convergence speed.

3.2.2 Optimization Strategies

Each selection method of the above requires specific optimization-process. For instance, the number of local rounds, or the training mechanism itself could be subject to change (e.g., SGD, Adam). In the following we highlight a few methods, for a more details we refer to [53].

- First-order optimization algorithms: these methods rely on the first derivative (gradient) for optimization (i.e., to choose the direction to move in the search space). The two core aspects of these methods are 1) how the gradient is calculated and 2) how it is scaled.
  - Gradient Descent (GD) / Stochastic Gradient Descent (SGD) / Batch Gradient Descent (BGD): In GD the gradient is calculated directly. On the other hand, in SGD and BGD, it is rather appropriated, using prediction error: in SGD it is based on one sample, while in BGD it is based on multiple.
  - Adaptive Gradient (Adagrad) / Root Mean Square Propagation (RMSProp) / Adaptive Moment (Adam): During training the scale of the gradients can be adapted automatically. Adagrad adjust the weights such that a high gradient will have low learning rate and vice versa. RMSprop adjusts Adagrad by reducing its monotonically decreasing learning rate. Adam takes RMSProp one step further by incorporating momentum into the scale.
- Second-order optimization algorithms: these methods, besides the first they also rely on the second derivative (Hessian) for optimization.
  - Newton's Method: It is an iterative approach to find a root of a function (i.e., the gradients). It start with an initial guess, and calculates the function's gradient (i.e., the gradient of the gradient) to update the guess with the interception of this tangent line with the axis.
  - Quasi-Newton methods: when the second order derivatives (i.e., Jacobian or Hessian) are not feasible to compute, they must be approximated. Such method is Broyden-Fletcher-Goldfarb-Shanno (BFGS), which estimates the inverse Hessian matrix to limit the search within the variable space. An enhanced version is Limited-memory BFGS (l-BFGS), which stores only a few vectors to represent the approximated inverse matrix implicitly.
- Besides, there are optimization algorithms which do not make use of the derivatives at all. The core reason for their use is the unavailability of the gradients, i.e., it cannot be calculated due to complexity or other real-world reasons.
  - Direct / Stochastic / Population search algorithms: Direct methods are deterministic, as they search the space using geometric shapes or decisions, e.g. patterns. On the other hand, Stochastic methods are non-deterministic, as they utilize randomness during the search. Population methods are similar, but they maintain a pool of candidate solutions for exploring the optima.

3.2.3 Aggregation Techniques

Lastly, one can set how the server should aggregate the models. The classical solution is Federated Averaging (FedAvg) [54], where aggregation is implemented as an average function, so the weights of the different local models are averaged to provide new weights and, thus, a new model. In the following we highlight a few variants. For more comprehensive lists we refer to [55, 56, 57].

- Proximal Federated Learning [58]: FedProx tackles the problem of data heterogeneity. It is based on the usage of the proximal term, as per the average convergence around the proximal terms are local sub-problems to effectively restricts the effect of variable local updates.
- Federated Multi-Task Learning [59]: MOCHA is a framework for federated multi-task learning.
- Federated Personalization [60]: FedPer tackles the personalization issue of FL. The principle is that the model is split in base and personalized layers, where the latter are not communicated to the server, so only the base layers are aggregated using transfer learning methodologies.
- Federated Match Averaging [61]: FedMA aims to update the global model via layer-wise matching and aggregation of inner model components, namely neurons. Although it outperforms several other aggregation mechanisms, it only works on simple neural networks (such as CNN and LSTM based models) due to its specificities.
- One-Shot Federated Learning[62]: OSFL tackles the communication bottleneck of FL. Essentially, the locally trained models are aggregated only once to form an ensemble model.

3.2.4 Incentive Mechanisms

Incentive mechanisms in federated learning refer to the strategies and mechanisms used to motivate participants to actively engage in the collaborative learning process and contribute their resources, such as data and computing power [63].

In cross device federated learning the challenge of incentive mechanisms is profound. Here, the participants are often individuals who may have limited motivation to contribute their resources. Hence, designing effective incentive mechanisms becomes crucial to encourage active participation and ensure the success of the federated learning process.

On the other hand, in the cross silo setting, incentive mechanisms are often less of an issue, primarily because the participants have clear boundaries and interests, which are more likely to be aligned. The benefits of participating, such as access to the shared model's improved performance or the opportunity to leverage a larger and more diverse dataset, act as natural incentives for participants. For instance, in healthcare applications, hospitals or research institutions may collaborate to improve disease diagnosis accuracy or advance medical research, which benefits each participating entity individually.

However, it is worth noting that even in the cross silo setting, there can still be challenges [64]. Issues may arise when participants have varying levels of contributions or when resource allocation becomes uneven. In such cases, designing fair incentive mechanisms that distribute benefits proportionally and ensure equitable participation remains important [65].

A key concept is the Shapley value [66], which was designed to allocate goods to players proportionally to their contributions. A high-level summary of the role of the Shapley value within machine learning is presented in [67]. The main disadvantage of the Shapley value is its exponential computational requirement, which makes it unfeasible in most usecases. Although several approximations were proposed in the literature using sampling [68], gradients [69] and influence functions [70], most of them utilize fine-grained individual level information, which makes them unfeasible for privacy-focused use-cases. Recently, privacy-enhanced solutions have also emerged [71, 72], but they leave plenty of room for improvement. Today, no federated learning framework exist that ensures security and privacy while enabling a non-trivial contribution score allocation amongst the participants.

3.3 Safety of Federated Learning

Federated learning, with its distributed and collaborative nature, brings forth several privacy and security concerns that need to be carefully addressed. These concerns arise due to the decentralized processing of sensitive data and the potential vulnerabilities associated with sharing models and aggregating information across multiple participants. Understanding and mitigating these concerns is crucial for ensuring the privacy and security of federated learning systems [73].

3.3.1 Threat Models of Federated Learning

In general, there are three aspects of risk to be considered within federated learning: 1) The privileges of each user on the model 2) what access on modifications each user has 3) and finally at which stages of the process has access to. It is important to assess and address these threats in the design and implementation of federated learning systems to ensure robust security and privacy protections. Mitigation strategies can be employed to counter these threats and maintain the integrity and privacy of the federated learning process [74].

Considering the privileges of each user, an attack can be carried out by insiders (e.g., the aggregator server or any participants) or outsiders (e.g., an eavesdroppers on the communication channel between participants and the server or a users of the final model). The insider threat is more severe, especially when executed by the server. Relative to the participants, the source of an attack can be a single client or multiple clients launching a coordinated attack.

Concerning the access on modifications each user has (i.e., the capability of the adversary), the two most common threat models in federated learning is below.

- Honest-but-Curious (or Semi-Honest) Participants: This threat model assumes that participants in the federated learning process follow the protocol but may try to gain additional information by analyzing the exchanged data or model updates.
- Malicious Participants: This threat model considers participants who actively deviate from the protocol and engage in malicious behavior by intentionally provide incorrect or manipulated model updates to undermine the integrity of the federated learning system.

Considering which stage of the process each user has access to, an attack can happen at training and at inference time. In the former, the attacker attempt to learn, influence, or corrupt the FL model itself for instance by running an active data or model poisoning attacks. In the latter, the attacker targets the gradients (individual or aggregated) to uncover sensitive details about the underlying datasets of other clients. The effectiveness of such attacks is determined by the available information about the model: white-box and black-box corresponds to full and query access, respectively.

3.3.2 Security Concerns of Federated Learning

The participants in Federated Learning may have misaligned incentives, which could lead to malicious behaviour. There are plenty of attack types that aim to compromise the system and even more mechanisms to realize those goals. Below we give a non comprehensive view of the corresponding notions [75].

- Sybil Attack [76]: an adversary creates multiple fake participants to disrupt or manipulate the collaborative learning process. The attacker aims to control a significant portion of the federation by simulating a larger number of participants.
- Byzantine Attack [77]: it involves participants behaving maliciously by providing incorrect or misleading updates to the federated learning system. These attacks can disrupt the training process, compromise the model's accuracy, and hinder the convergence of the learning algorithm.

- Adversarial Attack [78]: specially crafted input data that are intentionally designed to cause misclassification or incorrect behavior in machine learning models. These examples are created by adding imperceptible perturbations or modifications to legitimate input samples with the goal of deceiving the model.
- Poisoning Attack [79]: malicious participants intentionally inject malicious updates into the learning process. This can compromise the integrity and performance of the trained model and lead to erroneous or biased results.
- Backdoor Attack [80]: a hidden vulnerability or malicious behavior intentionally inserted into a model by an adversary. A backdoor attack aims to compromise the integrity and security of the federated learning process, allowing the attacker to manipulate the model's behavior during inference.

Ensuring the integrity of the federated learning system and implementing robust defense mechanisms to detect and mitigate such attacks is crucial for maintaining the security and reliability of the collaborative learning process. Below we highlight few corresponding core concepts.

- Byzantine Resilience [81]: the ability of the system to withstand and mitigate malicious behaviors or attacks from participants
- Adversarial Robustness [82]: the ability of the system to withstand and mitigate adversarial attacks, particularly those involving adversarial examples.
- Certified Defenses [83]: defense mechanism that provides a formal guarantee or certification of the model's robustness.
- Ad-hoc defenses strategies: these strategies provide additional layers of security and resilience against potential security threats.
  - Participant Validation and Reputation Systems: These can help verify the trustworthiness of participating entities. This involves evaluating credibility and track record of participants to identify and exclude potentially malicious or unreliable participants from the collaborative process.
  - Data Sanitization and Anomaly Detection: These can help identify and filter out potentially malicious contributors. This involves carefully examining the data for anomalies or patterns that deviate significantly from expected behavior to reduce the impact of the adversary.
  - Robust Aggregation: Enhancing the aggregation process can mitigate the influence of adversaries' malicious model updates. They are designed to mitigate the impact of outliers, biased updates, or intentionally manipulated contributions during the aggregation step.

3.3.3 Privacy Concerns of Federated Learning

One of the primary privacy concerns in federated learning is the exposure of sensitive data. Each participant in the federated learning process holds their own local data, which may include personal, sensitive, or proprietary information. Without appropriate safeguards, the sharing and aggregation of this data can pose privacy risks. Unauthorized access to participant data or the leakage of private information during the model training or aggregation process can lead to privacy breaches and compromises. Although Federated Learning provides privacy by design to some extent, many researchers have developed several techniques for this concept to show unintended information leakage [84].

- Model inversion [85]: it exploits the outputs of a machine learning model to infer sensitive information about individuals by reconstructing attributes of the datasets used for training.
- Membership inference [86]: the attacker aims to infer membership information (i.e., determine whether a specific data sample was part of the training dataset) by analyzing the model's outputs for the target data samples (i.e., exploiting patterns or discrepancies in the model's behavior).

- Property inference [87]: instead of individual level information, this attack aims to infer sensitive or confidential properties of the training data.
- Reconstruction attacks [22]: an adversary attempts to reconstruct the original training data by leveraging the trained model and its outputs, parameters, or gradients.
- (Hyper)parameter inference [88]: instead of the training data, this attack aims to infer sensitive or confidential information about the parameters of a machine learning model.

There are a handful of privacy-preserving techniques that either partly mitigate or entirely prevent the above listed attacks. A few common privacy-preserving mechanisms utilized in federated learning are below.

- Differential Privacy [89]: This Privacy-enhancing technology injects noise to the training process to prevent the identification of individual data samples. It ensures that the presence or absence of any particular data point does not significantly impact the overall model's behavior. There are several noise injection techniques, such as input, output, and objective function perturbation, as well as adding noise to the gradients. Could either be locally (e.g., noising the individual gradients) or globally (noising the aggregated gradients), corresponding to two different threat models.
- Secure Aggregation [90]: Secure aggregation techniques, such as Secure Multi-Party Computation (SMPC) or Homomorphic Encryption (HE) as described in Section 4, allow participants to aggregate their local model updates without revealing their individual contributions. This ensures that the privacy of the participants' local data is maintained during the aggregation process.
- Ad-hoc defense strategies: There are a handful of strategies which were shown empirically to mitigate to some extent the information leakage. They could be utilized before, during, and after training.
  - Before: Suppressing the sensitive data or injecting fake entries, using resistant models or model stacking / distillation techniques are all examples of this category.
  - During: These techniques manipulate the intermediate gradients. These methods are not limited to, but include compression techniques such quantilization (e.g., rounding) and sparsification (e.g., use random or Top-K elements, discard values below a threshold), gradient normalization, and regularization.
  - After: Rounding the output of the model or only returning the Top prediction also reduce the attack surface, although in Federated Learning they are less effective (due to the constant sharing of internal model updates).

Note that all these mechanisms have disadvantages: Differential Privacy could potentially change the functionality and output of the model, while Secure Aggregation does only protect the local updates, hence, information could still leak from the aggregates, which could potentially be attributed to particular participants with appropriate background knowledge. The later sections of this report are concerned with cryptographic solutions and anonymization techniques, so these are explained in more depth there.

3.4 Health Related Applications of Federated Learning

Applications of machine learning require large and diverse data sets. However, medical data sets are scarce and difficult to obtain. FL addresses this issue by enabling collaborative learning without centralising data [91]. Research has shown that models trained by FL in healthcare can achieve performance levels comparable to ones trained centrally on the union of the institutes' data sets and superior to models that are trained exclusively on single-institutional data [92].

FL has shown promise in utilizing large-scale EHRs for predictive modeling without compromising patient privacy [91, 93]. One example is the use of FL for predicting hospitalization of patients with heart-related diseases using EHR data [94]. In this approach, an FL-based decentralized scheme is implemented, where each device

(representing a hospital or healthcare provider) trains the model locally using its own EHR data. The trained model parameters are then shared with a central server, which aggregates them to update the shared model. The concept of federated autonomous deep learning (FADL) has been introduced in [95], focusing on the use of distributed EHR data for training deep learning models.

In the field of medical imaging, FL has been successfully used for tasks such as whole-brain segmentation in MRI [96] and brain tumor segmentation [92]. FL has also been employed in fMRI classification to identify reliable disease-related biomarkers [97]. Furthermore, FL has been suggested as a promising approach in the context of COVID-19, potentially enabling collaborative analysis of medical imaging data to identify imaging biomarkers associated with the disease [98].

Split learning allows for training deep learning models without sharing patients' critical data with the server. Different configurations for split learning models, such as vertically partitioned data-based configurations, have been proposed [99]. SplitNN is one framework that implements the split learning approach and ensures that sensitive patient data remains on the local device while still allowing for model training and inference.

FL has also been applied to accelerate drug discovery [100]. Drug discovery and development is a high risk process as there is a failure rate of around 90% for drug candidates that reach the clinical studies phase. Therefore, making the early stages of drug discovery more efficient and accurate holds the potential to have a significant impact on the pharmaceutical industry. The FL architecture proposed in [100] enhanced predictive Machine Learning models on decentralised data of 10 pharmaceutical companies, without exposing proprietary information.

There are several other large-scale initiatives and innovative collaborations to deploy FL for healthcare applications. The HealthChain project[1], implemented across four hospitals in France, focuses on developing and deploying an FL framework for the prediction of breast cancer and melanoma treatment response. By leveraging FL, the project aims to generate common models that can effectively analyze histology slides or dermoscopy images. The Federated Tumour Segmentation (FeTS) initiative[2] is a large-scale effort involving 30 international healthcare institutions. FeTS utilizes an open-source FL framework with a graphical user interface to improve tumour boundary detection in various cancer types. By combining data from multiple institutions, FeTS aims to enhance the accuracy and reliability of tumour segmentation models for brain gliomas, breast tumours, liver tumours, and bone lesions in patients with multiple myeloma. The Trustworthy Federated Data Analytics (TFDA) project[3] along with the German Cancer Consortium's Joint Imaging Platform[4] conducts decentralized research across German medical imaging research institutions. Another international research collaboration demonstrated the usefulness of FL for the assessment of mammograms and produced ML models that generalized across several institutes[5].

In summary, FL and related approaches offer promising solutions for healthcare applications especially if the available training data at each party is limited. FL allows for collaborative model training on decentralized data while maintaining privacy and data security, if appropriate data protection measures are taken.

3.5 Existing tools and Libraries

When considering to use Federated Learning, there are several open-source frameworks and software options available. Below we enlist some notable efforts (in alphabetical order).

- Clara (https://developer.nvidia.com/blog/federated-learning-clara/): Developed by NVIDIA, Clara Train SDK features the usage of NVIDIA EGX, the edge AI computing platform. It supports different distributed architectures, such as peer-to-peer, cyclic, and server-client. It emphasizes data privacy and security and provides a robust and secure environment for FL.

[1] https://www.substra.ai/en/healthchain-project
[2] https://www.fets.ai
[3] https://tfda.hmsp.center/
[4] https://jip.dktk.dkfz.de/jiphomepage/
[5] https://blogs.nvidia.com/blog/2020/04/15/federated-learning-mammogram-assessment/

- IBM Federated Learning (https://ibmfl.mybluemix.net/): Developed by IBM, IBMFL provides a basic fabric for Federated Learning on which advanced features can be added. It focuses on scalability and enterprise-grade capabilities, offering a comprehensive framework for distributed model training and supports integration with IBM Watson and other IBM Cloud services.
- EasyFL (https://github.com/EasyFL-AI/EasyFL): EasyFL is a user-friendly and accessible federated learning framework that simplifies the process of implementing federated learning algorithms and workflows. It provides a high-level API and pre-built components to enable developers to quickly build and deploy federated learning systems with ease.
- FedBioMed (https://fedbiomed.gitlabpages.inria.fr/): Developped by research centre INRIA, FedBioMed is a federated learning framework designed specifically for biomedical applications, enabling collaborative model training on distributed healthcare data while preserving privacy and security.
- Federated AI Technology Enabler (https://fate.fedai.org/): Developed by Webank, FATE is an open-source project that provides a secure computing framework to support the federated AI ecosystem. It supports multiple algorithms and deployment scenarios as well as it implements multiple secure computation protocols. It enables easy big data collaboration and model training across distributed networks with data protection regulation compliance.
- FedLearner (https://github.com/bytedance/fedlearner): Developed by byte-dance / Tencent, Fedlearner is a collaborative machine-learning framework that enables joint modeling of data distributed between institutions. It is scalable and efficient with a modular architecture and support to multiple machine learning frameworks, making it adaptable to different use cases.
- FedML (https://www.fedml.ai/): FedML is a comprehensive federated learning research library that provides a wide range of tools, algorithms, and benchmarks for developing and evaluating federated learning systems. It aims to facilitate the development of robust and efficient federated learning solutions across various domains while promoting collaboration and advancing the state of the art in the field.
- Flower (https://flower.dev/): Originally developed by the University of Oxford, Flower is a open sourced framework for building federated learning systems. It emphasizes simplicity and flexibility (i.e., customizable, extendable, and framework-agnostic) while offering a user-friendly interface and easy integration with existing ML pipelines. It support for various deployment scenarios.
- OpenFL (https://github.com/intel/openfl): Initially developed by Intel and hosted by the Linux Foundation, OpenFL is designed for large-scale collaborations by providing a flexible infrastructure for distributed model training with features like dynamic participant management and efficient communication protocols. It is a flexible, extensible, and easily learnable tool for data scientists.
- PaddleFL (https://github.com/PaddlePaddle/PaddleFL): PaddleFL is an open-source federated learning framework where researchers can easily replicate and compare different algorithms. Built for PaddlePaddle and based on Kubernetes, so it provides distributed training and flexible scheduling of training jobs. It also offers efficient communication protocols, advanced encryption techniques, and supports various scenarios.
- PySyft (https://docs.openmined.org/pysyft/): Developed by the OpenMined community, PySyft is an open-source library built on PyTorch and Tensorflow that provides tools for federated learning and encrypted computations (e.g., SMPC). It further enhances the secure and privacy-preserving collaboration by utilizing differential privacy techniques.
- Substra (https://www.substra.ai/): Developed by a multi-partner research project around Owkin, Substra is a federated learning software framework focusing on the medical field for data ownership and privacy by enabling the training and validation of machine learning models on distributed datasets. It provides a marketplace for data scientists to securely exchange models and datasets while maintaining data privacy.

- TensorFlow Federated (https://www.tensorflow.org/federated): Developed by Google, TFF is an open-source framework on which Android mobile keyboard predictions is based. It provides abstractions for federated computations and supports various federated learning algorithms.

These platforms offer different strengths, such as advanced security measures, scalability, simplicity, compatibility with specific ML frameworks, or support for specific deployment scenarios. The choice depends on specific requirements and preferences of users and organizations.

3.6 Unbiased Federated Learning Approaches

3.6.1 Bias taxonomy

3.6.1.1 Biases and discrimination

According to US law, the fairness of a decision-making process is often understood through two distinct notions: disparate treatment and disparate impact. On one hand, we want a negative answer to the question: Does the process based on subjective individual elements make treatment disparate? On the other hand, we want a negative answer to the question: Do the results show differences between people with different sensitive attributes?

For example, we want the outputs and errors of the machine learning model to be similar for two subpopulations. In the same way, two similar individuals should receive closed model decisions [6]. However, by default, machine Learning models tend to reproduce and amplify biases, leading to unfair outputs.

These biases can come from the data: there are known biases such as selection bias when the sampling is poor, historical biases, when a population is disadvantaged, etc.

But biases can also be due to algorithms: some recommendation algorithms lock people in bubbles instead of offering them new possibilities. Also, if a data set already has biases, future observations are less likely to contradict past predictions. This is because the new data collection could be driven by past decisions of the machine learning model, leading to confirmation bias.

Moreover, labels are often created by humans and a model will tend to reproduce those biases to increase its performance. At group-level discrimination, a minority group could be penalized because of a small sample size or features less informative for that particular group characteristics. The consequence is a disparate result between this group and the majority group.

As shown in [101], removing sensitive features from the training dataset can be insufficient. Indeed, these features can be correlated with others, and the model will find the new associations between the features to improve the loss function. So even if no sensitive feature is easily identified as a potential risk for fairness, sensitive information can still be spread or hidden through features and difficult to recognize. A popular solution is to keep them during the learning phase of the model as a constraint to assist the training towards a fairer direction. Then, during the testing phase, sensitive attributes can be used to measure the model fairness.

The effect of discrimination can be direct or indirect, which is mainly why removing sensitive features is not a solution, although it may be required by the law. We will see that there are several sources of discrimination that can affect a population. Some forms of discrimination are systemic discrimination that refers to policies, customs of behavior that are part of an organization's culture or structure that perpetuate discrimination against certain subgroups of the population. Conversely, statistical discrimination occurs when decision-makers use average (assumed) group statistics to judge an individual belonging to that group. This typically occurs when decision-makers (e.g., employers, or law-enforcement officers) use the obvious and recognizable characteristics of an individual as an indicator of hidden or more difficult to determine characteristics.

[6] Closed decision making is the term used when the person who is in charge of the decision operates in a group known to him

3.6.1.2 Data-to-algorithms bias

For the bias coming from the data, we propose the following taxonomy from [102] considering seven main categories.

Measurement bias Measurement or reporting bias arises from how we choose, utilize, and measure particular features. For instance, it is known that some city areas are more controlled by the Police, and consequently offenses are over-estimated in this area compared to area less controlled by the Police.

Omitted Variable Bias This bias occurs when some important variables are left out of the dataset and/or the model. Consider as an example an inter-company canteen that uses a model to predict the inventory needed to provide the meal. However, the explanatory variables of their model lack the exceptional closing dates of one of the companies. When the company in question is closed, the prepared stocks will be oversized compare to the actual demand.

Representation Bias This bias stems from how we sample from a population during the data collection process. Non-representative samples lack the diversity of the population, with missing subgroups and other anomalies. For instance, if we perform biometry with data collected in one specific local airport like e.g. Rennes or Clermont Ferrand, we will have an over-representation of some population sub-groups at the expense of the others population sub-groups.

Aggregation Bias Aggregation bias occurs when false conclusions are drawn about individuals from observing the whole population. A well-known aggregation bias is illustrated by the Simpson paradox. Here, we consider two treatments for a cancer: one based on aggressive chemotherapy and the other based on radiotherapy. At aggregate level, when we look at the total treatments results, we see that there is more cancer remission for patients who receive radiotherapy (61 patients) instead of chemotherapy (39 patients). However, if we look at results at disaggregated level, we see that the two sub groups of patients have not received the same kind of treatment. Patients with state 1 and 2 cancer, who are more likely to be in remission, overwhelmingly receive radiotherapy treatment while patients with severe condition received chemotherapy in large majority. If we consider only state 1 and 2, then patients receiving chemotherapy have a better probability of remission than patient receiving radiotherapy. Similarly, if only patients with state 3 and 4 cancer are considered, patients receiving chemotherapy have a better probability of remission than patients receiving radiotherapy. However, due to the distribution of the treatments, at aggregate level, the result is reversed. We extend this example in Table 4.

	Radiotherapy	Chemotherapy
Patient with state 1 and 2 cancer	60 remissions on 100 patients (60%)	9 remission on 10 patients (90%)
Patient with state 3 and 4 cancer	1 remission on 10 patients (10%)	30 remissions on 100 patients (30%)
Total	61 remissions on 110 patients (55,45%)	39 remissions on 110 patients (35,45%)

Table 4 – Illustration of the Simpson paradox

Sampling Bias Similar to the representation bias, this bias occurs when non-random sampling of subgroups is performed. Due to sampling bias, trends estimated for one population may not generalize to data collected from a new population. We take snowball sampling as an example, a technique where existing study subjects recruit future subjects from among their acquaintances. Thus, the sample group is said to grow like a rolling snowball. As the sample builds up, enough data are gathered to be useful for research. This sampling technique is often used in hidden populations, such as drug users, which are difficult for researchers to access. However, if we consider this approach to study a large heterogenous population, e.g. to make a political poll for a national election, there will be obvious bias.

Longitudinal Data Fallacy Researchers analyzing temporal data must use longitudinal analysis to follow cohorts over time to learn their behavior. Instead, temporal data is often modelled using cross-sectional analysis, that can introduce a Simpson's paradox (see explanation above).

Linking Bias This bias concerns the study of networks (social, transport, etc.). It occurs when network attributes obtained from user connections, activities, or interactions differ and misrepresent true user behavior, such as when they disregard connection intensity.

3.6.1.3 Algorithms and Algorithms to user interaction bias

For the bias coming from the algorithm, we propose the following taxonomy from [102] considering two main categories.

Algorithmic bias This bias is not present in the input data and is added by the algorithm. The algorithmic design choices, such as use of certain optimization functions, regularizations, choices in applying regression models on the data as a whole or considering subgroups, and the general use of statistically biased estimators in algorithms, can all contribute to biased algorithmic decisions that can bias the outcome of the algorithms. A such well-known and simple bias is to estimate the variance of a Gaussian distribution with the (uncorrected) empirical variance, which is a biased estimator of the variance.

Evaluation bias This bias occurs during the algorithm evaluation and happens when an inappropriate process is used for model evaluation (bias present in dataset used for evaluation, inappropriate evaluation metrics, results insignificant, etc.).

Others bias Some others bias exists like user-interaction bias, emergent bias and popularity bias, that will not be described further in this document.

3.6.1.4 User to Data bias

For the bias coming from the user, we propose the following taxonomy from [102] considering two main categories.

Historical bias Historical bias is bias that already exists, such as socio-technical problems in the world. It can infiltrate the data-generation process even given a perfect sampling and feature selection. Historical data bias occurs when socio-cultural prejudices and beliefs are mirrored into systematic processes. This becomes particularly challenging when data from historically-biased sources are used to train machine learning models. For example, if manual decision systems give certain groups of people poor credit ratings, then using this manually labelled data to train the automatic system, may cause the automatic system to reproduce and even amplify the original system's biases. For instance, natural language processing models can mirror and amplify the existing bias in a large textual dataset and can produced gender-biased analogies like: man is labelled doctor versus woman is labelled nurse.

Population bias Population bias arises when statistics, demographics, representatives, and user characteristics are different in the user population of the platform compared to the original target population. For instance, there is a population bias if we train a predictive maintenance model on data collected on a simplified test-bed and apply it in the real world.

Others bias Many other forms of user bias can come from this dimension (self-selection bias, social bias, behavioral bias, temporal bias, content production bias, etc.) but we will not describe them here.

3.6.1.5 Fairness in Federated Learning context

When considering Federated Learning, fairness under the scope of biasing is one of the most important requirement. Fairness is associated with the FL server and is related with the way such server is biased (or not) on the way that it interacts with the FL clients. Bias from the server's side may result in unfair treatment of clients that discourages them from actively participating in the learning process and damages the sustainability of the FL ecosystem. Therefore, the topic of ensuring fairness in FL is attracting a great deal of research interest [103].FL Fairness can have multiple meanings:

- Performance fairness (also denoted as FL faireness accuracy): encourage a uniform accuracy distribution across participants. For example, when multiple hospitals collaborate to learn a model predicting a patient's response to chemotherapy, such as in [104], we do not want the resulting model to perform poorly in any of the hospital datasets, even if the overall accuracy is satisfactory.
- Collaboration fairness: its aim is to provide the participants with the higher contribution to the learning process with higher rewards. This kind of fairness is of interest when the participants are self-interested or even competitors, like banks or states.
- Model fairness: The classical understanding of fairness, meaning that the trained model has no discrimination regarding some specific sub-groups or sensitive features.

Here, we focus on the model fairness; in a federated learning process each party trains its local model the same way it would be trained in centralized machine learning. Thus, all the different kinds of bias described below also apply to Federated Learning. Furthermore, federated learning approaches face unique challenges, described in [105] that introduce new sources of bias.

Data Heterogeneity Each participant has its own dataset and in a real-case scenario, it is likely that each party's subgroup of data differs greatly from the overall data composition from all parties involved. Besides, in cross-device settings, a participant can drop out from the learning process for various reasons (low battery, poor connection etc). Consequently, the overall and relative data composition may be constantly changing, affecting how the global model learns bias.

Fusion Algorithms In most Federated Learning approaches, an aggregator incorporates local model updates of the participants into the global federated model. The aggregation strategy is dictated by a fusion algorithm. The design of the fusion algorithm influences the bias measured in the final model. For instance, some equally incorporate the model updates from participants, while other perform a weighted average based on the participant dataset size etc. Most research on fusion algorithm focuses on improving the model accuracy and robustify the learning process. They can thus choose to ignore participant replies that are dissimilar from replies by other participants, a choice which can exclude minority groups.

It is a very difficult challenge, and unsolved to the best of our knowledge, to design a secure, robust, private and fair fusion algorithm.

Party Selection and Subsampling This challenge does not affect the cross-silo setting, but the cross-device settings, where only a small subset of the participants are chosen at each iteration to collaborate for the learning process [54]. For example, consider a scenario where each participant is a cell phone and a company wants to train a model to improve the user experience on its application. Then, if the participant selection is performed using the network connectivity, people living in the slower networks regions may be represented at disproportionately lower rates. Inclusion in the learning process is here correlated with a socioeconomic status, thus it is a systemic source of bias.

3.6.1.6 Bias Sources Overview

In Figure 2, we provide an overview of possible bias sources and where Bias can occur in a ML/DL setting. The different steps are:

- Real-life phenomenon: Phenomenon that we try to model;
- Business knowledge: Skills, knowledge, experiences, capabilities, insight about the phenomenon that we want to study;
- Experimental protocol: Translation of the business knowledge in terms of methodologies that we will follow to model the phenomenon (how many instances we need to have significant results, how we validate results, etc.);
- Data collection: Data gathering from the real life phenomenon according the experimental protocol;
- Data Understanding: Understanding of the data, for instance through a descriptive analysis, a data quality assessment, etc.;
- Data preparation: Feature engineering;
- Modeling: Modeling of the phenomenon with statistical, Machine Learning, Deep Learning, physical, etc. models;
- Evaluation: Verification of model accuracy, the respect of the model's assumptions, etc.;
- Result valorization: Extraction of important information from the modeling, plots, code packaging, etc.;
- Model deployment: Model and code deployment.

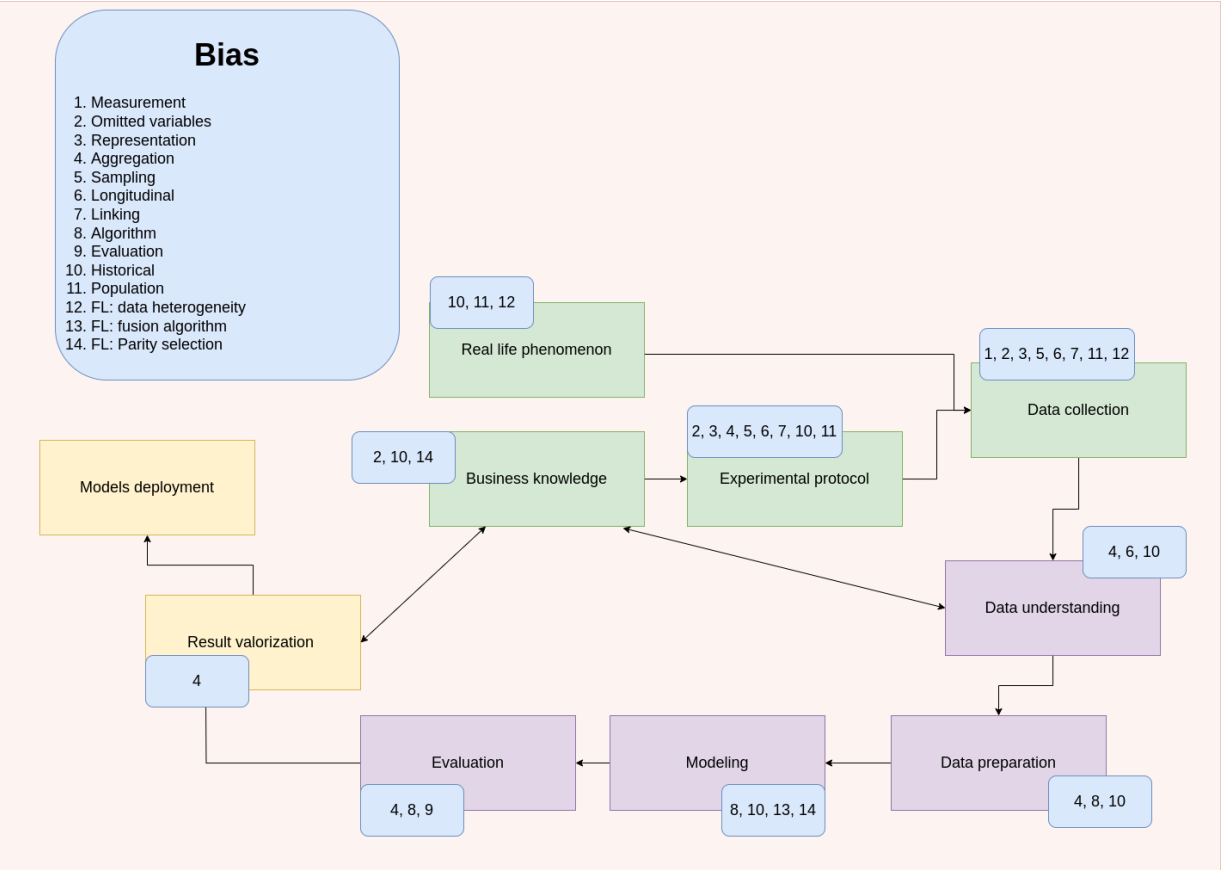

Figure 2 – Bias sources in ML/DL settings.

3.6.2 Bias measurement

3.6.2.1 Discrimination metrics

This paragraph of the deliverable, we focus on how to quantify bias of a given dataset by defining and analyzing the metrics to be adopte for bias assessment.To simplify the analysis, we use as an example a binary classification task with binary sensitive dataset attributes. There is no difficulty in extending these metrics to more complex cases (for example, a multi-label classification task, with a set of categorical sensitive features) at the cost of more complex equations. In this paragraph, we only aim to give the reader an understanding of metrics, following the IBM tutorial on fairness [106]. In the following deliverables (WP2 and WP3 related deliverables), we will describe these metrics more thoroughly and specifically to the SECURED use-case needs.

Group-discrimination metrics Group metrics aim to quantify how similar or different are the outputs of two distinct groups of individuals who differ by their sensitive attribute.

Base rate metrics

Initially two base rate metrics are presented, that rely on the predicted outcome:

- **Disparate impact**, that compares the percentage of favorable outcomes for a monitored group to the percentage of favorable outcomes for a reference group. The ratio value should be close to 1, the closer it is to 1, the fairer the model. If the value is below 1, then the privileged group has a benefit, if the value is above 1, then there is a benefit for the unprivileged group.

- **Statistical-parity difference, also called demographic parity**, it calculates the difference in the ratio of favorable outcomes between monitored groups and reference groups. The ideal value for this metric is

0; if it is below 0, it implies a benefit for the privileged group, if it is above 0, it implies a benefit for the unprivileged group.

Group-accuracy and group-calibration metrics

We now present two other group metrics that are based on the predicted and actual outcomes.

- **Equal-opportunity difference**, it calculates the difference of true positive rates between the unprivileged and the privileged groups. By true positive rates we mean the ratio of the number of true positives on the number of actual positives for a given group. The ideal value for this metric is 0, if it is below it implies higher benefit for the privileged group, if it is above it implies higher benefit for the unprivileged group.
- **Equalized odds**, its goal is to ensure a ML model performs equally well for different groups. It is stricter than statistical parity because it requires that the machine learning model's predictions are not only independent of sensitive group membership, but that groups have the same false positive rates and true positive rates.
- **Predictive rate parity**, based on the idea that the true label should be independent of the sensitive attribute conditional of the model prediction. It is equivalent to satisfying both the positive predictive parity and the negative predictive parity that respectively focus positive and negative true label. The positive predictive value being the ratio of the number of true positive to the number of points that are labeled as positive by the classifier in that same group. A classifier that respects the positive predictive parity is said to be **well-calibrated**.

Individual-discrimination metrics Individual-level discrimination measures how the model handles one individual comparing to the most similar individuals. It was first proposed by Cynthia Dwork et al. in 2012 in [107].

Impossibility theorem Impossibility Theorem [108] states that no more than one of the three fairness metrics of statistical parity, predictive parity and equalized odds can hold at the same time for a well-calibrated classifier and a sensitive attribute capable of introducing machine bias. It becomes possible in two special cases; when the prevalence of the outcome being predicted is equal across groups, or when a perfectly accurate predictor is used.

An example of this impossibility theorem, is the Correctional Offender Management Profiling for Alternative Sanctions (COMPAS) tool and related dataset [7] developed by NorthPointe that was eventually evaluated by ProPublica, an online news source organization [109]. In this dataset it was found that the COMPAS scores predicting the likelihood of a defendant committing a crime in the future, were biased against black defendants when compared to white defendants. Their work is based on the false positive rate (that is part of the equalized odds metric) and the false negative rate. On the contrary, the creators of the COMPAS scores, NorthPointe, justify their work by explaining that the have focused on the calibration of the model. In the course "Fairness and Algorithms" [110] by Atri Rudra of University of Buffalo, it is explained in detail that according to the impossibility theorem, ProPublica and NorthPointe have both correct analysis using classical fairness metrics even though their statements are contradictory.

However, a recent study [111] has shown that by relaxing the conditions, meaning by allowing a small margin-of-error between metrics, it becomes possible for a model to satisfy in this margin the different Bias/Fairness metrics described in this subsection simultaneously.

[7] https://s3.documentcloud.org/documents/2840784/Practitioner-s-Guide-to-COMPAS-Core.pdf

Others bias measurements Counterfactual fairness was introduced by Russel in 2017 [112]. It provides a possible way of interpreting the causes of bias. Counterfactual fairness gives us a way to check the possible impact of replacing the sensitive attribute (i.e. attribute affected by biasing) only. It provides a means of explaining the impact of bias via a causal graph.

A recent fairness metric similar to differential privacy has been introduced by [113]; it represents a more generic definition of fairness.

Bias measurement in Federated Learning context In a Federated Learning context it is not possible to access the whole training dataset. Thus, the different bias metrics can be applied either locally by each participant or globally but on a test dataset.

3.6.3 Bias mitigation

In Figure 3, the generic pipeline on a ML/DL setup in order to achieve fairness (i.e reduce bias) is presented [114]. An example instantiation of this generic pipeline consists of loading data into a dataset object, transforming it into a fairer dataset using a fair pre-processing algorithm, learning a classifier from this transformed dataset, and obtaining predictions from this classifier. Metrics can be calculated on the original, transformed, and predicted datasets as well as between the transformed and predicted datasets. Many other instantiations are also possible. In the following subsection, we provide more details on existing approaches loosely following the Figure 3 pipeline.

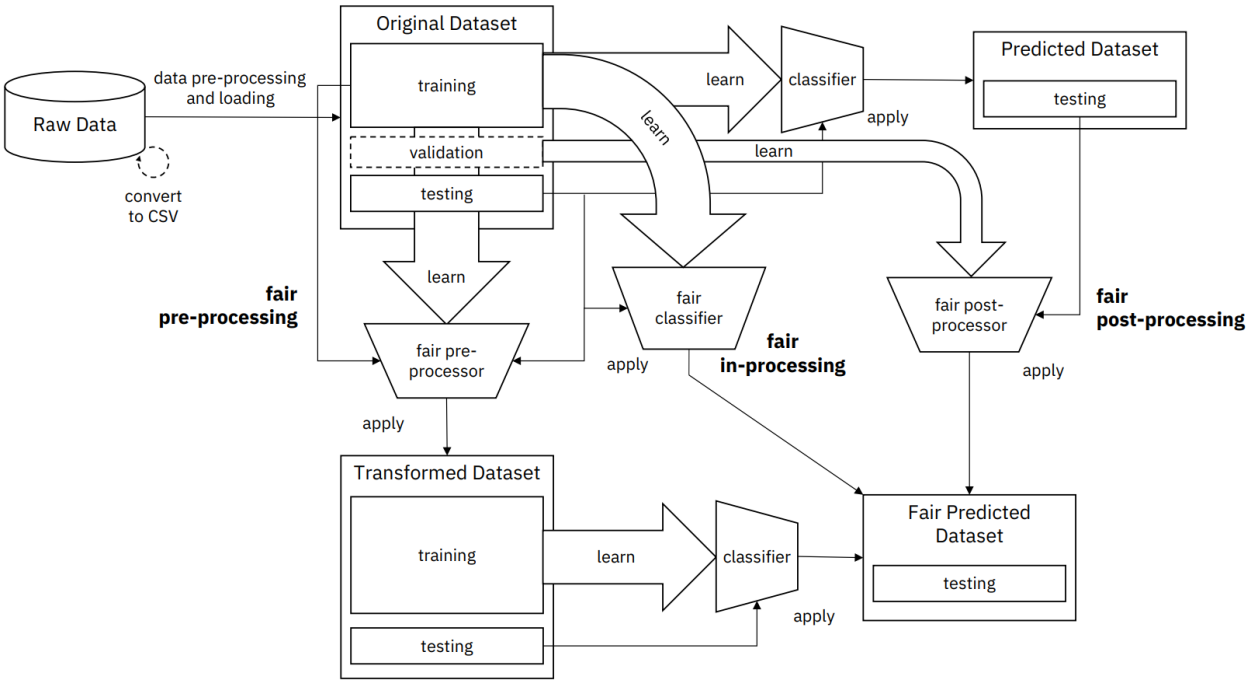

Figure 3 – The fairness pipeline as designed by AIF360 [114].

3.6.3.1 In classical approaches

Pre-processing During Pre-processing the objective is to enhance fairness of models by rectifying training data. Several methods to achieve this are associated with reweighing the training dataset. With these methods, we try to generate weights for the training sample to ensure fairness before ML models. Thus, we only change the weights of each instance, and do not change the features and the label of the instances. The weights are chosen to mitigate discrimination and aim to satisfy some fairness metrics such as statistical parity (see [115]). [115] proposes to reweigh the dataset by giving more weight to instance with discriminated sensitives attributes whose outcome is positive. Other approaches to reweigh the dataset are based on Transfer Learning. Using Transfer Learning we have a biased labeled training dataset and an unlabeled and unbiased test dataset acting as a sample bias. For example, in a survey, patients of a given age class can be over-represented in comparison to the real patients distribution. Here, the training dataset is biased according to the age distribution and models (and statistics) extracted from this training dataset can reproduce and amplify bias according the age. They cannot be well generalized when they apply to the test dataset. In the Transfer Learning setting, the sample bias can be described as an unsupervised domain adaptation problem. To correct this sample bias, we can use approaches like Kullback–Leibler Importance Estimation Procedure (KLIEP) [116] or Kernel Mean Matching [117]. These approaches try to find an optimal weighing of the training dataset to match the feature distribution of the test set. The finding of optimal weights is depended on specific criteria per approach, e.g. for KLIEP, the algorithm finds the weighing that minimizes the divergence of Kullback-Leibler between the training and test datasets features distribution. On the other hand, instead of using a weighing dataset, in [115] it is proposed to sample the dataset with proper data replacements using the optimal weights.

In [115] some others approaches are proposed that potentially can lead to the strong degradation of a dataset. For example, one of these methods rely on finding the most correlated features within the sensitive features and remove both sensitive features and these most correlated features. This method does not seem relevant to SECURED because, among others issues, there are no assurance that bias is indeed removed using it. Previous experiments showed that it is often more interesting to keep the sensitive attributes in the dataset and to correct their bias effects instead of removing them. Also, the definition of the correlation itself and the way it is calculated can complicate the bias removal process (depending on the correlation type eg. a linear correlation calculated with the Pearson coefficient or a monotone correlation calculated with the Spearman rho, or the rank correlation measured by the Kendall tau ? etc.). Another approach that is proposed in[115] consists of changing

the labels of some individual people in the dataset but by doing so, uncontrollable bias is introduced in the data.

Feldman et al. in [118] deal with disparate impact discrimination, they link the disparate impact with the balanced error rate and show that any outputs with disparate impact can be converted into one where the sensitive attributes leaks can be predicted with a low balanced rate error. Knowing this, they propose an algorithm to force the distribution of the sensitive attributes to be close across all protected groups.

Several authors like those in [119], [120], [121] propose to provide a fair representation of the dataset. Theses approaches aim to find a latent representation that encodes the data while keeping the information of the sensitive attributes hidden so that any association between sensitive attributes and the labels can be removed. Also, adversarial networks have been used in a various ways to generate fair datasets that can be used to train a model [122].

Finally, in [123] a probabilistic formulation of data processing for reducing discrimination according three dimensions is proposed. This is based on controlling discrimination, limiting distortion in individual data samples, and preserving data utility.

In-processing During In-processing we customize ML/DL algorithms to directly train fair models. Adversarial network are used to achieve such objective. Louppe et al. in [124] propose an adversarial network to incorporate systematic uncertainties. They focus on a problem where we want to learn a function that matches a dataset with labels in presence of some nuisance parameters. For that, they use an adversarial process with a first network whose objective consists in predicting the label according dataset value. There is a second network that takes as input the prediction of the first network and predict the nuisance parameter. While the second network is efficient to predict the nuisance parameter, it penalizes the loss of the first network. The objective of this adversarial network is that at the end of the training the first network is accurate enough to predict the dataset labels while its predictions are independent of the nuisance parameters. While in [124] this adversarial network is employed to avoid nuisance parameters in examples from particle physics, this or similar adversarial processes can be applied in fairness by considering the sensitive attributes as the nuisance parameter as discussed in [125], [126], [127]. Moreover, instead of the prediction of the first network (that will allow to work on base rate metrics), we can provide the first neural network errors. The adversary game can be integrated directly during the optimization of the network, as described in [128], by detecting neurons with contradictory optimization directions from accuracy and fairness training goals, and achieving a trade-off by selective dropout.

Another family of approaches rely on adding a penalization according to the direct and indirect effects of the sensitive attributes on the outcome, like e.g. in [129] and [130] where that modify the classical Random Forest algorithm [131] is modified by changing the cost function to consider both the impact of the improvement if a feature is used for a division during one tree growing and the association between the candidate feature and the sensitive attribute.

Post-processing Post-processing approaches revise the prediction scores of a machine learning model after training to make predictions fairer. To address the issue, Kamiran et al. in [132] propose the Reject-Option based Classification method. In this approach, they authors of [132] propose to add a reject option when the prediction made by the model is too close to the classification boundary. In this case, the consider that the prediction made for the given instance is uncertain and potentially due to bias. To reduce discrimination, these rejected instances are labeled either as belonging to the discriminated category or not (labeled in the positive class if they are in the discriminated category, in the negative class otherwise). In [133] some model predictions are randomly flipped with probabilities that depends on the original prediction and the sensitive attributes values. To improve the flipping a randomized threshold optimizer as the one proposed in [134] can be used.

3.6.3.2 In Federated Learning approaches

Most of the methods described in Section 3.6.3.1 assume that the totality of the training dataset is available, which is not compatible with data privacy. In a Federated Learning context the different metrics cannot be obtained for the whole training dataset but only in each subgroup handled by each FL client, assuming each participant agrees to share their metrics.

Furthermore, recently it has been shown in [135] that biased parties can unintentionally encode their bias in a small number of model parameters, and throughout the training, they steadily increase the dependence of the global model on sensitive attributes. The bias of a few parties can propagate through the training and affect the federated model more than on a centralised model. Thus, pre-processing, in-processing and post-processing in a FL context need to be redesigned to address the above concerns.

Pre-processing Reweighing is a centralized pre-processing bias mitigation method that attach weight to samples in the training dataset. In a FL setup, reweighting can be applied locally by each participant or globally, but this implies that participants agree to share information about the distribution of theit local data. In [136] the authors show that the local reweighing can significantly reduce the model bias, even under highly imbalanced data distributions.

Another approach that can be seen as a pre-preprocessing approach is proposed in [137]. Its aim is to train a PrivGan [138] in a FL setup to then generate synthetic dataset locally by each participant. They show that the synthetic data generated is less biased than the real dataset without reducing the usability of the dataset for FL.

In-processing AgnosticFair [139], FedFair [140] and [136], are different approaches that try to mitigate bias by introducing a fairness constraint into the global loss function.

In [141] the authors propose three different weighted-aggregation techniques in FL to mitigate the bias, based on the fairness metrics of each client's model updates. These metrics are evaluated on a test dataset, thus the reliability of this approach rests on the representativeness of the test dataset.

Besides, the Astraea framework [142] reschedules the training of the clients on the server side, to make the distribution of the collection of data close to the uniform distribution. The Astraea drawback is that it is based on the assumption that each client must share information about the distribution of its local data.

Post-processing To the best of our knowledge there is not yet published work on post-processing bias mitigation in Federated Learning. This statement is confirmed by a 2021 survey on bias mitigation for FL [143]. As explained in [143], this type of approach may be adequate for the constraints of FL, since it acts on the final available federated model, and considers the ML model and the data as a black-box.

3.6.4 Related State-of-the-Art Gaps

Finally, based on the section analysis, in Table 5 some preliminary State-of-the-Art (SoTA) Gaps have been identified. Note that in the Table, we also provide the SECURED flow associated with the gap (i.e Data, Processing or both flows) and the related identified components of the preliminary SECURED Architecture described in Subsection 7.2.

Challenge Gap ID	Description	Flows	Related SECURED Component(s)
SoTA-GAP-15	There are no studies to evaluate Differential Privacy with data labelling attacks on medical data	Data	Privacy Preserving AI-trained model Marketplace (ML/DL/FL modules)

SoTA-GAP-16	The Post-processing of bias mitigation on a FL context has not be substantially studied	Data	Bias Assessment & Unbiasing services and tools

Table 5 – PET ML/DL/FL main State-Of-the-Art Gaps

4 Secure Multi-Party Computation (SMPC) and Homomorphic Encryption (HE) for ML/DL Health Applications

4.1 The necessity for Privacy-Enhancing Technologies (PETs) in ML/DL Training and Inference

This section explores how Privacy-Enhancing Technologies (PETs), and in particular Secure Multi-Party Computation (SMPC) and Homomorphic Encryption (HE), can be used to provide enhanced security properties to ML/DL where the input(s) and the learning model are held by different parties[8]. SMPC allows a group of parties to jointly perform some computation without revealing any of the parties' inputs, while HE allows an evaluator to compute some function on encrypted data.

We focus on PETs for health applications and more specifically our focus will be on adding privacy protection to the computations that are involved in ML: a model is trained by one or more parties, and then later accessed for inference queries by one or more parties (possibly a completely different set from the training parties). Some operations performed in the training phase and the inference phase can often differ significantly, and the security and privacy properties required are very different in each case so it is important to distinguish, where necessary. The descriptions herein will begin using the simple two-party case where a client provides some input and a server wants to acquire or holds some DL model, and from there extend to multiple parties on each side of the computation, i.e., client or server. Much of this section is focused on feed-forward Neural Network (NN) approaches, however many of the techniques apply to other learning models such as linear/logistic regression: any important distinctions are emphasized.

For the training phase, a data set is provided as input in order to 'train' the DL model, which for the purposes of this section means fixing the parameters and weights of the function (circuit) that the client(s) and server will later jointly compute in the inference phase. Ideally the interaction should leak as little as possible about the underlying data to the server, and in particular if the data comes from multiple sources then the input parties should not learn the data of the other parties, in addition to hiding as much as possible about the parameters and weights of the resulting DL model[9] from the input party (or parties). This section will use Secure Deep Learning Training (SDLT) to describe protocols that serve this purpose.

In the simplest case of the inference phase, there is a single client with an input x and a server holding a DL model that is considered as a function f, where the client wishes to learn $f(x)$. The server wants to keep as much information about f secret as it possibly can, and the client does not want to reveal anything about x other than its existence (and perhaps its length). For the remainder of this section, the term Secure Deep Learning Inference (SDLI) is used for this context[10]. This section aims to describe what can be protected when implementing, and explain the trade-offs of efficiency, accuracy, and applicability under the various levels of achievable privacy. For a more thorough survey of secure inference for NNs, see [147].

A trivial solution to both SDLT and SDLI is for a trusted third party to receive input from the client (and the model from the server in inference) and run the operation in question, returning the appropriate model/value to the parties. Essentially, the aim of applying PETs is to emulate this third party as a cryptographic protocol among the participating parties, while minimizing the trust assumptions.

For the SECURED project, it is necessary to go beyond the simple case of one client and one server, and extend to multiple parties providing data as input both for training data and in the inference phase, and multiple servers holding (parts of) the NN. Note that, for inference, it is possible to speed up computation [150] by introducing a

[8] Note that tools such as differential privacy [144] and trusted execution environments (TEEs) can be used in combination with the PETs discussed in this section to increase privacy guarantees [145], however such combinations are out of scope. Differential privacy can only be useful in the training phase for the DL model, since in the inference phase we are only working with a single record (input).

[9] One can consider a scenario where multiple servers train their own models based on data from the same group of input parties, but give e.g. additional weighting to their own input data [146]. This more general idea results in different models (circuits) at each server, and many of the techniques described in this document also cover this case, but for simplicity we describe here a single model as being the outcome of the training phase.

[10] Note that alternative terminology for almost identical concepts exists in the literature, for example Secure Neural Network Inference [147], Secure and Correct Inference [148], and Private DL Inference [149].

third party that does not participate in the inference interaction between client and server, but sends information such as correlated randomness (independent of x) to all of the parties. It is necessary to assume that this third party does not collude with (all of) the server(s), and this party can be realized using hardening techniques such as a trusted-execution environment or even a distributed protocol.

It is now possible to shift our attention to the goals that the system should achieve and under which conditions the protocol is regarded as achieving these goals. Independent of the target privacy properties, the setting needs to consider the capabilities of the adversarial parties. Most works in the literature assume that both client and server are *Honest-but-Curious (HbC)*, meaning that they will follow the protocol as instructed and attempt to infer as much information as possible[11] using the intermediate values and calculations. A (much) stronger model is that of malicious (or active) security, allowing a party to misbehave arbitrarily. Some works consider an HbC server and a malicious client, but most works in this category investigate what is possible when both client and server are regarded as malicious.

In the healthcare setting where very sensitive data is being provided as input, it is foreseeable that only malicious security will be acceptable in order to guarantee that these data items *cannot* be leaked by the protocol. This may even be enforced by privacy regulations or other organisational policies regarding consent for data sharing and analytics. Malicious security comes in two flavors: *malicious-security-with-abort* ensures that if a party misbehaves then the other parties will, with high probability, abort rather than provide output. The second variant, often called *robust security*, offers stronger security guarantees and allows the honest players to continue with the protocol. Robust security requires an honest majority, while malicious-security-with-abort can be achieved even with $n-1$ dishonest parties (where n is the total number of involved parties). Additional techniques and machinery are required to achieve malicious security, making protocols in this setting less efficient than in the HbC setting. For this reason it is essential to specify early on in the planning process which functions need to be computed, for training and/or inference, and assess feasibility of maliciously-secure protocols for these functions.

In the training phase, it is possible to aim for the following secrecy goals:

T1. The input training data is not leaked to the server that constructs the DL model, since it is encrypted (beyond what can be learned by querying the inference phase), and
T2. Internal values of the DL model (i.e., weights and biases) are not leaked to the client.

In the case where multiple clients provide training data as input, then an additional property is desirable:

T3. The input training data of each client is not leaked to the other input clients.

For the inference phase the following secrecy goals are of interest:

I1. The input to the inference is not leaked to the server(s),
I2. The output of the inference is not leaked to the server(s),
I3. Internal values of the DL model (i.e., weights and biases) are not leaked to the client.

Most approaches aim to achieve these three goals simultaneously. Some works attempt to provide an additional property:

I4. The architecture of the DL model (e.g. number, types and sizes of layers of the NN) is not leaked to the client.

Independent of any techniques used to provide SDLT/SDLI, it may be possible for a server to infer information about the training data or for the client to infer information about the model simply by observing input-output pairs (for example via a model extraction attack [88, 151], a model inversion attack [85], or a membership inference attack [86, 152]), making T1 and I4 difficult to formalize and realize in practice; see Section 3.3.3 for a discussion of these attacks.

Any approach realizing SDLI needs to be either interactive, where parties communicate in phases corresponding

[11] All definitions of security for SMPC are with respect to information that is not trivially leaked by the protocol. Consider a multi-party average protocol: each party provides an integer as input and at the end all parties learn the average of their inputs. The two-party version of this protocol is not even a candidate to be performed using SMPC, since the output and one input trivially reveals the other input. In a privacy-preserving auction, the bids that do not win are usually kept secret but it is expected that the winning bid value would be revealed. Our focus on deep learning which will mostly allow us to sidestep these discussions.

to each layer of the NN, or non-interactive, where communication only happens at the beginning and end of the protocol. Clearly, the interactive approaches reveal the number and type of layers in the NN which violates security property I4 above, and additionally the potentially large number of rounds of communication may be problematic in some applications. On the other hand, doing the layers stepwise can enable significant speedups in computation time as special-purpose functionalities, can be used in these layers; for example, the non-linear activations can be replaced with polynomial approximations. Choosing which approach is better is clearly application-specific.

Many approaches to SDLI (and indeed SMPC in general) employ an "offline" pre-processing phase that is independent of the inputs of the parties, with the goal of speeding up the inference requests in the "online" phase [153, 154, 150, 155, 156]. This is only feasible if the client and server participation patterns are known, and idle time can be efficiently allocated to performing the pre-processing. Again the choice to use this approach is application-specific.

Before specific techniques for SDLT and SDLI can be introduced, it is prudent to briefly describe some technical challenges that are present throughout the rest of this section. The first is data representation and efficient conversion: training data and DL weights are often decimal values while the SMPC approaches operate on some ring [12], usually $\mathbb{Z}_{2^k}$. This requires efficient techniques for acquiring values in the correct representation, and much work has covered this challenge in the two-party [157] and the three-party setting [158]. Performing operations like stochastic gradient descent requires careful selection of k to define the ring, to ensure that accuracy is not sacrificed. In the inference phase it will be necessary for the parties to jointly compute some function f and the efficiency will be decided by the (multiplicative) circuit depth of f. This means that techniques are needed for representing the non-arithmetic operations such as non-linear activation functions and piecewise polynomial functions in a way that can be handled by the SMPC/HE protocol while again wanting to retain accuracy. Some of the software tools that are detailed later in this section only support a subset of popular activation functions. As a general rule, techniques for SDLI based solely on HE will usually not support non-linear activation functions and require the client to perform them in the clear (this is not always the case, e.g. [159]), while techniques solely built on SMPC (with a single server) will be able to support these functions but will leak the model architecture: this trade-off can be avoided by using a hybrid model that incorporates both approaches [160, 155, 149]. Past literature can be split into frameworks that cover both training and inference [158, 156], inference protocols that assume a pre-trained model [161], and those which transform a pre-trained model to make it more appropriate for SDLI [162, 163, 164, 155].

A fundamental building block for many SMPC-based protocols is Oblivious Transfer (OT) [165]. In the simplest case, party A provides two messages m_0, m_1 as input, party B provides a selection bit b, and at the end of the protocol B learns m_b while A learns nothing. This can be generalized to more than two messages, and Oblivious Transfer Extension (OTe) [166] enables a large batch of OTs to be performed at a cost far cheaper than individual computation. Another important variant is Random Oblivious Transfer (ROT) [167], where the selection bit b is not provided as input but is generated at random by the protocol: the output of ROT is two correlated pairs of bits (m_0, m_1) and (b, m_b). Given a ROT instance the two parties can compute OT using just three bits of communication.

Note that in some healthcare applications, a prediction algorithm is trained on private data and then the resulting algorithm is made public as part of a public health initiative [168]. In this case, SMPC/HE techniques are only of assistance in the training phase, and can be used to combine multiple input data sets. For the inference phase, a user can input data using a web/app interface[13] and retrieve their result without any data being transmitted over the internet. If this user wishes to keep their input (potentially containing sensitive details about their own health situation) secret from their web browser, they could run the public algorithm locally on their own machine if they have the requisite technical knowledge.

[12] A ring is a set that fully supports two binary operations satisfying properties analogous to those of addition and multiplication of integers.
[13] For example, the QRisk3 predictor for heart attacks and strokes, trained on 35 million UK NHS patients: `https://qrisk.org/`.

4.2 Techniques for Privacy Enhancing

Three techniques with different features will be detailed in the following; their most important aspects are summarized in Table 6. For the number of computing parties, the client may or may not participate as a computing party in the secret sharing and garbled circuit contexts. As we will see later, given these various features and the complexity of the functions present in DL applications, it is nearly always preferable to use a combination of techniques for increased efficiency.

	Garbled Circuits (4.2.1)	Secret Sharing (4.2.2)	Homomorphic Encryption (4.2.3)
Circuit Representation	boolean	arithmetic ($\mathbb{F}_p$)	polynomial
# Computing Parties	$n \geq 2$	$n \geq 2$	1 server
Round Complexity	constant	circuit depth	1
Communication Complexity	256 bits per AND gate ($n = 2$)	$2n \log p$ bits per mult. gate	circuit independent

Table 6 – Comparison of PETs for secure computation of a single function between one client holding an input and one or more servers holding the function.

4.2.1 Garbled-Circuit Approaches

The first class of protocols is SMPC based on circuit garbling, as introduced by Yao [169]. For any efficiently computable function that can be expressed as a Boolean circuit of only AND and XOR gates, this function can be transformed into a Garbled Circuit (GC) that the parties in the protocol jointly compute to minimize the leakage of inputs. In the SDLI setting this means that the structure of the NN is known to both parties (and thus goal I4 from above is not met): the client's secret input is its inference input and the server's input is the set of weights and parameters for the NN: this allows the encoding of the entire NN as a Boolean circuit when the bitlength of all input values is fixed. Note however that this is not the only way to use GCs in SDLI: when using a mixed-protocol approach that uses GCs and other tools for each part of the model, for non-linear layers it is possible to reverse the roles [155].

The number of communication rounds is constant and independent of the function being computed, and follows two phases: an input-independent circuit garbling phase allows the parties to construct the garbled circuit, and the evaluation phase. In this second phase the gates of the circuit can be computed in parallel, and in particular for XOR gates no communication is required: using GCs in SDLI needs to consider the multiplicative size of the circuit only.

While Yao only considered two parties, this was extended to the multi-party setting by Beaver, Micali and Rogaway [170]. The intuition here is that the computing parties perform a distributed generation of the garbled circuit, so that no single party (or acceptable size subset of the parties) knows the label assignment, and this distribution generation can be done in parallel for all circuit gates. The first demonstration of the feasibility of using Yao's GCs was given in the Fairplay system [171].

A crucial component of GC approaches is OT. Intuitively, the garbler will prepare a representation of each of the Boolean circuit's gates for all possibilities, and the evaluator will then traverse the gates and 'blindly' evaluate according to permuted truth tables. Here, OT comes in: the garbler acts as the sender of the (encrypted) wire labels corresponding to 0 and 1. Another major efficiency improvement can be made by using a fixed-key blockcipher during the gate garbling process [172].

The computational costs for both the garbler and evaluator are dominated by calls to the encryption function, which is usually an established cipher like Advanced Encryption Standard (AES) in some pre-agreed mode of operation. As we will see in Section 4.5.2, it is often beneficial to use symmetric encryption schemes that are designed to have low-depth circuit representation instead of AES: see Section 4.5.2 for a more detailed discussion of this issue. For SDLI the circuit representation of the model may be enormous, so automated tools can be used to optimize (each part of) the circuit to minimize the number of non-XOR gates [173, 163]. When treating the DL model as a combination of components, GCs are particularly useful for activation functions such as ReLU, MaxPool and (an approximation of) Sigmoid. This means that mixed protocol approaches may only use GCs for activation functions, and secret sharing or homomorphic encryption for other operations.

4.2.2 Secret-Sharing Approaches

In Additive Secret Sharing (A-SS), a party can distribute a secret value in two or more shares, where each individual share reveals no information. To reconstruct, all or some sufficient number (threshold) of the shares are needed. This idea can be used to compute boolean or arithmetic circuits, where the computing parties simply process the circuit gate-by-gate with their blinded inputs. Similarly to the GC approach, ADD gates (or XOR in $\mathbb{F}_2$) can be done locally: both parties apply the operation to their own shares. An important enabler of the A-SS technique is doing non-linear gates (multiplication in larger rings) with the help of either random oblivious transfer (correlations) for boolean circuits, or multiplication triples (so-called Beaver triples) [174] which are shares of values a, b, c such that $c = a \cdot b$ for arithmetic circuits[14]. Since each triple acts as a one-time pad for each multiplication gate, they are "used up" in the online evaluation of a gate and a sufficient number of triples must be produced in advance. This allows shares of multiplication inputs to be produced in the input-independent pre-processing ("offline") phase at a time that is convenient to both parties, giving a reduced online communication cost. Finally, after all the gates are computed, the server party in two-party SDLI will transmit its output share to the client (the parties would exchange their output shares if both parties should get output).

Many modern approaches that use A-SS employ the GMW protocol [175] as the core framework, and impart many optimizations and improvements. An alternative to A-SS is a class of protocols with information-theoretic security, avoiding public-key primitives such as OT (and their computational assumptions) altogether: BGW [176] forms the basis for many such schemes. BGW-like schemes require an honest majority of participants and usually do not involve a pre-computation phase. Note that ROT can also be used to efficiently generate Beaver triples [177], while an alternative approach for generation is use of HE. In order to achieve malicious security, the parties require Beaver triples where shares can be opened reliably, and modern works generally make use of Message Authentication Codes (MACs) that operate in the appropriate field with certain special properties. BDOZ [178] and the TinyOT improvement [179] (that uses OTe) use one-time information-theoretic MACs to generate authenticated bits, then use an interactive protocol to turn these into Beaver triples: the local storage scales linearly with the number of parties. A way to avoid this linear scaling was introduced in the SPDZ protocol [180], and it was later shown how to improve the triple generation using OTe [181]. ISO/IEC are in the process of producing a standards document for secret sharing-based SMPC [182].

Schemes that use A-SS can support an arbitrary number of parties, but for the SDLI setting it might be difficult to find three or more mutually distrusting computing parties, leaving two servers and thus ruling out use of BGW-like schemes that require an honest majority. In general it could be that the client is one of the $n \geq 2$ computing parties, however this would reveal the weights and parameters of the model (i.e., leaking much more than just the model architecture) to the client. A more common usage scenario is for n non-colluding servers to hold the joint model, and for the client to share their input to those servers: this has the added benefit of low computational and communication complexity for the client.

For SDLI, additions can be done locally on shares so the parties only interact to compute for multiplications in the linear layers. In each of these non-linear layers, the communication cost is proportional to the number of multiplications being performed but no generic and efficient technique is known: many different approaches have been used in the literature with different trade-offs.

[14] To see how this works, if two or more parties want to multiply secret-shared values x and y, they each reveal their own shares of $x - a$ and $y - b$ and then compute (their share) $[x \cdot y] = (y - b) \cdot [a] + (x - a) \cdot [b] + [c] + (x - a) \cdot (y - b)$.

4.2.3 Homomorphic-Encryption Approaches

Homomorphic Encryption (HE) is a category of encryption schemes that allows certain functions to be evaluated on data while it is still in its encrypted state. This allows a user to encrypt their plaintext data, send the ciphertexts to some untrusted server, and then query the server for functions[15] of the encrypted data, all in a way that leaks nothing about the plaintexts to the cloud. To illustrate, given a homomorphic encryption procedure Enc that supports addition, $\mathsf{Enc}(2) \odot \mathsf{Enc}(5)$ would yield $\mathsf{Enc}(7)$, for some operation $\odot$ which would be the equivalent of the plaintext addition in the ciphertext space. Unlike traditional encryption schemes, which do not give any guarantee about the meaning of the sum of ciphertexts, HE promises, amongst others, to perform ML—most often inference (SDLI), but training (SDLT) is also possible—directly on the encrypted data. To emphasize, in this situation the ML model owner does not need to decrypt the encrypted data it is fed before performing inference or training: confidentiality of the user's data is preserved even if the server behaves maliciously[16]. A more formal treatment of the syntax of HE is not necessary for the purposes of this report, but can be found in the Homomorphic Encryption Standard document [183], a community-led effort to define schemes and security properties in the HE space.

HE schemes, of which the most popular ones are given in Table 7, mainly fall under one of the following three branches [184, 185]. Firstly, Partially Homomorphic Encryption (PHE) schemes only allow one type of function (i.e. gate type) to be evaluated, albeit an arbitrary number of times[17]. In practice, the function is either addition or multiplication. PHE may be sufficient in several application scenarios. One such example is e-voting, where the tallying (counting) of votes normally only requires addition to be supported, as shown by schemes that rely on PHE for security and anonymity [187, 188]. Other examples include similarity testing [189], and membership queries [190]. Secondly, Somewhat Homomorphic Encryption (SWHE) schemes allow more types of functions to be supported, and might for instance also allow branching on the encrypted data, but the number of evaluations is bounded. Applications include secure pattern-matching [191] private database querying [192], and steganography [193]. Finally, the holy grail is promised by Fully Homomorphic Encryption (FHE) schemes that allow any arbitrary function to be evaluated for any arbitrary number of times. FHE has proven, for example, to be capable of securing complex data-centric applications, including databases [194]. In the remainder we will purely focus on FHE, as the complexity of ML algorithms demands this.

In general, an FHE scheme, of which the first one was lattice-based and proposed by Gentry [204], depends on the addition of randomness—called *noise* or *error*—to the ciphertext. Each HE evaluation also adds noise to the ciphertext (growing slowly for addition, and very quickly for multiplication). However, there is an upper bound on the amount of noise tolerated. If that bound is exceeded, decryption will result in failure. Gentry's breakthrough scheme [204] consisted of the following generic blueprint:

1. Start with a Somewhat Homomorphic Encryption (SWHE) scheme that can evaluate low-degree multivariate polynomials homomorphically,
2. "Squash" the decryption circuit of the scheme, by transforming into a new scheme with equivalent homomorphic capabilities with a simpler decryption circuit,
3. Add a *bootstrapping* procedure that "refreshes" a ciphertext by homomorphically evaluating the decryption function on it with an encrypted secret key.

The result of this procedure is a Leveled Fully Homomorphic Encryption (LFHE) scheme where the parameters can depend on the depth of the circuits that the scheme can evaluate, but not the size of these circuits. To get from here to a fully-fledged FHE scheme it is necessary to introduce an additional assumption of *circular*

[15] These functions could be i) known to both user and server, ii) a secret of the server that is not known to the user, for example a proprietary machine learning model, or iii) a secret of the user that is not known to the server, in which case the function must be sent in encrypted form to the server by the user.

[16] Note that a server could attempt to perform a different function on the ciphertexts than the one intended by the user, in order to save computation steps or just to manipulate the result. Avoiding this issue is possible (along with the result, the server sends a proof that in essence binds the function to the result) however a discussion is beyond the scope of this document: in general we will assume that the server is essentially Honest-but-Curious (HbC).

[17] We demonstrate how additively homomorphic encryption works by example, using the Paillier encryption scheme [186]. To encrypt a message m the algorithm is $\mathsf{Enc}(m) = g^m r^N \bmod N^2$ where g is a generator, r is a randomly chosen integer in $\{0, \ldots, N-1\}$ and N (the public key) is a product of two large primes. The operator $\odot$ here that can be applied to ciphertexts is multiplication modulo N^2, and using our example from earlier: $\mathsf{Enc}(2) \cdot \mathsf{Enc}(5) = g^2 r_1^N g^5 r_2^N \bmod N^2 = g^{(2+5)}(r_1 r_2)^N$, which is an encryption of 7 under randomness $r_1 r_2$.

Scheme	Year	Key features	Security
ElGamal [195]	1984	multiplication	Discrete logarithm
Paillier [186]	1999	addition	Composite residuosity
DGHV [196]	2010	addition, multiplication	LWE
CMNT [197]	2011	addition, multiplication	LWE
BGV [198]	2011	addition, multiplication, bootstrap	LWE
BFV [199, 200]	2012	addition, multiplication, bootstrap	LWE
GSW [201]	2013	addition, multiplication, bootstrap	LWE
TFHE [202]	2017	addition, multiplication, bootstrap	TLWE
CKKS [203]	2017	addition, multiplication, bootstrap, $\mathbb{R}$	RLWE

Table 7 – Widely used homomorphic encryption schemes. Note that Fan and Vercauteren [199] gave a ring variant of the scale-invariant LWE scheme proposed by Brakerski [200] and consequently this version is commonly referred to as the BFV scheme.

security: encrypting the LFHE secret key under its own public key does not leak any information about the secret key. In general, bootstrapping is a very expensive procedure since it requires homomorphic evaluation of what is often a complex function.

Later works demonstrated how LFHE schemes can be designed directly, without bootstrapping, which has seen utility given that in many cases it is straightforward to bound the depth of the circuits that are expected to be executed. BGV, BFV and CKKS [203] are examples of LFHE schemes.

A separate line of work has looked to work in particular domains where bootstrapping can be made more efficient. TFHE [202, 205] and FHEW [206] are examples of FHE schemes supporting bootstrapping.

4.3 Existing Software Libraries/tools for PETs

4.3.1 Libraries for FHE

As academic and commercial interest in FHE has grown over the years, many open source libraries have been developed and released, such as Microsoft's `SEAL` [207], IBM's `HElib` [208], IARPA's `PALISADE` [209] which is now `OpenFHE`, etc. The libraries support variants of HE schemes that are have become established in the literature, and whose security has not been broken in over a decade of FHE literature. Some of the HE libraries also allow exponentiation, square, signing, and therefore subtraction. We also indicate which libraries support Single-Instruction-Multiple-Data (SIMD) operations. We briefly discuss the main libraries below, and we summarize them in Table 8.

Library	Developed By	Schemes	Operations	Languages
`SEAL`	Microsoft	BFV, BGV, CKKS	Addition, Multiplication, Rotation, Matrix, SIMD	C++
`HElib`	IBM	BFV, BGV, CKKS	Addition, Multiplication, SIMD	C++, Python
`PALISADE`	IARPA	BGV, BFV	Addition, Multiplication, Rotation, Comparison, SIMD	C++, Python, Java
`TenSEAL`	OpenMined	`SEAL`-*based*	PyTorch Integration	Python
`OpenFHE`	Duality Tech	`PALISADE`-*based*	Addition, Multiplication, Comparison, Inner Product, SIMD	C++, Python
`TFHE`	*Various*	TFHE	Addition, Multiplication	C++, Python
`TFHE-rs`	Zama	TFHE	Addition, Multiplication	C++, Python, Rust

Table 8 – Comparison of HE Open Source Libraries

The many homomorphic encryption libraries available offer different features, implementations, and supported

functionalities. The libraries differ in terms of their development teams, supported encryption schemes, supported operations, optimization techniques, and ease of use. No clear winner or de facto standard has so far emerged, and the libraries have to be evaluated depending on the specific application requirements and use case, to choose the one that best fits the requirements.

`SEAL` [207] (Simple Encrypted Arithmetic Library) is a MIT-licensed library that supports a wide range of mathematical operations, including addition, multiplication, rotation, and more developed by Microsoft Research. It offers optimized performance with SIMD and serialization, and is designed to work well with large-scale computations, while enabling efficient computation on large sets of data. It currently supports the BFV, BGV and CKKS schemes. `SEAL` also provides a list of secure parameter sets that have been put to test in the LWE estimator library from Albrecht et al. [210].

`HElib` [208] is a library developed by IBM, and released under Apache License v2.0. It offers powerful functionalities for homomorphic operations such as addition, multiplication, and the GHS bootstrapping algorithm. While most of its features resemble `SEAL`, it is aimed at providing a simpler in construction. The performace of SEAL and HElib has been compared in the literature, showing that each library may perform better than the other depending on the parameters [211].

`PALISADE` [209] is a library developed with the support of IARPA, released under the 2-clause BSD open-source license. It supports both HE and FHE schemes and provides multiple encryption schemes, including BGV and BFV. `OpenFHE` [212] is an open-source library created on the basis of `PALISADE`, merged with some capabilities of `HElib` and `HEAAN` [203] by CryptoLab, with the aim of making homomorphic encryption more accessible to a broader audience by categorizing basic to advanced FHE and lattice-based implementations. `OpenFHE` offers functionalities for basic arithmetic operations, as well as more advanced operations like comparison and inner product. Additionally, `OpenFHE` includes functionalities for distributed, multi-party computing scenarios and proxy re-encryption.

`TFHE` [205, 213] (Fast Fully Homomorphic Encryption over the Torus) is a library that focuses on optimizing the performance of fully homomorphic encryption schemes, released under the Apache License version 2.0. It provides an implementation of the FHE-over-the-Torus Homomorphic Encryption Scheme (TFHE), which is an extension of the GSW FHE scheme with improvements suggested in a line of works [206, 202, 205]. The library offers low-level operations. The `TFHE` library has been superseded by `TFHE-rs` [214], described below.

`TenSEAL` [215] is an open-source (Apache License version 2.0) library developed by OpenMined for secure machine learning using HE. It is built on top of the Microsoft `SEAL` library and provides an easy-to-use interface and high-level abstractions because of the Python front-end for working with homomorphic computations. It integrates well with PyTorch and can be used for privacy-preserving machine learning tasks.

`TFHE-rs` [214], formerly `Concrete`[18] is a pure Rust implementation of TFHE for boolean and integer arithmetics over encrypted data by Zama. It is released under the BSD 3-Clause Clear License. The aim of TFHE-rs is to have a stable, high-performance library supporting the advanced features of TFHE. `Concrete` (Zama TFHE Compiler) has instead been redeveloped as a compiler for TFHE, that converts Python programs into FHE equivalents.

4.3.2 Libraries for SMPC: protocols and sub-components

Many of the academic works that have performed privacy-preserving DL have used existing software libraries in constructing their complete tools. We will now detail some of the common sub-components for oblivious transfer and circuit garbling, and then briefly mention fully-fledged SMPC protocol frameworks.

For oblivious transfer, `LibOTe` [216] provides a framework for implementation of 14 OT protocols from the academic literature with various efficiency and security properties. The repository has 95 forks and is regularly updated, and it has been used in at least three SDLI papers [158, 160, 164].

A Rust implementation of circuit garbling is given in `fancy-garbling`, which is now incorporated into the `swanky` suite by Galois Inc. [217]. `fancy-garbling` is used in the *Delphi* tool [155] and *Muse* [151]. *Gazelle* uses an

[18] https://www.zama.ai/post/announcing-concrete-v1-0-0

extension of the `justgarble` software [218] which was developed by the team behind the fixed-key blockcipher paper [172].

The `ABY` software framework [219], that underpins the *ABY* publication [157], offers numerous speedups when converting data between arithmetic, binary, and Yao (garbled circuit) representation for assisting various tasks in two-party computation. It has been used as a building block library for at least four SDLI publications [154, 150, 220, 149].

Going one abstraction level higher, `MP-SPDZ` [221] is a SMPC software framework that provides a unified tool for benchmarking a large number of SMPC protocols, in various security models, and has developed from an improved version [222] of the SPDZ protocol [180]. It has been used in the *Muse* protocol [151] and recent work on convolutions [223]. The `SCALE-MAMBA` software tool [224], which as of June 2022 is no longer being maintained, provides an alternative for implementing various SMPC protocols in the malicious security setting. Like `MP-SPDZ`, it is also a fork of the original `SPDZ-2` software.

4.4 Software and Libraries for Privacy-Preserving DL

4.4.1 HE only

CryptoNets [162], published in 2016, was the first attempt to apply FHE on Neural Networks (NNs). A key assumption here is that FHE is only applied to the process of inference, not training. While it is technically possible, literature concludes that also performing the training process in an encrypted state incurs too much overhead. Besides, this allows pre-existing models that have already been trained to be adapted for inference on encrypted data, ensuring that the training phase does not need to be repeated. *CryptoNets* employs a leveled HE scheme implemented in Microsoft's `SEAL` [207] library and shows that the same accuracy can be achieved in the encrypted state. The *CryptoNets* solution has various optimizations in order to cope with the complexity of HE and the constraining of introduced noise at small levels. *CryptoNets* operations are defined on the ring $R_q^n := \mathbb{Z}_q[x]/(x^n+1)$ thus it is defined using the q, n and t parameters (where t is the plaintext input ring characteristic i.e., $R_t^n := \mathbb{Z}_t[x]/(x^n+1)$). To keep noise at at a manageable level the authors propose the use of large t values which practically mean that an input message used in the scheme will be encoded as a polynomial of degree n where each coefficient is modulo a large t. To reduce the complexity of the computations using large t the authors propose the use of Chinese Remainder Theorem (CRT) by practically "breaking" t into relatively prime moduli (in *CryptoNets* t_1 and t_2). Then the computations, i.e the encoding of the message, can be done in parallel for t_1 and t_2 without any change. In *CryptoNets* the authors also take into account the fact that when encoding neural network atomic constructs that are real numbers into atomic constructs compatible with HE that are polynomials in R_t^n there is some loss of accuracy in the process. This in *CryptoNets* is handled by converting real numbers into fixed precision numbers and then use their binary representation to convert them into a polynomial with the coefficients given by the binary expansion. However, the paper also proposed another approach that can take place after real number to fixed point conversion. In this case, since t is a very large integer, the fixed point number can be included as a single coefficient on an n degree polynomial of the R_t^n ring (a single scalar). To remedy with the fact that operations done on such polynomial practically are only relevant to a single coefficient (one scalar) the authors suggest to take advantage of the all polynomial coefficients but splitting this single scalar into multiple ones using CRT. In a way, the above process can exploit parallel computation offered by SIMD operations offered in modern processors. Finally, the *CryptoNets* solution use optimal HE scheme parameter selection that will allow the best fit of the scheme to the NN at hand. These parameters are the values used are $t_1 = 1099511922689$ and $t_2 = 1099512004609$ moduli for the CRT of t, $q = 2^{383} - 2^{33} + 1$ and $n = 8192$. Note, that *CryptoNets* assumes that there is a ML as a service scenario where a client is encrypting the input data, then sends them to a NN on a server offering inference as a service, the server performed inference on the encrypted data and returns the encrypted result to the client which then can decrypt them. The *CryptoNets* is tested using the MNIST dataset[19]. The NN linear operations are done using the adopted HE scheme while the non-linear ones i.e., Mean Pooling layer and Square activation layer, are done using linear approximations. The Sigmoid Activation function existing in the MNIST NN is omitted.

[19] https://en.wikipedia.org/wiki/MNIST_database

Several works since *CryptoNets* have aimed at improving the performance and/or scalability for inference on HE-encrypted data. *LoLa* [225], short for *Low-Latency CryptoNets*, tries to enhance *CryptoNets* by treating each layer as an encrypted message instead of each node. The results indicate a speedup of 11 times and the RAM usage for the CIFAR-10 dataset dropped from 100s of GBs to only a few. *CryptoDL* [226] employs leveled HE as implemented as `HElib` and states that in their case only addition and multiplication are the allowed mathematical operations. Therefore, the authors propose alternative activation functions restricted to those mathematical operations and show that accuracy is still on par while being faster than previous work on secure ML inference. *Faster CryptoNets* [227] proposes several more efficient approximations of the activation functions to make them HE-friendly. Additionally, they adapt the pruning and quantization process to optimize the compression of the model's size to make it a better fit for HE. They employed BFV-RNS as the HE scheme. Alternatively, *TAPAS* [228] proposes to switch to a different kind of NNs called Binary Neural Networks (BNNs), which is more suitable for the TFHE scheme. However, the limitations of BNNs compared to traditional NNs are not discussed by the authors, although they do plan to support in future work non-binary NNs. *FHE-DiNN* [229] is less restrictive, as they switch to Discretized Neural Networks (DiNNs) which allows $\mathbb{Z}$ for the signals and weights instead of just $\{-1, 1\}$. Moreover, each neuron is bootstrapped, ensuring that the homomorphic evaluation of a neuron's output is not restricted by the size of the NN anymore.

Finally, *CHET* [230] takes a different approach. They authors observed that creating an FHE application, i.e. adapting software to work on FHE-encrypted ciphertext rather than the plaintexts, is a painful process. There is lots of complex mathematics involved to finetune the parameters HE schemes offer, which is not necessarily part of the toolbox of the average software engineer. To accelerate adoption, *CHET* proposes a domain-specific language to specify the tensor circuits in that need to be evaluated in an encrypted fashion. *CHET*'s compiler performs some kind of parameter exploration to optimize the parameter values automatically for the given circuit. They evaluated their approach on an industry-grade medical imaging model. While a naive implementation would require 18 hours of runtime per image, and could be hand-tuned by an expert to 45 minutes per image, *CHET* managed to reduce this even further to just 5 minutes.

4.4.2 SMPC only

Chameleon [150] extends the Secure Two-Party Computation (2PC) model with a semi-honest third party in the offline phase with the task to provide correlated randomness, enabling performance optimizations that this framework exploits. In addition, the framework adds support for signed fixed-point arithmetic. Amongst others, the authors have evaluated and compared their work with *MiniONN* [154] (see below) in two configurations. In the configuration offering the same accuracy as *Chameleon* [150] on MINST, the total communication is reduced from 657.5 MB to 10.5 MB and the total training time from 9.32 s to 2.24 s. When *MiniONN* [154] is configured such that the accuracy is 1.4 percent point less, *MiniONN* is faster (1.04 s opposed to 2.24 s), but the communication overhead is still larger: 15.8 MB versus 10.5 MB.

ABY^3 [158] fully extends to a three-server model, or Secure Three-Party Computation (3PC), where at most one server may be malicious. Note that this is a stronger adversarial model than the semi-honest one, as now the adversary may deviate from the protocol. Switching to the 3PC model enables optimizations, such as a significant reduction of the communication overhead for vectorized operations. Fixed-point multiplication is also supported. With the same neural network configuration as *SecureML* [153], the online time is reduced from 193 ms to 3 ms and the communication overhead from 120.5 MB to merely 0.5 MB.

EzPC [220] is a 2PC framework that follows a similar approach to *Chet* [230] that generates 2PC protocols from higher-level programs, with `EMP` [231] and `ABY` [219] as the underlying frameworks. The contribution in this work is to provide an automated method of choosing whether arithmetic or boolean circuits are most appropriate for the program that is provided as input, while maintaining the security properties of the underlying 2PC protocols.

In *XONN* [164] the authors assume a single server and used binary neural networks, i.e. limiting the weights, activation values and biases to $\{-1, 1\}$, in order to make the (Boolean) circuits that represent the NN simpler and more SMPC friendly. In particular, multiplications become XNOR gates, which can benefit from the free-XOR technique. The inputs to the NN are not necessarily binary so two approaches to this first layer are given, using a dedicated Boolean circuit that can be combined with the other GC layers, and a combination of A-SS

and OT (thus the GCs start at the second layer). If the first option is used then the entire system is based on garbled circuits and thus the evaluation phase is entirely non-interactive. Further, this approach allows the servers to keep the architecture of the DL model hidden: this would not be possible in the single server setting. In the training phase, the number of neurons in each layer must be increased to compensate for the accuracy lost by moving to binary weights. This might cause issues when converting existing machine learning tools to the *XONN* framework.

ABY2.0 [232] is a framework that targets computation on an ℓ-bit field in a two-party setting, or 2PC. The adversary is considered to be semi-honest. That is, parties will perform the computation in accordance with the protocol used, but a party might try to obtain more information from the protocol messages that have been communicated. To support decimal numbers, fixed-point arithmetic is supported. The framework was evaluated on neural networks with the same configuration as *SecureML* [153]. Compared with *SecureML* [153], the number of training iterations per minute was increased up to 3.46 times. For inference, runtime was improved up to 3.3$\times$ and throughput even in best-case scenario by a factor 754$\times$.

4.4.3 Hybrid approaches

As discussed earlier on in this section, many of the modern works that we will describe in this section use a combination of PETs in the process of performing ML tasks.

SecureML [153] employs three-party computation by assuming two servers that do not collude. The idea is that the two servers compute the desired circuit amongst themselves on shares of the client's input, for both training and inference. A pre-processing phase was employed to generate multiplication triples using both HE and OT, with optimizations for the vectors and matrices that they use to boost efficiency. A-SS was used for the linear layers, while garbled circuits are used for the non-linear layers. The main theoretical advancements provided by the paper are an activation function that can be computed using a small garbled circuit, which is essentially the sum of two ReLU function calls, and the replacement of the softmax function (for the output layer) with a combination of ReLU functions and the simple operations of addition and division. A large portion of this work was devoted to efficient linear and logistic regression in the privacy-preserving scenario, while the results on neural networks are mainly to demonstrate feasibility: this was the first instance of training a neural network using PETs.

MiniONN [154] assumes the same semi-honest 2PC model as *ABY2.0* [232], but focuses specifically on not changing the training phase of the model. This way, pre-trained models can be transformed without retraining into *oblivious* ones, i.e. those where the remote server performing inference cannot infer anything from the input data. To this end, the authors identified widely used ML operations, such as sigmoid and tanh, and developed oblivious-friendly counterparts. Compared to *SecureML* [153] with the MNIST dataset, the accuracy has increased from 93.1% to 97.6%, the offline latency has been reduced from 4.7s to 0.9s, and the online latency from 0.18s to 0.14s.

Gazelle [160] continued the trend of identifying the computation- and communication-heavy aspects of NNs inference and attempting to reduce multiple stages of the workflow. This was effectively the first major work that used specialized operations for each part of the NN, namely HE for the linear layers and GCs for the activation functions. The paper introduced a number of optimizations that have become core parts of later literature and libraries, and uses a Packed Additively Homomorphic Encryption (PAHE) scheme, namely a modified version of BFV. The idea of PAHE when used here is that it allows packing multiple plaintexts into individual ciphertexts, and the transition to SIMD means that the *Gazelle* system never uses the very expensive operation that is homomorphic multiplication of ciphertexts (only scalar multiplication, addition and permutation of slots are needed). The first main contribution is acceleration of permutation support: SIMD homomorphic computation requires slot rotation, and instead of using a slower prime-order transform the paper suggests to use a power of two for NTTs. The second major speedup comes from using FHE moduli that are Barrett-friendly [233], again leading to much faster NTTs. Finally, the paper gives custom linear algebra kernels for fully connected and convolution NN layers, mapping these layers to homomorphic matrix-vector multiplication/convolution operations.

Delphi [155] follows the same system setting as *Gazelle* namely NN inference using a combination of SMPC and FHE. The core idea of *Delphi* is to move as many operations as possible to a pre-processing phase that is independent of the client's input: the linear layer input by the server (model owner) is known beforehand, meaning that secret shares of the server's input can be made (using the BFV PAHE as in *Gazelle*) and sent in advance. In addition to reducing communication costs for linear layers in the online phase, the operations that are performed during the online phase are done over small prime fields and can therefore benefit from CPU and GPU acceleration. The second contribution is to move away from solely using GCs for activation functions, and the paper introduces a planner that identifies which activation functions can be replaced with polynomial approximations (thus gaining great speedups in the online phase) without reducing the accuracy of the system, and which functions cannot be replaced and should be calculated as is. Computing `ResNet-32` requires 60 MB of communication and 3.8 s in the online phase.

Muse [151] also takes into account malicious parties, but here only malicious *clients*. In the paper, the authors argue that servers in general should be more secure than clients. Hence, this should be a safe assumption. The contribution of this work is twofold, as it firstly proposes an attack on frameworks under the semi-honest adversarial model demonstrating the practicality of malicious clients breaking the secure protocols. The protocol employed by *Muse* [151] is based on *Delphi* [155]. To protect against malicious clients, Message Authentication Codes (MACs) are employed. The authors argue that the MACs can be computed and used in the preprocessing phase. Thus, the online phase has a similar execution time as *Delphi*. Overall, the performance is slightly worse when compared to semi-honest alternatives, but the performance is better when compared with prior art assuming malicious computational parties.

nGraph-HE [234] and *nGraph-HE2* [235] extend the Intel DL graph compiler `nGraph`[20] to include a homomorphic encryption backend, using the `SEAL` library [207]. In follow-up work, *MP2ML* additionally incorporates SMPC components in the privacy-preserving backend via the `ABY` framework [157]. The main contribution of these works is to incorporate optimizations automatically while separating privacy-enhancing technology from DL as much as possible: the privacy-preserving layer has its own instruction set and can perform for example batch-norm folding and parallel operations via SIMD packing. The works only target inference and not training, and for the HE-based approaches the non-polynomial activation functions (ReLU, MaxPool) are evaluated in the clear by the data owner (thus leaking weights and parameters of the model). By incorporating `ABY`, the *MP2ML* framework allows the hiding of the intermediate feature maps, and needs a mechanism for converting between CKKS [203] and SMPC to keep the activation function operations hidden from the client. *Chimera* [159] uses the TFHE scheme [202] to perform ReLU, and other functions are done with BFV/CKKS, enabling the hiding of non-linear layers (not just weights and parameters, but even the function being computed in these layers), but at a cost of requiring conversions between the two HE schemes.

`CrypTFlow2` [148] is a set of tools that can transform `TensorFlow` [236] models to a secure implementation. `CrypTFlow2` uses secure two-party computation (2PC) and guarantees that outputs are bit-wise equivalent to the cleartext model stated in `TensorFlow`. Specifically, `CrypTFlow2` relies on introduced 2PC protocols that achieve secure comparison and division. It targets inference and it is showcased for prominent deep NNs that have successfully addressed the ImageNet challenges. `CrypTFlow2` focuses on two axes: (1) realistic deep NNs use ReLU activations, expensive to compute securely; (2) a faithful implementation of secure fixed-point arithmetic is required to maintain the inference accuracy of a given plaintext model. More specifically, `CrypTflow2` relies on:

- Introduced protocols for millionaires' problem and a DReLU, which are suitable for the implementation of non-linear layers of NNs such as ReLU, Maxpool and Argmax.
- Introduced protocols for division. Combined with introduced theorems on fixed-point arithmetic over shares, they are demonstrated on the evaluation of linear layers, i.e., convolutions, average pool and fully connected layers. The derived evaluations are faithful.
- Two different types of Secure Deep Learning Inference (SDLI) for the evaluation of linear layers, based on HE and OT.

[20] An interface for ML frameworks such as TensorFlow [236] that has been developed by Intel: `https://www.intel.com/content/www/us/en/artificial-intelligence/ngraph.html`.

The cryptographic primitives used in CrypTFlow2 are 2-out-of-2 Additive Secret Sharing (A-SS), Oblivious Transfer (OT), Multiplexer and Boolean-to-arithmetic (B2A) conversion, and HE. The FHE scheme employed is the Brakerski-Fan-Vercauteren Homomorphic Encryption Scheme (BFV), also used in Gazelle [160] and Delphi [155]. The system uses optimized algorithms of Gazelle for homomorphic matrix-vector products and homomorphic convolutions. The implementation of SDLI using OT is based on EMP [231]. The linear layer implementation using HE is based on SEAL [207] and Delphi. The performance achieved is quantified on commodity hardware for relevant benchmarks. Specifically, the hardware machines used, each comprise a four-core 3.7-GHz Intel Xeon processor featuring 16GBs of RAM. Two communication scenarios are comparatively evaluated, a LAN setting and a WAN setting. The LAN setting achieves 377 MBps bandwidth and 0.3 ms echo latency. The WAN setting (transatlantic) has a 40 MBps bandwidth and an 80 ms echo latency. The result achieved shows that in the WAN setting where communication cost is high, HE-based inference is always faster while in a LAN setting, neither OT or HE is always better, the choice depends on the benchmark. CrypTFlow2 improves CrypTFlow [161]. It keeps the prior front-end that transforms TensorFlow inference code into an intermediate format, and it modifies the back-end system which finally derives the secure code. CrypTFlow comprises three components, namely, *Athos*, an end-to-end compiler mapping TensorFlow models to a variety of semi-honest SMPC protocols; *Porthos*, a semi-honest three-party protocol that significantly accelerates TensorFlow-like applications; and *Aramis*, which uses hardware with integrity guarantees to convert any semi-honest SMPC protocol into an SMPC protocol that provides malicious security. The derived implementations match the inference accuracy of plaintext TensorFlow models. Both systems, i.e., CryptFlow and CryptFlow2 rely on *EzPC* [220] for the back-end operations.

Cheetah [237] is a highly optimized system architecture comprising of the software implementation of protocols for secure two-party computation neural network inference (S2PCNNI). It achieves faster and more communication-efficient performances than certain implementations of state-of-the-art [148], firstly by the careful redesign of encryption-based protocols that can evaluate the linear layers without any expensive rotation operation and secondly by including several lean and communication-efficient primitives for the non-linear functions (e.g., ReLU and truncation). More specifically, in the linear layers, based on the fact that polynomial multiplication can be viewed as a batch of inner products if coefficients are arranged properly, *Cheetah* replaces the matrix-vector multiplications involved in the functionalities $\mathcal{F}$: {convolution, batch normalization, and fully-connection} by polynomial arithmetic, i.e., polynomial multiplication, therefore eliminating the expensive rotations included in the original matrix-vector multiplication computations. To achieve this, *Cheetah* introduces three pairs of encoding functions $(\pi^i_{\mathcal{F}}, \pi^w_{\mathcal{F}})$, (one pair for each of the functionalities $\mathcal{F}$), which map the values of the input (e.g., tensor or vector) to the proper coefficients of the output polynomial(s), thus allowing the evaluation of the linear layers of the deep NNs via polynomial arithmetic circuits instead of matrix-vector multiplications. With the help of $(\pi^i_{\mathcal{F}}, \pi^w_{\mathcal{F}})$, it is shown in [237] how to evaluate the functionalities $\mathcal{F}$ privately. Also, $\pi^i_{\mathcal{F}}$ and $\pi^w_{\mathcal{F}}$ are well-defined for any $p > 1$ such as $p = 2^l$, allowing the protocols of *Cheetah* to accept secretly shared input from the ring $\mathbb{Z}_{2^l}$ for free. In this context, special concern is given to avoiding extra information leakage when only certain coefficients are expected to be received by a party, while receiving the polynomial arithmetic outcome. Also, a partitioning scheme is proposed to split large input tensors and kernels into smaller blocks and zero-pad the margin blocks so that each of the smaller blocks can fit into one polynomial. The proposed protocols can then be applied on the corresponding pair of subtensor and subkernel. *Cheetah* outperforms SIMD-based approaches [238], which require the plaintext to be from a larger prime field. In the non-linear layers *Cheetah* uses (Vector Oblivious Linear Evaluation) VOLE-type OTe of protocols for the non-linear functions [239], delivering lower communication complexity for the cases of the parameters used in neural network inference. *Cheetah* also offers improvements to the truncation protocol required after each multiplication so that fixed-point values will not overflow, thus achieving further performance gains in the non-linear layers. The improvements are designed based on specific observations deriving from simulation experiments, as, for example, that the two probability errors introduced by the local truncation protocols appear to be of different impact on the overall NN computation, or, that sometimes the most significant bit (MSB) is already known before the truncation. The protocol designs of *Cheetah* in the non-linear layers result in faster running time and bring down more than 90% of the communication cost compared to the corresponding protocols of [148]. For the implementation of *Cheetah* the SEAL library, the HEXL accelerator and the EMP [231] toolkit have been used. Intensive benchmarks over several large-scale deep neural networks are being reported in [237], all showing the latency and communication performance improvements. For example, an end-to-end execution for ResNet50 under a

WAN setting costs less than 2.5 minutes and 2.3 gigabytes of communication with *Cheetah*, which outperforms `CrypTFlow2` [148] by about $5.6\times$ and $12.9\times$, respectively.

SecretFlow [240] is a framework that supports the development of privacy-preserving machine-learning and data-intelligence applications through a Python interface. It includes, among other components, re-written versions of most of the *Cheetah* protocols and offers a layered approach to privacy-preserving application development. At the lower layer, *SecretFlow* provides abstractions of devices featuring secure processing and a variety of HE schemes. Using higher levels of the framework, algorithms and applications can be build exploiting the secure devices. The development of secure machine-learning algorithms is supported with a focus on Federated Learning (FL)-related algorithms, providing suitable Python abstractions. Several means are supported for secure aggregation, including the use of SMPC-based *SecretFlow* security devices, and masking with one-time pads [241]. Plaintext aggregation is also supported for evaluation purposes. The framework utilizes `TensorFlow` [236] and `PyTorch` [242] further facilitating its integration to widely used ML system design flows. More on FL-related libraries and tools can be found in Section 3.5.

4.5 Scaling up SMPC/FHE solutions

The vast majority of the literature on using PETs for DL assumes small numbers of parties and the use of commodity hardware, usually a laptop-grade device for the client(s) and either a powerful laptop or small server-grade device(s) for the model-owning server. In this section we will explore how performance gains can be attained in this context from dedicated, powerful hardware, HE component-level acceleration and software tools/techniques to more accurately reflect the health application use cases that are within the purview of the SECURED project.

4.5.1 Hardware Acceleration

The computational load of SMPC/HE-based processing applications, such as SDLI and SDLT, and the communication overhead required to attain secure data exchange between the parties, whether referring to a 2PC client-server model or to a multi-party model, limits their widespread application to real-world problems. Although software libraries, as mentioned in the previous paragraph, provide a very useful tool for privacy-preserving processing, the gap between performances on plaintext *vs.* ciphertext on general-purpose computing platforms is enormous, especially in the context of NN/DL computing. Hardware accelerators provide a powerful tool to bridge this gap. Computationally demanding parts (modules) of the HE algorithms are mapped to GPUs, ASICs and FPGAs, after being optimized for this purpose; optimizations aim to achieve efficient performance metrics, such as low latency, high through-put, etc, while keeping track with the security parameters.

4.5.1.1 GPUs for HE

Graphical Processing Units (GPUs) offer an attractive choice for hardware platforms to cope with processing secured by HE techniques. The superior amount of computational capabilities incorporated by GPUs can accelerate the intensive arithmetic operations of modular multiplication, large polynomial multiplication, and matrix-vector multiplication, which appear in such applications, while exploiting their inherent parallelism.

The first implementation of an accelerator for an HE scheme on a GPU appears in 2012 [243]. It concerns efficient large-number modular multiplication in the size of million bits, and employs Strassen's FFT-based algorithm combined with Barrett's modular reduction algorithm [233]. Experimental results for the small setting of the Gentry-Halevi FHE scheme [244] with a dimension of 2048 on NVIDIA C2050 GPU, show speedup factors of $7.68\times$, $7.4\times$ and $6.59\times$ for encryption, decryption and noise reduction (recrypt) respectively, when compared with the available, at that time, CPU implementation of GH [244].

Since then, a number of implementations of GPU-accelerated HE schemes have been reported in the literature [245, 246, 247, 248, 249, 250, 251, 252, 253]. Advances are made in terms of execution time via the parallel execution of certain HE schemes (CKKS, BFV, CMNT, LWE) on GPUs, with the improvements of

the various implementations most of the times achieved by optimizing the NTTs in the HE scheme. Modular-reduction techniques or reductions of specific moduli and/or RNS/CRT arithmetic offer improvements in certain cases. Optimization may also regard the scheduling of primitives and memory access. For example, reducing host-to-device and device-to host transfers, applying register-based constant-coefficient (i.e., twiddle factor) storage, multi-stream computing, and asynchronous computing are some of the techniques which have been applied. An efficient GPU-based implementation of the Torus FHE scheme [254] appears in [255], along with a comparison to CPU-based implementation. Besides NN-layer core operations, bootstrapping has also been treated [249, 256]. Recently, a GPU-based acceleration of a privacy-preserving inference scheme, which classifies encrypted genome data for tumor types has been reported [253]. The performances reported on GPU-accelerated HE, appear promising for the further development of implementations related to secure medical environment ML applications, such as privacy-preserving inference for the classification/interpretation of data sets or images and scans of low/medium resolution.

4.5.1.2 Intel HEXL acceleration

Several of the existing FHE libraries take advantage of Intel's Homomorphic Encryption Acceleration Library (Intel HEXL [257]) that manages to offer an optimal realization of core FHE arithmetic operations at ISA level when used. Intel HEXL is a C++ library which provides optimized implementations of polynomial arithmetic for Intel processors by taking advantage of Advanced Vector Extensions 512 (AVX 512) instruction set operation. The library is focused on optimizing polynomial multiplication and NTT operations for large sizes like the ones used in FHE by practically providing efficient Intel AVX 512 implementations for element-wise vector-vector and element-wise vector-scalar polynomial multiplication and the forward and inverse NTT. Intel HEXL is designed to intercept HE libraries at the polynomial layer assuming that the library uses polynomials in Residue Number System (RNS) form. Since most of the HE operations at runtime are including many operations at the polynomial layer, the expected polynomial level speedup will result in substantial speedups at higher-levels of HE operations. Intel HEXL is integrated into the Microsoft `SEAL` library, the `PALISADE` library and its extension `OpenFHE` library as well as any similar libraries built on-top of `SEAL` that follow RNS approaches.

4.5.1.3 Techniques and Algorithms for Optimizing Homomorphic Encryption Custom/Dedicated Hardware Accelerators

In the literature, several custom-FHE hardware accelerators have been reported, based on arbitrary parameter sets such as plaintext parameters, ciphertext parameters, parameters for noise distributions, which were not tested to be secured. An effort has been made to define parameter sets for each scheme in the community-led HE Security Standard [183], where parameter sets and schemes are believed to be secure against state-of-the-art attack literature. This section focuses on techniques expected to be useful for the hardware implementations of the libraries in Table 8, based on the parameter sets of [183], targeting Field Programmable Gate Arrays (FPGAs) and/or Application-Specific Integrated Circuits (ASICs). Prior custom implementations that rely on arbitrary parameters, which may not be secure, are not addressed; furthermore, custom FHE schemes that are not widely used, are also omitted, keeping the state of the art of this part beyond 2019.

Accelerating Multiplier. The plaintext polynomial ring is defined as $\mathbb{Z}_t[x]/(x^n+1)$, i.e., the set of polynomials with degree less than n and coefficients in $\mathbb{Z}_t$, where plaintext modulus t and the ring dimension n are both integers. The ciphertext space is defined as $\mathbb{C} = \mathbb{Z}_{q_l} \times \mathbb{Z}_{q'_l}$, where $\mathbb{Z}_{q_l} = \mathbb{Z}_{q_l}[x]/(x^n+1)$ with integer q_l defining the chiphertext modulus at level l. The FHE operations that are applied on these polynomial rings are addition, multiplication and modulo reduction.

As the ciphertext modulus q_l becomes wider (128-bit is commonly used) to leverage the security robustness of the schemes, the complexity especially of multiplication gets larger demanding techniques to decrease it's complexity from $O(n^2)$. Application of RNS is common, where modulus q_l can be split into smaller moduli q_{l_i} and operations from $\mathbb{Z}_{q_l}$ are mapped to multiple operations on $\mathbb{Z}_{q_{l_i}}$, decreasing the complexity of hardware-implementations on multiplication and automorphishm.

Based on Schönhage-Strassen Algorithm. Common multiplication schemes like KA or schoolbook multiplication method, become nearly infeasible for very-large integer multiplication (million of bits). The first most outstanding implementations in terms of designing a multiplier accelerator , which was based on Schönhage-Strassen Algorithm and was operating on multi-million bit numbers, is [258]. At that current time it was the first chip-implementation that outperformed GPU state of the art, based on 90-nm (72–102$\times$ faster than GPU state of the art) and was the start of meaningful FHE custom hardware acceleration. This was a general implementation though, not focused on a specific FHE scheme.

Based on Number Theoretic Transform. One of the basic modules in multiplication over integers is Number Theoretic Transform (NTT), where operations are transformed to a "frequency domain," thus decreasing the multiplication complexity from $O(n^2)$ to $O(n \cdot \log(n))$. In most cases Negative Wrapped Convolution (NWC) is used, in order to minimize the complexity of the algorithm, which facilitates the evaluation of a full polynomial multiplication that implicitly includes the reduction modulo $X^n + 1$, without increasing the length of the inputs, otherwise the input size would be double the input-length.

Most implementations of current schemes are based on RNS, as discussed previously, designing efficient NTT multipliers which are splitting a large multiplier into smaller ones [259], optimizing furthermore resources and execution time, making Schönhage-Strassen Algorithm useless in terms of area-utilization.

Based on Karatsuba Algorithm. Karatsuba Algorithm (KA) [260] is a divide-and-conquer algorithm, which reduces the multiplication of two k-digit numbers to three multiplications of $k/2$-digit numbers, a process which can be recursively applied to at most $k^{\log_2 3} \approx k^{1.58}$ single-digit multiplications. KA has several advantages compared to FFT/NTT, despite its highest asymptotic complexity. First, it is a simple algorithm, with basic pre- and post-computations and, therefore, can be easily implemented. Second, KA can perform polynomial multiplications with non-power of two degrees, allowing it to fit more precisely to the required parameters.

KA also has some limitations. First, KA requires a software/hardware co-design approach to meet competitive computation times, which is not the case for FFT/NTT. Second, KA is a good alternative to FFT/NTT only to a certain degree of n, as the advantages disappear after. Migliore et al. [261] present an extensive investigation of offered benefits, for different n values, comparing KA to FFT for the BFV scheme (optimization of execution time at 11.9 ms instead of 15.46 ms and 50% reduction of the logic utilization and registers of the FPGA until a certain value of n).

Modulo Reduction. As discussed previously, Modulo Reduction is also on of the basic operations. Is used to keep the integer numbers, after calculations, inside $\mathbb{Z}_q$. There are several ways to perform Modulo Reduction, the most commonly used ones are Montgomery Reduction with KA [260], which is the optimal way in terms of hardware implementation. Another classic reduction technique is Montgomery reduction optimized for particular modulus q_l values [262].

Accelerators/Co-processors. There are several implementations regarding accelerators/co-processor regarding FHE schemes.Table 9 summarizes the best in terms of speedup relative to software, frequency and area utilization, allowing a straight forward comparison and a starting point for the reader. These solutions are implemented on FPGAs, in contrary to Table 10, which contains solutions on ASICs. Need to mention here that all solutions except CoFHEE [263] has not real ASIC hardware and aiming to do so in their future work. Also need to note that comparison of speedup was made by running `SEAL` [207] benchmark Set-1 and Set-2 by a single-thread, or translate the results to end up with comparable metrics.

Work	Platform	× Speedup w.r.t. software	Freq (MHz)	Area Utilization (%)
ReMCA [264]	Virtex-7	NA	250	6.5L+20.8B
SRTJ [265]	ZCU102	13	200	50L+90B
HEAWS [266]	AWS-F1	20	250	75L+83B
HEAX [267]	Stratix 10	164	300	64L+80B
Medha [268]	U250	137	200	55L+72B

Table 9 – Comparison of real hardware implementations, data from [268] (In the Area column, 'L' and 'B' stand for % of logic and on-chip RAM elements used).

Work	Technology (nm)	× Speedup w.r.t. software	Freq (MHz)	Area Utilization (mm^2)
F1 [269]	14/12	17K*	1 to 2	151.4
BTS [270]	7	2.2K*	up to 1.2	373.6
BASALISC [271]	12	4K*	1 to 2	150
CraterLake [272]	14/12	8.7K*	1.2	472.3
ARK [273]	7	36K*	NA	418.3
CoFHEE [263]	55	2.5	0.25	15

* Throughput is estimated by simulating a model of the accelerator.

Table 10 – Comparison of simulated ASIC implementations, data from [268]

ASIC proposed solutions In this section we will analyze two of the most promising ASIC-implementations of Table 10. Analyzing their architectures and the techniques used that outperform the ones of Table 9, in terms of speedup.

Trebuchet. TREBUCHET project [274] accelerates commonly used FHE schemes (BGV, BFV, CKKS, FHEW, etc.) providing 128-bit security at least, while integrating with the open-source `PALISADE` and `OpenFHE` libraries (offering $10\times$ acceleration relative to previous state of the art implementations). TREBUCHET supports common lattice-based FHE schemes and provides a means to explore trade-offs offering a wide range of chip sizes in order to achieve execution time performance an order of magnitude faster than the solutions in Table 9 The system architecture comprises three layers:

1. The Application Layer, which comprises users' applications written in C++, employing `OpenFHE`.
2. The Software Layer, which maps applications into the TREBUCHET hardware accelerator and is composed of three sub-layers, (1) the `OpenFHE` [212] library, which provides secure FHE schemes, (2) the so-called SPIRAL NTTX system, which maps high-level calls of `OpenFHE` API into kernels, i.e., software microcode functions, used to program the TREBUCHET hardware accelerator, and (3) a microcode compiler which generates the firmware instructions that control the hardware units which process Large Arithmetic Word Size (LAWS) data.
3. The Hardware Layer, which consists of the DPRIVE[21] Accelerator ASIC (DA) and an FHE Processing Board (FPB) which features the DA. The DA is composed of multiple modular components.

The DA is organized as a modular parallel and vectorized architecture. The main building blocks are the so-called Ring Processing Units (RPUs). RPUs are on-chip tiles that contain multiple Arithmetic Logic Units (ALUs) and perform modulo arithmetic. Data, such as ciphertexts and keys, are stored in shared vector-data SRAM. Furthermore, RPU perform memory management opting for data to be placed near computational elements. Data movement is minimized and data-level parallelism is exploited by using multiple instances of the tiles, across the device. The architecture is scalable in terms of the bit width and allows for customization of several parameters such as the number of multipliers and the memory size per tile, and the number of tiles.

[21] DPRIVE is a U.S. Government research project investigating hardware acceleration for FHE `https://www.darpa.mil/program/data-protection-in-virtual-environments`.

Basalisc *BASALISC* [271] is an architecture family of hardware accelerators designed to perform FHE computations in the cloud. *BASALISC* implements the Brakerski-Gentry-Vaikuntanathan Homomorphic Encryption Scheme (BGV) supporting up to 128-bit security and includes bootstrapping in the computations. As the majority of HE hardware implementations, *BASALISC* exploits the massive parallelism available in NTT/RNS. For the bootstrapping procedure, it relies on Montgomery-friendly primes to achieve savings of up to 46% in logic area and 40% in power consumption, compared to generic multipliers capable of supporting all moduli [275].

The main parts of the architecture data-path are a computational core, and a four-layer memory system which is used for storing ciphertexts and keys. In the computational core, three different Processing Elements (PEs) are distinguished: the asynchronous-logic Multiply-Accumulate, the Permutation PE, and the NTT PE. The asynchronous logic provides important area and latency savings. The memory components are structured as follows: **i)** Distant Memory, i.e., off-chip DRAM where data scheduled for processing and results ready for retrieval by the host are kept. Two DDR4 interfaces work in parallel, connecting with two DRAMs, so that certain operations such as loading new data to be processed and storing results to be retrieved by the host can be done in parallel, without causing stalls. **ii)** Middle Memory, i.e., a conflict-free ciphertext buffer (CTB), which, when co-operating with the available layout permutation unit and the generator for the constants required in the transform (twiddle factors), reaches the delivery of 32 Tb/s radix-256 NTT computations. **iii)** Local MAC Register File, and **iv)** Local MAC Accumulation Register.

The *BASALISC* system is connected to a CPU host, and they communicate either via Direct Memory Access (DMA) or via Peripheral Component Interconnection (PCI), with the assistance of a simple interrupt-driven protocol.

BASALISC provides three independent levels of abstraction in the instruction set. The highest level comprises **Macro-instructions**, which use the largest data types, such as plaintexts and ciphertexts in their entirety, and realize the operations needed to implement the BGV scheme, i.e., ciphertext addition, multiplication, operations needed for ciphertext refreshing, [276] and bootstrapping. **Mid-Level Instructions** include memory management instructions and instructions that facilitate the use of the combined RNS/CRT data representation; i.e., a residue polynomial comprising up to 2^{16} 32-bit polynomial coefficients. Operations supported at this level include element-wise multiplication of a residue polynomial by a constant, computation of NTT, base extension etc. At the lowest level, the **Micro-instructions**, refer to the basic functioning of the PEs. Data at this level are formed by as many a number of coefficient words that can be dealt with by a PE at the same time, or fetched in one cycle. An example of an operation on this level may be a multiplication of two operands and the addition of the result to accumulate. Instructions at this level are fed to the processor via the PCI bus. Memory hierarchy is managed in *BASALISC* by means of explicit Load-Move-Store semantics. A register-like addressing mode is used for all memory-hierachy levels.

The *BASALISC* 1.0 is the first implementation of the *BASALISC* architecture. It is a single-chip FHE co-processor designed in a 12-nm Global Foundries technology. It uses additional off-chip memory. Performance simulations reveal a 4000$\times$ speedup to bootstrap a ciphertext, compared to `HElib` on a Intel Xeon E5-2630 CPU at 2.6 GHz running a single thread.

4.5.2 Algorithmic Acceleration

Sliced Implementations. To achieve high-performance cryptographic implementations in various platforms, developers have largely relied on bitslicing [277] and more recently fixslicing [278] techniques that emulate a SIMD architecture in assembly. These techniques have successfully accelerated standard AES cryptography [279] and various symmetric or public key cryptosystems, often coupled with secret-sharing countermeasures [280, 281, 282]. Expanding to FHE, implementors have naturally utilized slicing to accelerate the underlying cryptographic primitives of schemes like AES, increasing the overall performance. More recently, implementors opted for sliced representations directly on the FHE algorithm. Cheon et al. [283] have extended existing schemes like DGHV to a batch processing mode that utilizes byte-level and state-level slicing to pack and process many plaintexts in parallel using the Chinese Remainder Theorem (CRT). Similar approaches utilized horizontal and vertical packing techniques [284] of coefficients to improve performance. Finally designers have considered slicing in the context of ML algorithms implemented using FHE [285].

Beyond AES-128 and SHA-256. The standardization of AES in 2001 has drawn large attention from the cryptography implementor community and yielded numerous incremental improvements with respect to latency, throughput, area and power consumption, in both software and hardware [286, 287]. Despite these advances, the widespread demand for cryptography on small scale such as IoT applications, the quick growth of SMPC, FHE and Zero Knowledge (ZK) technologies (often coupled with cloud technologies) and the need for fault and side-channel resilience has rendered visible the shortcomings of the secure yet computationally intensive Rijndael algorithm (standardized as AES). In a similar fashion, the sub-optimal SHA-2 hash function has often caused performance bottlenecks in modern applications. In many SMPC schemes and in particular those based on garbled circuits, the computation bottleneck is performing symmetric encryption operations either locally or as a joint protocol execution. Although the literature on joint execution of AES is rich [288, 289, 290], and these papers emphasize the difficulty of computing a function which has a relatively high multiplicative depth in the garbled circuit setting.

As a result, in recent years, research strands emerged aiming to replace existing cryptographic primitives with more efficient algorithms. This new frontier of algorithmic design aims at primitives that are tailor-made to the application context. For instance, the need for small-scale cryptography led to various lightweight symmetric AEAD[22] and hash schemes [291, 292] that can provide security in a resource-constrained environment. Likewise, the need for security against side-channel and fault attacks called for easy-to-mask primitives. This strand resulted in the design of non-linear layers with low multiplicative complexity to reduce the computational overhead associated with the quadratic (w.r.t. the masking order) cost of masking multiplications [293, 294]. Several lightweight and easy-to-mask primitives utilized 4×4-bit sboxes with multiplicative complexity of 4 [295].

SPN cipher designs for SMPC, HE, ZK. Motivated by such advances, designers have identified the need for primitives with low multiplicative complexity and low multiplicative depth in the application context of SMPC, FHE and ZK. Beginning with *LowMC* [296] which is based on a Substitution-Permutation Network (SPN) design strategy; designs improved performance by minimizing the number of $GF(2)$ multiplications needed per encrypted bit using an sbox with multiplicative complexity (MC) of 3 and multiplicative depth (ANDdepth) of 1. In addition, *LowMC* utilized partial non-linear layers (a Partial Substitution-Permutation Network (P-SPN) structure) that further improve performance by relaxing the requirements of the wide trail design strategy [297]. Notably the design demonstrates flexibility since it can be instantiated for various block and key sizes, and became the core function of the *Picnic* post-quantum digital signature scheme [298]. *LowMC* however did not provide inherent support for multiplications in $GF(2^m)$ and $GF(p)$ and required costly conversions between field types, leading to performance issues in several protocols and triggering the development of the *MiMC* encryption and hashing algorithm [299]. *MiMC* is also a flexible primitive that can also be utilized within a sponge construct [300] and was extended for usage in a Feistel network [301] in the *GMiMC* variant [302]. It has successfully accelerated zero-knowledge SNARK[23] applications (including Zerocash [303] that underpins the Zcash cryptocurrency) and zero-knowledge Boolean circuits. Notably, the low multiplicative complexity of both *LowMC* and *MiMC*/*GMiMC* makes them also well-suited for masking countermeasures against side-channel threats, an attack vector which merits consideration in the context of hardware ledgers for digital currencies.

These novel P-SPN designs faced several security issues such as the differential and linear attacks against Zorro [304, 305], coupled with algebraic attacks on *LowMC* and *MiMC*/*GMiMC* [306, 307]. This exacerbated the need for generic security frameworks for SMPC/FHE-friendly ciphers that use the P-SPN structure. To this end, the *HADES* framework [308] combined full and partial non-linear layers to provide wide-trail security arguments for P-SPN, while still resisting algebraic attacks. This design effort culminated in the *HADESMiMC* cipher and later in the *POSEIDON* and *POSEIDON2* family of hash functions [309, 310]. In a similar fashion, the *Marvellous* framework [311] proposed a two-step design and developed the AES-like *Jarvis* cipher and *Friday*, a Merkle-Dåmgard hash [312] that accelerate ZK STARK protocols [313] proposed for applications like the Ethereum cryptocurrency. Algebraic cryptanalysis has, in turn, found weaknesses in their structure [314]. Additional derivatives of the *Marvellous* framework include the *Vision* and *Rescue* ciphers [315] that are optimized for

[22] Authenticated Encryption with Associated Data, a form of symmetric encryption that can be achieved using a blockcipher such as AES in a carefully designed mode of operation such as Galois Counter Mode (GCM).

[23] Succinct Non-interactive ARgument of Knowledge, a type of proof system that allows a prover to demonstrate knowledge of something without revealing it using a short proof.

usage in SMPC and ZK applications and the later *Chagri* design [316] that uses MDS-based linear layer and is optimized for HE. Still, cryptanalysis in this domain is developing quickly, forcing a change in parameters of *Chaghri* [317].

Stream ciphers for HE. The development of HE-friendly ciphers has been largely motivated by the popular compressed encryption application by Naehrig et al. [318]. Combining symmetric ciphers with FHE (also known as hybrid FHE) avoids a lengthy and computationally intensive FHE operation on the client side by encrypting symmetrically the data and encrypting homomorphically the symmetric key. Subsequently, the encrypted data gets homomorphically decrypted on the server side by running the decryption circuit. At this point the server can proceed with further privacy-preserving computations on homomorphically encrypted data. This application has highlighted the need for an efficient encryption function that minimizes the overhead of homomorphic evaluation of symmetric decryption (also known as the decompression overhead).

Such an application context led to first considering the eSTREAM[24] finalist *Trivium* [319] as the underlying primitive for compressed encryption and triggered the development of *Kreyvium* [320], a 128-bit key variant of *Trivium*. Notably, both stream ciphers managed to outperform *LowMC* in this particular application context while not demonstrating the same security concerns w.r.t. interpolation attacks [306]. Following, the *FLIP* cipher design [321] aimed to combine the positive aspects of block and stream ciphers and counter their negative features. Thus *FLIP* combined the constant per block noise of *LowMC*-like designs with the low noise level achieved by stream ciphers like *Kreyvium*. Both *FLIP* and the later *FiLIP* [322] designs utilize novel filter permutators, aiming for a symmetric encryption scheme whose homomorphic evaluation of decryption is as cheap as possible w.r.t. the error growth. The designs are suitable for FHE schemes where the error growth depends on the multiplicative depth of the circuit but also FHE schemes with asymmetric error growth.

Notably the earlier design trend of ciphers with low multiplicative complexity persisted in the field of HE-friendly stream ciphers. The design of the stream cipher *Rasta* [323] expanded on *LowMC*/*MiMC* aiming to concurrently minimize the multiplicative depth and number of multiplications per bit, utilizing ASASA permutations [324] that randomize the affine layer of the cipher. Several variants followed, such as the *Dasta* [325] that uses the *Keccak* χ operation [326] for non-linearity and fixes the affine layer using BCH-based diffusion to improve performance, *Masta* [327] that extends *Rasta* to support modular arithmetic and *Pasta* [328] that aims to accelerate hybrid homomorphic encryption. Another variant, *Fasta* [329] integrates the bitslice parallelism to the cipher structure in an improved fashion to further increase performance for specific HE schemes. Similar approaches by *HERA* and *Rubato* [330, 331] aim for applications aimed for improvements in the context of approximate HE and the CKKS scheme [203].

4.5.2.1 Tailoring SMPC and HE to the DL context

As the use of SMPC and HE in the context of DL has become more prominent, the research literature has began to take a more fine-grained approach to the more challenging operations that occur in DL systems, and particularly the difficulties in executing these operations when requiring malicious security. Early works on this topic considered speeding up matrix multiplication in the semi-honest setting by replacing Beaver triples with more complex variants of correlated randomness [153], and this work was extended into the malicious security setting soon after [332]. More recent papers have shown how to perform convolutions [223]. A separate line of work has considered packing HE ciphertexts to perform (among other tasks) convolutions in parallel using (encryptions of) specially constructed matrices [333].

Most efficient SMPC protocols in the dishonest majority setting make use of pre-processing, where input-independent correlated randomness is created: for secret sharing this is random multiplication triples, while in garbled circuit protocols this is the one-time generation of a garbled circuit. Schemes are in general relatively fast when the number of parties is small, however they scale quite badly for large numbers of parties. A recent line of work has attempted to bridge this gap, by proposing optimizations and techniques that target multi-party computation where the number of computing parties is large, with the aim of speeding up both the offline pre-processing phase and the input-dependent online phase [334, 335, 336, 337, 338].

[24] A stream cipher competition ran by ECRYPT between 2004 and 2008 https://www.ecrypt.eu.org/stream/.

4.5.3 Scaling up Privacy-Preserving Federated Learning

The largest part of this section has focused on training and inference in DL systems, however FL techniques introduce security issues that are not present in ML systems trained on a single centralized dataset and these issues have been described in Section 3.3. We briefly summarize here some existing approaches to tackling the security and privacy concerns in FL[25], before providing insight into how these techniques can be scaled up to larger numbers of parties or more complex FL models. Where possible, we will indicate where the other acceleration techniques described in this section are applicable to FL.

As described in Section 3.3.3, secure aggregation has been demonstrated to be somewhat practical in the single-server setting [342, 343], i.e., where one global model is being trained by an *aggregator* and the training parties wish to conceal their own model updates (gradients) from other training parties and the aggregator. This line of work has been enhanced by strengthening the corruption models in which the privacy-preserving federated learning system operates [344, 345], and increasing the computational and communication efficiency [346, 347].

A recent line of work has focused on how to securely aggregate in the setting where there is a *distributed aggregator*, ensuring security even if a subset of the participating aggregating servers collude [348, 347, 349], in order to mitigate privacy leakage to a single aggregator [350, 351]. These works generally employ SMPC techniques both for the secure aggregation between the clients, and for combining the aggregated values into a joint model between the servers, and therefore can benefit from the acceleration discussed in this section that is focused on speeding up SMPC. Incorporating Homomorphic Encryption (HE) techniques into more of these workflows also appears to be a promising area for future research.

An area that has so far not received much attention is categorizing the types of training in FL in terms of ease of integration of PETs. It is evidently important to speed up techniques for FL while introducing additional privacy features, however the landscape at the moment is not mature enough to indicate which model types and use cases experience the lowest overheads in the privacy-enhancing context.

4.6 Outlook for Privacy-Preserving Technologies in Machine Learning

In the previous sections we have introduced the core concepts that underpin Secure Multi-Party Computation (SMPC) and Homomorphic Encryption (HE), described how they are used in existing literature and tools and then indicated where acceleration techniques can benefit components in the workflows of privacy-preserving training and inference for machine learning. The existing literature is evolving rapidly, reflecting the rapid developments in all aspects of machine learning, and this means that developments are occurring in multiple dimensions simultaneously.

We will now aim at summarizing the recent trends and indicate the inevitable trade-offs that occur, with the aim of providing a high-level overview of the current state of research in the area.

4.6.1 Hybrid Schemes

In modern approaches, it is rare for a single privacy-enhancing technology to be used, in contrast to the earlier efforts such as *CryptoNets* [162] (HE only) or *ABY3* [158] (SMPC only). Each component of the workflow can often be considered separately, with more and more published works focusing on enabling faster conversion between data representations for seamless switching between technologies.

It would be unsurprising for this trend to continue: as more protocols for individual components appear, there is more of a requirement to integrate these new protocols into other systems. Increased modularity is not just desirable from an efficiency or performance perspective: many modern protocols are proven secure in the

[25] We emphasize again that we do not focus on differential privacy in this section, which has been used to boost client privacy in federated learning, for details on this topic see e.g. [339, 340, 341] and references therein.

universal composability framework [352] [26] that ensures combinations of (possibly complex) protocols will not be less secure than their component parts.

4.6.2 Expansion of Use Cases

So far, much of the research literature has focused on performance on inference in relatively simple neural networks or other small-scale ML classification tasks, such that the conclusions drawn may not be applicable to more general ML/DL use cases. This is unsurprising: academic papers will wish to demonstrate how much their technique improves on the state-of-the-art by showing a performance increase on a somewhat equal task (and performance numbers that run in relatively short time periods, rather than hours or days).

As we have seen in Section 4.5.1, there has been an emergence of (large-scale, well-funded) projects that aim to accelerate components of PETs in multiple dimensions, bringing more complex DL tasks into play. Adequately comparing the existing approaches then runs into the problem of reproducibility, particularly in the case of custom hardware. However, with these performance increases we can expect to see privacy-preserving techniques to be added to more and more challenging ML environments to assess the current limits of the technologies.

4.6.3 Assessing Performance for the Online phase

In general, the bottleneck of SMPC and HE approaches is the performance in the input-dependent online phase. SMPC protocols based on secret sharing require low communication between the parties for each layer of multiplication gates in the circuit, and their round complexity is linear in the depth of the circuit. Therefore the bandwidth requirement is low, giving very efficient performance in a Local Area Network (LAN) setting but poor performance when high latency comes into play in Wide Area Network (WAN) scenarios. In garbled circuit protocols, the online phase consists of the parties providing their garbled inputs and run in constant rounds, but with each round being slower: this means that they perform better in the WAN setting. For approaches based on HE, the online communication is simply a single ciphertext, so the time taken by the online phase is entirely dependent on the evaluation of the desired function on that ciphertext; the choice of activation function will have a big impact on online performance, since the use of the squaring function will run much faster than (a polynomial approximation of) ReLU or Sigmoid.

4.6.4 Assessing Performance of Underlying Cryptographic Primitives

The various options and challenges presented by applications involving PETs translates to a plethora of design goals, features and metrics for the underlying cryptographic primitives that support such applications. In Table 11, we provide a list of common distinguishing features that can help to pinpoint the design choices of a cipher and assess its implications. We note that the listed features do apply to all cipher designs since the landscape of SMPC/HE/ZK-friendly primitives is fairly heterogeneous and fragmented in order to fulfill various criteria. More importantly, we stress that many listed features are of a qualitative nature and cannot be directly translated to quantitative metrics, unless analyzed in a strictly specified context.

[26] According to Canetti et al. universally composable definition of security is " that they guarantee security even when a secure protocol is composed of an arbitrary set of protocols, or more generally when the protocol is used as a component of an arbitrary system" [352].

Design Goals	Description
Accelerated Method	SMPC, HE, ZK or their combination
Accelerated Application	Specific application or scheme such as BGV, CKKS (approximate HE), compressed encryption, SNARKs, cryptocurrencies, etc.
Design Features	**Description**
Cipher Type	SPN-based or stream-based
Non-linear Layer	Lightweight sbox, AES-like sbox, partial substitution, *Keccak* χ, NLFSR, filter permutator
Linear Layer	Random/fixed, MDS matrix, BCH code, rotation-based, ASASA
Design Strategy	*HADES* P-SPN, *Marvellous* 2-step
Arithmetic	Support for $GF(2), GF(p)$ or $GF(2^m)$ schemes
Mode of Operation	Supported modes like sponge, Feistel, etc.
Bitslicing	Support for batch operations
Metrics	**Description**
MC	Multiplicative complexity (number of multiplications)
ANDdepth	Multiplicative depth of circuit
ANDs per bit	Number of multiplications needed per encrypted bit

Table 11 – Design features of PETs-friendly cryptographic symmetric primitives.

4.6.5 Related State-of-the-Art Gaps

Finally, based on the section analysis, in Table 12 some preliminary State-of-the-Art Gaps have been identified.

Challenge Gap ID	Description	Flows	Related SECURED Component(s)
SoTA-GAP-01	Non Linear Operations are hard to be realized in FHE Schemes	Processing	SMPC Engine
SoTA-GAP-02	Oblivious Transfer or Secret Sharing Schemes have high Communication delay	Processing	SMPC Engine
SoTA-GAP-03	Hardware accelerators for FHE schemes require extremely high number of resources and chip covered area	Processing	SMPC Engine
SoTA-GAP-08	Efficient sliced implementations of SMPC/FHE-friendly ciphers in software platforms	Processing	SMPC Engine

Table 12 – SMPC main State-Of-the-Art Gaps

5 Synthetic Health Data Generation

Data is central to all areas of research, including health. However, access to data in health care is tightly controlled, as the procedures to access them is usually slow as ethical approvals have to be obtained. Moreover, this kind of data is not accessible for the general public given the ethical concerns that this might raise, because of the lack of contact with the data providers, concerns over data security, re-identification and the lack of control of the patients over their data [353]. These facts limit the innovation, development and efficient implementation of new research, products, services or systems that can be potentially beneficial for the general public.

Synthetic data can be a solution for this problem. Synthetic data is defined as "microdata records created by statistically modeling original data and then using those models to generate new data values that reproduce the original data's statistical properties." This definition highlights the strategic use of synthetic data because it improves data utility while preserving the privacy and confidentiality of information [354].

Notice that this function can be well covered with the open datasets available, which are usually well curated and strongly deidentified. Section 5.1 shows a collection of some of these datasets. Also, there are new possibilities to use the original data, while keeping them secured in the premises of the data provider, e.g., a hospital. These techniques are a good solution but are not perfect as they require security measures that might hinder the accuracy of the data, as mentioned in previous sections.

In this sense, synthetic data is one of the many innovative ways to allow organizations to share datasets with broader users, minimizing the need to access real personal data and complementing both anonymization and secure data usage approaches like Federated Learning. Moreover, this approach is not limited to the possibility of sharing data, but it is also relevant to do *Data Augmentation*, i.e. the ability to enlarge the dataset by generating new data. In this way, medical institutions can share their anonymized data or provide a framework to use them without sharing and also include more synthetic data to complement their data. This is specially important for Deep Learning based techniques as the more complex the model is, the higher the number of parameters that needs to be trained, and, therefore, the more data is needed. For these reasons, we explore the literature and remark the utility of synthetic data in the health care domain.

In the following sections we describe a review of the State of the Art in synthetic data generation regarding four kinds of data: Medical imaging, time-series, genomics and electronic Health Records. We particularly focus on the necessities of the SECURED project. The review is driven by the following research questions, addressed in the respective sections:

1. Which kind of synthetic data is relevant for the SECURED project? Is there any open/accessible dataset for each kind of data modality? (Section 5.1)
2. Which techniques have been tested for each particular data modality? (Section 5.2)
3. Is there any existing software or tool that provides this functionality? (Section 5.3)
4. How are the methods evaluated? Which technologies or methodologies are promising for each data type and which are the gaps in the State of the Art? (Section 5.4)

5.1 Data types and Data profiles

In this section we consider the first two research questions. First, which kind of data generation is needed for SECURED and then which kind of open data we can find related to it.

In Table 13, we can observe the types of data that could be available in the use cases/pilots of the SECURED project. This reference is used to drive this section for the selection of the State of the Art manuscripts. In particular, we select the four main data types: Images, time-series, genomic data and electronic health records. Notice that there are some data modalities already present in the table, but as this list might not be definitive, we explore modalities that are outside of this list. In terms of research of synthetic data, there is a wide amount of research performed for image data which will be discussed later in this work. Time-series are also reviewed

Type of data	Data modality
Image	X Ray (Mammographies...)
	Digital tissue images (colon, liver, lung, mamal, brain...)
	MRI (spine, nervous system, brain...)
Time-series	Electrocardiogram (ECG)
	Cardiotocogram (CTG)
	Oxigen saturation
	Other respiratory variables
Genomics	DNA SNPs
Tabular	Electronic Health Records

Table 13 – Summary of the types of data and modalities that could be available for the project.

showing that this research area is not that well explored. Finally, we also include a brief information about genetic/genomic data, as our prior knowledge on this kind of data is scarce and performing a deep analysis of the synthetic data generation in this particular field requires wide expertise in the use cases. All of this is present in the following sections. On the other hand, tabular data is not covered as there are many recent reviews regarding this type of data that cover our requirements, like Hernandez et al. [355] and Yan et al. [356].

Apart from collecting the data that could be available through the use cases of the project, we have also included the results of an initial compilation of the most commonly used open datasets for the different types of data mentioned in Table 13. This will be of great utility for the initial phases of the project. We will be able to start the analysis of the available methods in the literature while the data of the pilots is being gathered. In addition, making use of the open data will help to better validate the results of our research outcomes and make them reproducible by the research community. We introduce the collected open data in the following subsections.

5.1.1 Images

One of the main sources of data in the medical domain are images. The availability of diverse and well-annotated datasets plays a crucial role in ensuring the accuracy and generalizability of generated medical images. Recent advancements in this field have led to the creation of various datasets that cater to different modalities, anatomical regions, and medical applications. Even if medical images are hard to obtain due to their privacy issues [357], we summarize some notable datasets.

Datasets are generally designed for a specific medical application. As such, these datasets can be categorized based on the modality of the images.

For MRI the body part are diverse: brain, cochlea, pelvic region, knee, prostate:

- **BraTS** (Multimodal Brain Tumor Segmentation) [358]: focuses specifically on brain tumor segmentation and provides multi-modal MRI scans, including T1-weighted[27], T1-weighted with contrast, T2-weighted, and Fluid attenuated inversion recovery (FLAIR) sequences. This dataset enables the development and evaluation of synthetic image generation techniques for brain tumor analysis.

- **ADNI** (Alzheimer's Disease Neuroimaging Initiative) [359]: The ADNI dataset includes neuroimaging data, including MRI scans, from individuals with Alzheimer's disease, mild cognitive impairment, and healthy controls. It facilitates the study of synthetic image generation for early detection and monitoring of Alzheimer's disease.

- **CrossMoDA** 2021 [360]: contains MRI images of type T1, T2 designed to perform cochlea segmentation.

[27] There are different types of contrast images in Magnetic Resonance Imaging (MRI), T1-weighted MRI which enhances the signal of the fatty tissue and suppresses the signal of the water and T2-weighted MRI which enhances the signal of the water.

- **Gold Atlas male pelvis dataset** [361]: consists of MRI images (T1 and T2) and CT images of 19 male patients over the pelvic region.
- **fastMRI** [362]: deidentified imaging dataset comprises raw k-space (the 2D or 3D Fourier transform of the image measured) in several sub-dataset groups: knee, brain, prostate.

X-ray datasets (mainly from chest):

- **ChestX-ray14** [363]: The ChestX-ray14 dataset consists of over 100.000 frontal-view chest X-ray images with associated radiologist-labelled annotations for 14 common thoracic pathologies. It serves as a benchmark dataset for synthetic image generation in chest X-ray analysis and aids in the development of automated diagnostic systems.
- **NODE21** [364]: frontal chest radiographs with annotated bounding boxes around nodules. The images come from other datasets and have been labelled: JSRT [365], PadChest [366], Chestx-ray14 [363], Open-I [367].
- **CheXpert** [368]: large dataset of chest X-rays, with more than 224 chest radiographs, which features uncertainty labels and radiologist labelled reference standard evaluation sets.

Mammography datasets (more abundant than other modalities):

- **INbreast** [369]: Mammographic database that includes several types of lesions (masses, calcifications, asymmetries and distortions) for 115 cases.
- **OPTIMAM**[370]: image database that contains mammography images and associated clinical and pathological information. It contains over 2.5 million images from 1.7 million women. It includes normal breasts, benign findings, screen-detected cancers and interval cancers.
- **BCDR** (Breast Cancer digital Repository) [371]: contains mammography and ultrasound images, clinical history, lesion segmentation and selected pre-computed image-based descriptors. They have been labelled by specialized radiologists.
- **CBIS-DDSM** (Curated Breast Imaging Subset of Digital Database for Screening Mammography) [372]: a database of 2.620 scanned film mammography studies. It contains normal, benign, and malignant cases with verified pathology information. Updated ROI segmentation and bounding boxes, and pathologic diagnosis for training data are also included.
- **CSAW** (Cohort of Screen-Aged Women) [373]: images from mammography screening with more than 1 million examinations. Also has manually annotated labels and metadata for the patients.

RGB imaging:

- **REFUGE** (Retinal Fundus Glaucoma Challenge) [374]: contains retinal fundus images for the diagnosis and analysis of glaucoma. It provides a diverse set of images with various degrees of glaucoma severity.
- **DRIVE** (Digital Retinal Images for Vessel Extraction) [375]: retinal images collected from diabetic retinopathy patients.
- **STARE** (STructured Analysis of the Retina) [376]: contains retinal images together with expert annotations that consist of diagnoses for each image, blood vessel segmentation, artery/vein labelings, optic nerve detection.
- **ISIC** [377]: The dataset contains dermoscopic images of unique benign and malignant skin lesions from over 2000 patients. Diagnoses have been confirmed using expert agreement or histopathology.
- **PH2** [378]: is a dermoscopic image dataset that allows both segmentation and classification algorithms.

- **HyperKvasir** [379]: Contains real gastro and colonoscopy examinations with partly labels by experienced gastrointestinal endoscopists. The dataset contains 110.079 images and 374 videos where it captures anatomical landmarks and pathological and normal findings, giving in total around 1 million images and video frames.

Furthermore, some datasets contain multiple data modalities:

- **MSD** (Medical Segmentation Decathlon) [380]: comprises ten different medical imaging challenges covering various tasks such as brain tumor segmentation, liver segmentation, and cardiac segmentation. It provides a large-scale benchmark for evaluating the performance of synthetic image generation models across different anatomical structures and modalities.
- **UK Biobank** [381]: a large-scale biomedical database and research resource, containing in-depth genetic and health information from half a million UK participants that contains data or the study of diseases such as cancer, heart disease, stroke, obesity, genomic anomalies, mental health.

These datasets, among others, provide valuable resources for training and evaluating deep learning models in synthetic medical image generation tasks. They contribute to the advancement of research in medical imaging and assist in the development of reliable and accurate diagnostic tools for various medical conditions.

5.1.2 Time series

Apart from images, the medical domain also contains a large volume of time-series data. Good examples are electrocardiograms and electroencephalographs, very valuable for the study, analysis and diagnosis of multiple health problems.

A time series is a set of samples of data obtained in different moments of time. Figure 4 shows a group of twelve series with different forms and patterns. Usually, time series are divided into four distinct components: level, trend, seasonality and noise. The first three components are typically referred to as systematic components and, on the other hand, the noise is characterized as the non-systematic component. In this way, we can view a time series as a composition of systematic components with added noise. The level is the average value of the samples from the time series and the trend is the change between one sample and the consecutive one along all the samples of the series. Regarding the seasonality, it describes the cyclical behaviours that can be appreciated consecutively repeated in the overall trace [382].

As previously introduced, we have included a compilation of the most commonly used datasets in the literature, focusing only on the medical data that is open for its usage in research. For creating a more practical summary, we have incorporated Table 14 that lists the different types of available time series. In this way, it is easier to find the datasets that are more helpful for specific use cases. Moreover, Table 15 shows the full list of datasets along the linkage to their corresponding types of series that they incorporate. It is worth mentioning that we have only included the main types of traces, although additional subtypes and tabular data can be found in some of these datasets.

Both tables will be expanded and improved over the course of the project. It is expected to include a considerable number of new datasets and to severely enhance the current taxonomy and ordering. The next list includes a brief definition of the different initials corresponding to distinct biological signals that appear in the tables:

- Electrocardiogram (ECG): electrical signals in the heart, useful for detecting heart problems and monitor the heart's health.
- Fetal Electrocardiogram (FECG): electrical signals in the heart of the fetus, useful for fetal monitoring during pregnancy.
- Electrooculography (EOG): corneo-retinal standing potential that exists between the front and the back of the human eye, useful for analysing the eye's health and for understanding its behaviour.

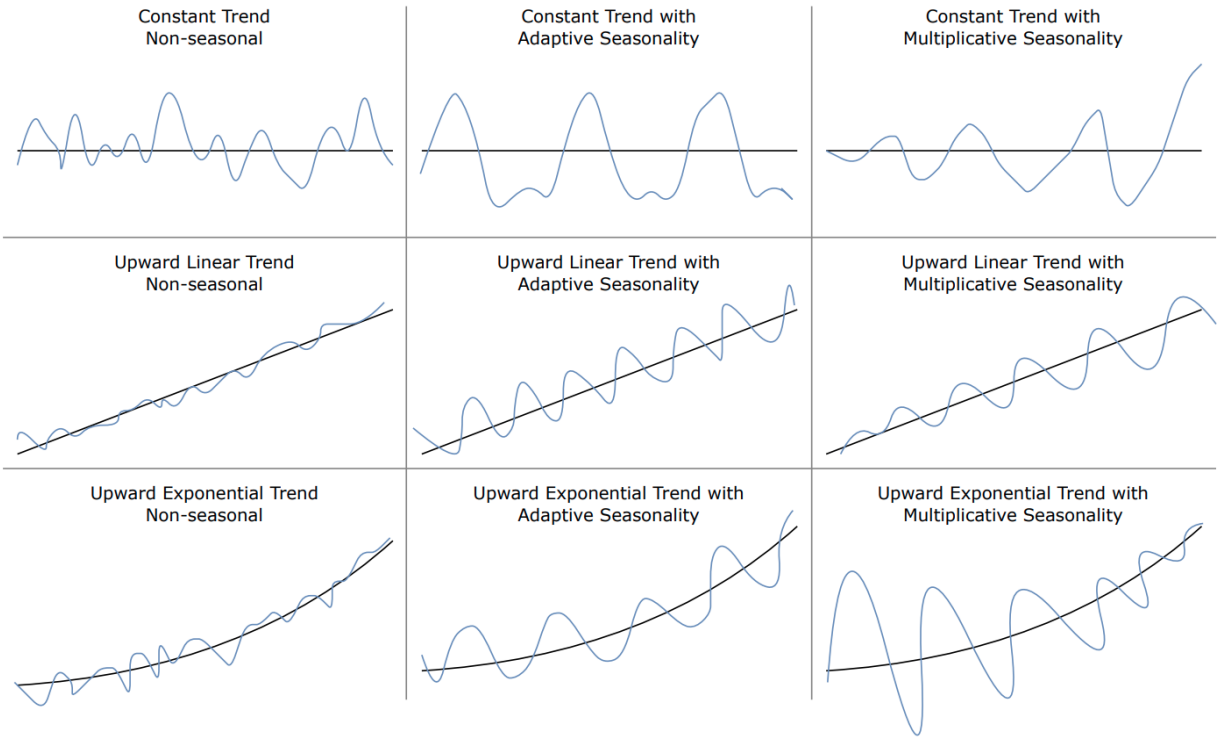

Figure 4 – A group of different time series with distinct level, trend and seasonality.

- Electroencephalography (EEG): electrical activity in the brain, useful for studying and analysing the cognitive processes.
- Magnetoencephalography (MEG): magnetic activity in the brain, mainly useful for research about the functions of the brain.
- Electrocorticography (ECoG): similarly to EEG, electrical activity in the brain, although ECoG is recorded through sensors located inside the skull, in the cerebral cortex. It is used to confirm the location and extent of epileptic tissue for brain surgeries.
- Electromyography (EMG): muscle response to a nerve's stimulation of the muscle, useful for detecting neuromuscular abnormalities.

Apart from this, we include the following list that briefly introduces all listed datasets for a general presentation of their purpose and origin.

- eICU Collaborative Research: compilation of vital sign measurements from admissions to intensive care units (ICU) across the United States. It comprises over 200 thousand patient unit encounters for over 139 thousand unique patients [383]. It can be found through this link at the Physionet repository [384, 385].
- MIMIC-III (Medical Information Mart for Intensive Care): compilation of measurements from patients admitted to critical care units at a large tertiary care hospital, containing data associated with 53.423 distinct hospital admissions for adult patients and 7.870 neonates [386]. Past versions of the dataset are also used in the literature. It can be found through this link at the Physionet repository [387, 385].
- Alcoholism EEG: compilation of EEG signals of 122 subjects to study EEG correlations to genetic predisposition to alcoholism. The samples come from the State University of New York Health Center, United States. It can be found through this link at the UCI Machine Learning Repository [388].
- Diabetes: compilation of measurements performed to subjects suffering from diabetes and healthy subjects. The samples come from the National Institute of Diabetes and Digestive and Kidney Diseases, United States. The dataset can be found through this link at the UCI Machine Learning Repository, which

type of data	subtype	identifier
Breath	CO_2 concentration	1
	exhalation measurements	2
Blood	pressure	3
	glucose %	4
	insuline %	5
Heart	ECG	6
	FECG	7
Head	EEG	8
	EOG	9
	MEG	10
	ECoG	11
Other	vital signs	12
	EMG	13

Table 14 – Compilation of time series data types that are used in the reviewed works concerning open medical time series

dataset	patients	traces type	source[28]
eICU Collaborative Research	139.000	[12]	US
MIMIC-III	46.467	[12, 6]	US
Alcoholism EEG	122	[8]	US
Diabetes	70	[3, 4, 5]	US
MIT-BIH Arrhythmia	47	[6]	US
BCI Competition IV	30	[8, 9, 7, 11]	GER/AUS/US
Sleep-EDF	22	[8, 9, 13]	NTH
Siena Scalp EEG	14	[8, 12]	ITA
Breath Metabolomics	4	[1, 2]	SWI
UCR TS Classification Archive	N/A	[6, 12, 9, 13, 7]	N/A

Table 15 – Compilation of the available open datasets used in the reviewed works that include medical time series ordered by their number of patients. Unfortunately, the number of patients is quite arbitrary and not a good option for the ordering approach. The real value of the dataset heavily relies, between others, on the number of time series per patient, the length of each time series, the sampling rate, the number of null values and the general quality of the data.

was used in 1994 at the AAAI Spring Symposium on Artificial Intelligence in Medicine. It can be found as well, with other characteristics, at the Kaggle repository through this link [388].

- MIT-BIH Arrhythmia: compilation of ECG recordings obtained from 47 subjects. The records are manually annotated by two or more cardiologists for research into arrhythmia analysis and related subjects. The samples come from Boston's Beth Israel Hospital, United States [389]. It can be found through this link at the Physionet repository [385].

- BCI Competition IV: competition containing a set of four datasets compiling single-trials of spontaneous brain activity with the goal of validating signal processing and classification methods for Brain-Computer Interfaces (BCI) [390, 391, 392, 393]. Datasets from past competitions are also used in the literature. It can be found through this link.

- Sleep-EDF: compilation of 197 whole-night PolySomnoGraphic sleep recordings, containing EEG, EOG, chin EMG, and event markers. It contains sleep patterns labels that were manually scored by well-trained technicians. The samples come from the Leiden University Hospital, Netherlands [394]. It can be found

through this link at the Physionet repository [385].

- Siena Scalp EEG: compilation of EEG recordings from 14 patients for the diagnosis of epilepsy and the classification of seizures. The samples come from the Unit of Neurology and Neurophysiology of the University of Siena, Italy. It includes labels performed by an expert clinician after a careful review of the clinical and electrophysiological data of each patient [395]. It can be found through this link at the Physionet repository [396, 385].
- Breath Metabolomics: collected data from breath metabolomic experiments in a pilot study with four subjects. The samples come from the University Children's Hospital Basel, Switzerland. It can be found through this link at the UCI Machine Learning Repository [388].
- UCR Time Series Classification Archive: a huge collection of datasets for time series classification that includes ECG, EOG, FECG and EMG signals. The dataset is widely used in the general domain of time series classification. The specific datasets that contain medical data are the following: CinCECG-Torso, ECG200, ECG5000, ECGFiveDays, EOG, MedicalImages, NonInvasiveFetalECGThorax, Semg-Hand and TwoLeadECG. The origin of the samples of each dataset requires further research. The dataset can be found through this link [397].

5.1.3 Genomics

Since the success of the Human Genome Project [398], the technology has become more efficient and cheaper, so that DNA sequencing has become more accessible. With the availability of this data, its importance has been recognized and research has been pushed towards its understanding and use. Nowadays, this kind of data is used in fields like healthcare.

Naveed et al. [399] mention the high impact of genomic data but also that there are many risks when sharing it as in further research we discover new information that can be found in the genome. This study shows that re-identification is a well explored topic and even attacks to Machine Learning models knowing part of the background information of the person. For example, Lin et. al [400] show that with just 75 independent SNP (Single Nucleotide Polymorphism) it is possible to identify an individual. This is also an issue in more restricted environments where only specific queries are replied to but the data is not shared, like it is shown in the work of Shringarpure and Bustamante [401]. Hence, publishing the sequences individually is a risk for the donor. The re-identification approaches and the attacks on the Machine Learning models raises concerns about sharing genetic data, particularly because we still do not know all that we can find in the genomes, as mentioned before.

Given these risks, there has been efforts towards de-identifying this data, like the approach of Ziegenhein and Sandberg [402]. Still, Bernier et al. [403] hold that the de-identified data is not anonymous, however in the legislation of some countries like Canada, the threshold to consider data anonymous is not "zero risk".

Genomic synthetic data offers a trade-off between privacy of the samples and utility. By generating fake patients statistical analysis of the genomes can be performed. For example, hiding a sensitive part of the genome might not be enough as high-order correlation models can be exploited [404]. Therefore, this approach is a promising one.

Some examples of open and controlled data are available in the following list:

- **1000 Genome Project**: A project that ran from 2008 and 2015 to create a large repository of human variation and genotype data. The data is accesible by their FTP and contains 2.504 individuals from 26 populations. Available in this link.
- **International Human Epigenome Consortium data portal** [405]: A consortium stablished in 2012 in Canada to support large-scale human epigenome mapping. Currently they have over 7.000 different donors. Part of the data is accessible through the website, but other data requires approval to access.
- **Harvard Personal Genome Project**: A project started in 2005 which aims to provide genome sequencing data to boost medical research. Currently they have over 5.000 donors. The data is available on their website.

- **DREAM challenge datasets**: The ICGC-TCGA DREAM Genomic Mutation Calling Challenge was a challenge launched to improve the standard methods for identifying cancer-associated mutations. This challenge had two different datasets, the real one (4TB) that requires permission to download and a synthetic one (2TB). More information can be found in the following link.
- **UK Biobank** [381]: As presented in the images section, the UK biobank is a large repository that also includes genomic data, however the access is strictly regulated.
- **GEO**: The Genomics Expression Omnibus is a public functional genomics data repository. It currently hosts 4.348 datasets with over 200.000 series. All the data is available and the instructions to download them are in their website. Notice that this data can be downloaded directly using software like R using the bioconductor packages.

Notice that other repositories like HapMap have been shutdown because of security issues.

5.2 Data Generation Techniques

5.2.1 Images

Image generation across medical imaging modalities is an active area of research. This synthetic generated data has the potential to enable faster research on model development and, in the medical educational field, alleviate cost associated with obtaining new data samples.

Note that it is often challenging to obtain high quality, balanced datasets with labels in the medical domain. Medical images are mostly imbalanced and time-consuming to obtain their labels, and contain private data. To overcome these issues, several studies exploit generative models to increase the size of the training set by artificially synthesizing new samples. This process is often referred to as data augmentation, and it is a very popular technique in computer vision.

Given the rapid progress in the fields of machine learning and computer vision over the last two decades, image synthesis is now viable and has a growing number of exciting applications. Deep learning, as a broad subdiscipline within machine learning and artificial intelligence, has dominated the field for the past several years [406].

Deep learning methods use neural networks to extract useful features of images. In the context of image generation, these methods usually share a common framework that uses a data-driven approach. First, the training is performed as regularly done, and then the model is ready to be used for prediction. In this case, what is obtained from prediction is a new generated image.

In this study, we review deep learning methods for data augmentation, and classify them by the taxonomy of the used architectures. We can manly distinguish between three differentiated architectures. Furthermore, since the models are application specific, we will also group them by imaging modality.

Further comprehensive information and in-depth analysis of different methodologies can be explored in peer-reviewed surveys [407, 408, 409, 406, 410, 411].

The taxonomies of the reviewed studies can be grouped into three categories: Variational Autoencoders (VAEs) [412], Generative Adversarial Networks (GANs) [413] and Diffusion models (DMs) [414]. These three groups are not completely different from each other, but represent increases in architecture complexity. For example, VAEs are a type of network that can act as a basic component in advanced architectures such as GANs or Diffusion. Furthermore, an extension of the VAE network, called U-Net [415] has been used as a backbone for the generative part both in GANs and Diffusion Models. Although Transformers [416], another deep learning architecture, have been used for medical imaging [417] we are not aware of any work that uses them for unconditioned synthetic image generation.

5.2.1.1 VAEs

VAEs extend the basic Autoencoder (AE) architecture by incorporating probabilistic modeling techniques. Similarly to AEs, VAEs have an encoder, typically based on convolutional layers. Contrary to AEs, VAEs encode the input data into a latent space distribution, typically a multivariate Gaussian distribution, rather than a fixed point. This distribution is defined by mean and variance parameters, learned during the training process. VAEs simultaneously optimize two losses: the per-pixel reconstruction loss, the same used in AE and the regularization loss that ensures that the latent variable follow a normal distribution. The generative aspect of VAEs comes into play during the decoding phase, where random samples from the latent space distribution are fed to the decoder to generate new data points. By sampling from the latent space, VAEs allow for the creation of diverse and novel data samples that capture the underlying characteristics of the training data. The latent space serves as a continuous and structured representation of the input data, enabling interpolation and smooth transitions between samples. VAEs are known for their ability to capture the underlying data distribution, handle missing or incomplete data, and enable interpolation and exploration in the latent space. They outperform other generative methods in terms of output diversity and easier training. However, they tend to produce blurry output images due to the regularization loss, and this is one of the reasons why they have received less attention than other generative models such as GANs.

However, works do not use VAEs in this basic form, but add variants that improve the quality of VAE-generated data.

- U-Net [415]: a type of convolutional autoencoder that was designed to perform semantic segmentation by adding skip connections.
- Inverse autoregressive flow [418]: rather than mapping a Gaussian distribution, this model introduces a more flexible approach by using an autoregressive model to transform a simple distribution (e.g., Gaussian) into a more complex distribution. This transformation is performed in reverse during the decoding process and allows the VAE to capture more complex and structured latent space representations.
- InfoVAE [419]:incorporates an additional information bottleneck into the latent space. By introducing a regularization term in the VAE's objective function that maximizes the mutual information between the latent variables and the input data. It leads to better interpretability and control over the generated samples.
- VQ-VAE2 [420] (Vector Quantized Variational Autoencoder 2): it introduces a discrete latent space. To learn it, it adds a vector quantization objective, where each codebook represents a unique prototype. This model captures discrete structures in the data and produces high quality reconstructions.
- CVAE [421] (conditional VAE): conditions the generation process on additional information, such as class labels or attributes.
- VAE-GAN [422]: combines the power of GANs and VAEs, further details the next section.

An overview of study use cases where VAEs have been used in the medical field can be found in Table 16. Variations to cater to specific applications have been made, and some of the most successful applications use the VAE-GAN [423, 424] architecture.

Citation	Method	Dataset	Body part	Measures
MRI				
[425]	ICVAE	Private	brain	segmentation
[426]	CVAE	OpenfMRI, HCP, NeuroSpin, IBC	brain	classification
[427]	GeometryVAE	ADNI, AIBL	brain	classification
[424]	IntrospectiveVAE	Private	brain	classification
[428]	RHVAE	OASIS	brain	classification
[429]	MM-VAE	UK Biobank	heart	MSE, MMD
Ultra sound				
[425]	ICVAE	Private	Ultra sound spine	classification
[423]	VAE-GAN	Private	Ultra sound thyroid	segmentation, SSIM
Others				
[430]	AL-VAE	Private	OCT	segmentation, Wasserstein distance
[431]	DM-VAE	Private	Otoscopy	tympanometry measurements

Table 16 – VAE-based works for medical augmentation, divided by imaging modality and including architecture, dataset and measures used to asses performance.

5.2.1.2 Generative Adversarial Networks (GANs)

In GANs, two neural networks are trained jointly in a competitive manner: the first network (generator) generates synthetic data, and the second network (discriminator) is trained to distinguish between real and the synthetic data generated by the first network. This process is called adversarial training, where the generator and discriminator play a min-max game, with the generator striving to produce increasingly realistic samples and the discriminator attempting to improve its discriminative ability. Given that the generator is a generative model, a VAE can be used to produce the generated samples, and the GAN setup can be seen as a VAE with an extra loss term, which is the adversarial loss. Once trained, new data points can be synthesized by feeding the generator with a noise sample. GANs excel in generating visually compelling and highly realistic samples. Nevertheless, GANs also have their drawbacks, such as learning instability, difficulty in converging, and suffering from mode collapse [432].

To address this challenges, several variations of GANs have been proposed:

- WGAN [433] (Wasserstein GAN): replaces the Jensen-Shannon divergence as in the original GAN formulation by a Wasserstein distance. It leads to more stable training than original GANs with less evidence of mode collapse, as well as meaningful curves that can be used for debugging and searching hyperparameters. In practice, the downside of the WGAN is its slow optimization.
- CGAN [434] (conditional GAN): adds a conditioning variable to the latent vector in the generator, allowing for more control over the generated samples and partially mitigating mode collapse.
- pix2pix [435]: introduces a conditional generator to learn to translate images from one domain to another by replacing the traditional noise-to-image generator with a U-Net, a type of VAE that adds skip connections.
- DCGAN [436] (deep convolutional GAN): In this model, both the generator and discriminator follow a set deep convolutional network architecture, exploiting the efficacy of spatial kernels and hierarchical feature learning. Concepts such as Batch-Normalization and Leaky-ReLU have been included to improve training

stability and increase the resolution of synthesized images, but issues such as mode collapse were not entirely resolved.

- PGGAN [437] (progressive growing GAN): The key idea behind PGGAN is to gradually increase the resolution of both the generator and discriminator as the training progresses. The training begins with a low-resolution generator and discriminator and then progressively adds new layers to model finer details. This progressive growth allows the model to capture high-resolution details and generate higher-quality images.
- CycleGAN [438]: learns the mapping from a source domain to a target domain and vice versa, without the need for explicit correspondences between individual samples. It introduces a cycle consistency loss, which enforces that translating from one domain to another and back should result in the original input.
- ACGAN [439] (auxiliary classifier GAN): adds an auxiliary classifier in the discriminator that predicts additional class labels associated with the samples. It enables both the generation of realistic samples and the control over the generated samples' class attributes.
- VAE-GAN [422]: uses VAE as the backbone for the GAN generator. It replaces the VAE reconstruction error term with a reconstruction error expressed in the GAN discriminator.

Next, we study use cases where GANs have been used in the medical field. Table 17 shows a summary of relevant studies. Several variations from the previously mentioned have used to generate synthetic images. Most of the works focus on a secondary task to evaluate the generations.

5.2.1.3 Diffusion models (DMs)

Recently, Diffusion models (DMs) have demonstrated promising ability to generate realistic and diverse outputs [457, 458]. This type of model is based on the diffusion process: they learn the progressive mapping from noise to the actual data distribution. The training consists of two processes: a forward diffusion process that gradually adds noise to the input and a reverse denoising process that learns to generate data by denoising. In the models proposed, a U-net backbone is used to learn the progressive reverse denoising. By the end of training, the model is able to map a noise input to an initial data point, hence, similarly to GANs and VAEs new images can be synthesized by sampling a random noise vector. One of the drawbacks of DMs is their high computational cost and huge sampling time.

To combat the issues, researchers have proposed several variants of diffusion models that aim to improve the sampling speed while maintaining high-quality and diverse samples:

- Progressive distillation: distils a trained diffusion model that contains many steps into a new diffusion model that takes half as many sampling steps.
- FastDPM (Fast Diffusion Probabilistic Model): uses a modified optimization algorithm to reduce the sampling time and introduces a concept of continuous diffusion process.
- DDIIM (Denoising Diffusion Implicit Model): implements a non Markovian diffusion process, to speed up the sampling process.
- DM-GAN (denoising diffusion GAN): a combination of DM and GAN that has shown to provide high-quality and diverse samples at a much faster sampling speed (by a factor of 2000).

Another drawback that has been recently found is that diffusion models tend to memorize training data. Carlini et al. [459] show in their work that due to the large amount of parameters, Diffusion Models have a tendency to not generalize enough so that similar data to the one provided by the training can be extracted, leading to the discussion that the model may be just doing interpolation between the training images. This is still an underexplored area of research.

Citation	Method	Dataset	Body part	Measures
MRI				
[440]	LAPGAN	Private	brain	human observer, inception score
[441]	semi-coupled GAN	Private	heart	classification
[442]	WGAN	BraTS	brain	human observer
[443]	DCGAN	BLSA	brain	human observer
[444]	DCGAN	BraTS, HCP	brain	segmentation
[445]	PGGAN	private	brain	segmentation
[424]	StyleGAN	private	brain	classification
CT scans				
[446]	PGGAN	private	mammography	Inception score, FID, SSIM
[447]	DCGAN /ACGAN	M30	liver lesion	
[448]	CycleGAN	NIHPCT		segmentation
[449]	pix2pix	BraTS, ADNI	brain	segmentation
X-Ray				
[450]	DCGAN	Private	chest	classification
[451]	DCGAN	NIH PLCO	chest	classification
[452]	DCGAN	NIH PLCO	chest	classification
[453]	WGAN+infoGAN	CellDetect	bone marrow	classification, segmentation
Fundus imaging				
[454]	CGAN	DRIVE	retina	segmentation
Dermoscopy				
[455]	LAPGAN	ISIC	skin	JS divergence, MAE, MSE
[456]	CatGAN+ WGAN	ISIC, PH2	skin	classification

Table 17 – GAN-based works for medical augmentation, divided by imaging modality and including architecture, dataset and measures used to check performance.

Table 18 summarizes works that apply diffusion models to generate synthetic data. This works are very recent, but show promising results, and they represent the increasing attention these type of models have gained in the medical image field.

5.2.2 Time series

The domain of time series has received less attention by the research community in comparison with the enormous push in contributions from the disciplines of computer vision and natural language processing. This is clear when taking a look at the low amount of publications concerning time series, and specifically, regarding the generation of medical traces. Nevertheless, the drastic revolution of deep learning has allowed to address more complex problems with far more powerful models and techniques. In the last couple of years, plenty of new contributions have appeared using these developments in the field of time series generation: many are adaptations of models and methods from the domains of computer vision and natural language processing.

Time series data has some characteristics that require special attention when creating new techniques. The

Citation	Method	Dataset	Body part	Measures
MRI				
[460]	CLDM	UK Biobank	brain	FID, SSIM
[461]	IITM-Diffusion	BraTS	brain	segmentation
[462]	brainSPADE	SABRE, BraTS	brain	segmentation
CT scans				
[463]	DDPM	ADNI, LIDDC-IDRI	chest	segmentation
X-Ray				
[464]	LDM	CXR8	thoracic	classification
Histopathology				
[465]	MF-DPM	TCGA	brain	classification
Dermoscopy				
[466]	DALL·E 2	Fitzpatrick	skin	classification

Table 18 – Diffusion-based works for medical augmentation, divided by imaging modality and including architecture, dataset and measures used to asses performance.

time dimension is of great importance, and the shape and pattern usually take more significance than the individual values of the time points. Some time series tend to stretch through large fractions of time, making the capture of shapes and patterns a quite complicated task. Nowadays, the majority of methods work with multivariate series, where each point in time comes along a set of features. Nevertheless, univariate traces are still present in the industry in plenty of places.

The generation of time series is applied in the literature for the use cases of data augmentation, missing values imputation, data denoising and anomaly detection. Data augmentation is the main field of research and contributions. In this way, the vast majority of surveys are focused on this specific use case [467, 468, 469, 470]. It is worth pointing out the works from Kenji, et al.[469] and Wen, et al. [470] that present a handy and effective taxonomy that characterizes the existent contributions along a list of examples for each specific category. The survey from Kenji, et al. is the only work, to our knowledge, that empirically compares an extensive set of methods from the literature[29]. They are evaluated on the final use case of data augmentation in a time series classification set up. Unfortunately, the work performs only a comparison of classical methods, it does not test any deep learning model.

Regarding the generation of time series, without focusing on a specific use case, we have only found one survey coming from the team of Brophy, et al. [471]. The work compiles, in a well written manner, the contributions in regard to the generation of time series with GAN. In addition, it highlights some interesting contributions that tackle differential privacy and privacy preservation. Section 5.4 includes an extended description of the evaluation methodologies, including the topic of privacy, of substantial importance in the domain of health data.

On the other hand, concerning medical traces directly, one more time we have only found one survey from Lashgari et al. [467]. It comprises the multiple contributions about data augmentation of EEG traces. Unluckily, the survey is specific exclusively to this type of traces.

With all the prior, is interesting to highlight the gap in the literature in regard to surveys of time series generation. Would be of great interest for the research community, an empirical survey comparing the available methods to generate medical time series, taking special emphasis on deep learning models and using an extensive list of different types of medical traces.

[29] indeed, it provides an open implementation for all them that can be found through this link

5.2.2.1 Taxonomy

To summarize and categorize all methods of the literature, we have decided to merge the proposed taxonomies from the works of Kenji, et al. [469] and Wen, et al. [470] previously cited. Emphasizing the good pieces of both of them. We present a new division for the most general category, thinking that it improves the legibility and the classification of the available techniques. We will highlight one or two contributions of every main division, putting the focus only in the ones related to medical traces. For an extended list of related papers, please consult the aforementioned surveys.

Table 19 exhibits the suggested taxonomy. The first division splits the methods into three different families: *per time series approaches*, *mixing approaches* and *global approaches*. The second division splits the techniques into multiple domains. We may introduce new nested divisions over the course of the project, specially for the case of the *global approaches* which have received the most attention lately.

Family	Domain	Medical examples
Per Time Series Approaches	Magnitude	[472]
	Time	
	Frequency	
Mixing Approaches	Magnitude	[473]
	Time	
	Frequency	
Global Approaches	Decomposition Models	[474, 475, 476, 477, 478, 479, 480]
	Statistical Models	
	Learning Models	

Table 19 – Taxonomy of time series methods for data generation

Per time series approaches correspond to methods that actuate over individual time series, only using their contained information. These techniques are usually referred to as classical methods, and conformed the state of the art in many fields of the time series domain before the apparition and expansion of deep learning methodologies. It is worth mentioning that some method names are quite general, thus it is possible to find in the literature some other meanings that slightly vary from the ones here exposed. The *per time series approaches* can be divided in the following domains, depending on which information they modify:

- Magnitude: these methods change the values of the time points without changing their ordering in the time dimension. Some common examples are jittering, rotation and scaling. The jittering transformation adds noise to each time point sampled from a chosen distribution, whereas, the rotation transformation rotates the full length of the time series for a specific number of degrees. On the other hand, scaling uniformly applies a linear function to all the time points of the time series.

- Time: these methods disrupt the order of the time points. Good examples are the techniques named cropping, permutation and flipping. Cropping puts attention only in a specific section of the time series, discarding the remainder. Instead, the permutation transformation slices the series into different segments and modifies their order in the trace. Conversely, the flipping technique inverts the order of the time points, making the first position become the last time point of the trace and vice versa.

- Frequency: these methods are less common, they actuate over the frequency spectrum rather than over the time dimension. The Fourier transform is usually mentioned as a representative case of this kind of techniques. Aside from the last, other frequency approaches obtain the spectrogram of the time series and apply some of the already mentioned magnitude and time domain techniques in its new form.

Figure 5 shows visual examples of some of the previous approaches. Notice that all these methods can be mixed together to conform hybrid techniques. We have included one of these in the Figure, named as window wrapping, which compresses or extends a determined segment of the time series.

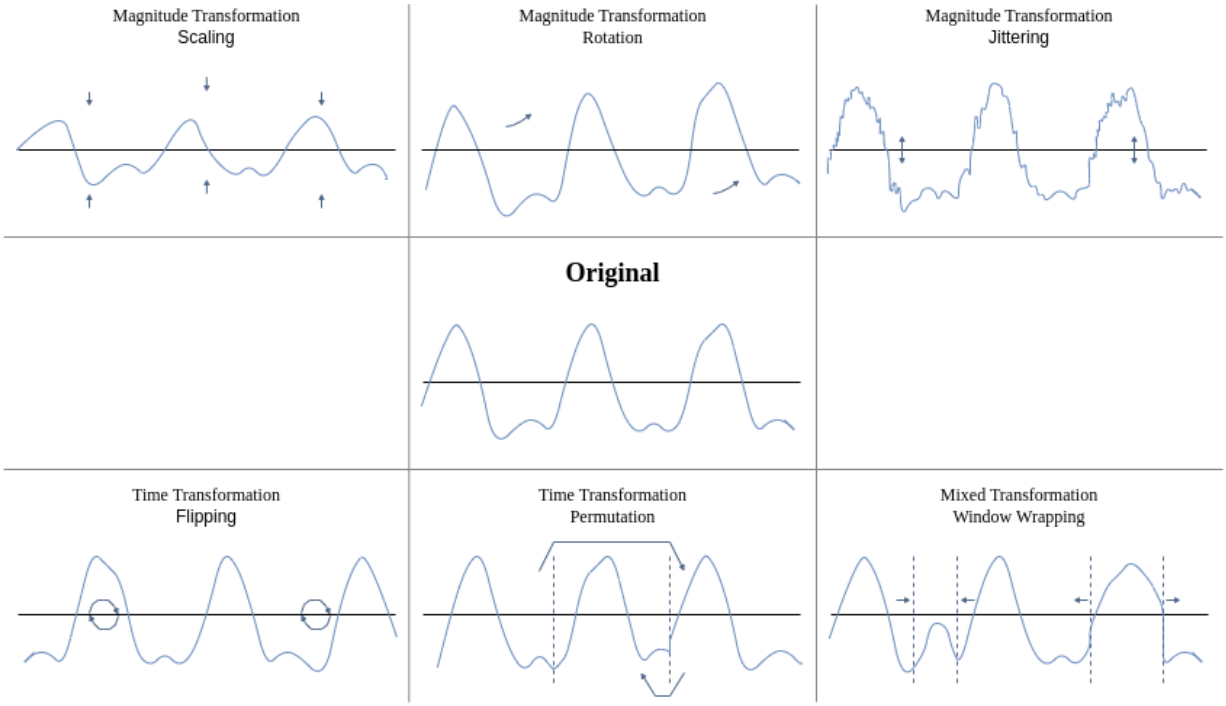

Figure 5 – Examples of *per time series approaches* for generating new data.

Concerning medical traces, we can cite the work from Guennec, et al.[472]. They experimented with the benefits of data augmentation techniques with ECG traces among other types of time series in a classification set up. They proposed a variation of the cropping mechanism, where each slice of a time series was analysed separately using a voting mechanism for taking the final decision for the overall time series. As well, they experimented with the window wrapping transformation, improving the final performance of the classification use case.

Mixing approaches relate to the methods that merge the patterns from a very small set of time series, usually only two. The advantatge of these techniques, alongside the *global approaches*, is that they do not generate new patterns from scratch, they use the information already available on the dataset to create new time series, mainaining somewhat the same latent distribution, unlike the *per time series approaches* family.

These methods can be divided as well into the three previous domains, magnitude, time and frequency. Depending on the information they modify they can end on one of them, being the hybrid domain also a possibility. The interpolation mechanism is the most popular. It performs an interpolation in the magnitude domain between two different time series, obtaining a new time series that has its values at a specific distance to both parent traces.

It is worth mentioning the publication from Kenji, et al. [473] that worked, between others, with ECG traces from the UCR Time Series Classification Archive. They generated a new time series from two distinct time series named as student and teacher. The new time series would maintain the features of the student time series while having the same pace of the teacher trace. This was performed with the help of a variation of the Dynamic Time Warping (DTW) measure which is widely used in the domain of time series. The last will be introduced in the Section 5.4. Interestingly, they slightly modified the DTW measure to use shapes instead of individual time points, obtaining more visually realistic results.

Global approaches comprise all those methods that use the information from a full set of time series. Methods that can generate new traces with patterns present in an extensive list of distinct time series. These approaches can be subdivided in the next domains:

- Statistical Models: This domain of techniques is quite direct in its approach which consists in choosing a statistical model that is suitable for representing the shapes and patterns of a dataset of time series. Once the model has been decided, the last can be used to sample new traces modifying the internal values of the model. These types of methods are very useful when the time series mirror a pattern or behaviour that is similar to a known statistical model. A clear example is using a linear function when the time series follow a linear pattern or using a sinusoidal model when the time series obey some kind of sine wave behaviour.
- Decomposition Models: These methods could be considered as a subset of the statistical models. They are based on the systematic components of the time series. As their first step, they find and obtain one or multiple statistical models for representing the level, trend and seasonality of the dataset and a model for depicting the noise distribution. From this point forward, they can generate infinite new time series varying the values of the statistical models and sampling from the noise distribution.
- Learning Models: Given the revolution of deep learning, these are the methods that attract the majority of the attention of the research community these days. They are methods that do not come with a predefined representation of the time series to generate from, instead they try to learn this representation directly from the data through a training phase. This kind of methods are useful for more complex time series that contain data patterns that change over time or which cannot be represented with only linear models. They also require less expert input and can be better automated, at least, theoretically.

  Due to the nature of these methods, they are less coupled to the type of data. In this way, the available learning approaches regarding time series in the literature are mainly adaptations from the disciplines of computer vision and natural language processing as previously mentioned. Therefore, the subdivision into VAEs [412], GANs [413] and Diffusion Models [414] from the Section 5.2.1 is equivalent in the case of time series data. The main difference comes from the much lower amount of contributions and thus, a less consensus in models, methodologies and datasets and the existence of big areas of research that required much deeper exploration.

Concerning medical traces, we have performed a review of the works from 2017 onwards, finding mainly approaches based on Generative Adversarial Networks (GANs) [474, 475, 476, 477, 478, 479, 480]. Also the publications from Zhou, et al. [476] and Hartmann, et al. [475] are worth mentioning. The team of Zhou proposed the BeatGAN, a hybrid generative model, having the global structure of a GAN and employing a VAE for the generator architecture. The final model was used for detecting anomalous beats in electrocardiogram readings, checking the distance between the real trace and the generated sample which represented the standard behaviour. The chosen methodology allowed the system to give clear explanations of the model results. On the other hand, the work from Hartmann, et al. presented and exposed a detailed description of how to properly train GANs to avoid the well known problems that they endure. They tested their results generating electroencephalographic signals with multiple evaluation metrics, obtaining realistic EEGs.

5.2.3 Genomics

In this section we present a brief introduction to the current state of the art techniques. There are mainly two types of techniques: based on statistical properties of the DNA and machine learning techniques.

Samani et al [481], an example of statistical exploitation side, they use a k-order Markov Chain in which the k defines the number of previous Single Nucleotide Variant (SNV) to consider to predict the current SNV. This SNV can have values from 0 to 2 considering the diploid genotype, which is the combination of alleles that this particular SNV has. This model considers the statistical correlation between the SNVs, as they have a "correlational" structure because of how the recombination process in the meiosis (cell split in sexual reproduction). On the other hand, a simpler approach is the Bernoulli model which is a model that assigns the alleles randomly using the population genome frequency for that particular variant position.

On the other hand, there have been approaches to this kind of data using Machine Learning techniques. Restricted Boltzmann Machiness (RBMs) and Generative Adversarial Networks (GANs) are used for this task.The RBM model is a neural network trained to create an embedding of the original data losing the minimum information possible. It is trained with the *Contrastive Divergence* approach which compares the input with the recreated output through a Gibbs sampling. Yelmen et. al [482] compared RBMs, GANs, Bernoulli and Markov chain models. Overall, RBM and GAN models performed better than Bernoulli and Markov chains regarding the generated shape in a Principal Component Analysis space and statistical properties found for genetic studies (e.g., Linkage Disequilibrium, minor allele frequencies...). RBMs tended to overfit the data more than GANs, however the statistical properties were more respected in this model. Moreover, RBMs have two additional features: Part of the input can be fixed in order to condition the synthetic generation output and the embeddings created in the model can be used to visualize the data. On the other hand GANs showed more resilience to attacks due to their underfitting compared to RBMs. However, the overfitting can be better controlled if a Nearest Neighbour Adversarial Accuracy metric is included in the training process. This metric tries to maximize the distances between the training set and the generated data, as can be found in the work of Yelmen et al. [482]. Finally, given that the GANs underfit the data, rare allele combinations are fixed to the most common one.

To address some of the concerns from this work, Arjovski et al. [433] define a new approach called Wasserstein GAN (WGAN), that has been applied also in genomic synthetic data generation [483, 484, 485], based on the Wasserstein's distance to include the haplotypic structural information into the generator as the loss function for the model, i.e., the metric to optimize when training the model. In this study they also include VAEs and Conditional Restricted Boltzmann Machines (CRBM), which is a kind of RBM that includes a window of data as extra information. In this case nearby SNV, like in the case of the Markov Chains, however with CRBMs the actual SNV and the "history" or the nearby SNVs network weights are computed as different matrices that affect each other. Both WGAN and CRBM outperform GAN and RBM, but both have their trade-offs in desirable properties.

In the work of Oprisanu et al. [404] a comparison between a copying model (from HAPGEN), a coalescent simulator, a recombination approach, RBM, GAN, WGAN and two new approaches introduced in this paper, Rec-RBM and Rec-GAN, can be found. The recombination approach is based on a genetic map which includes the recombination rate of SNVs, using them as prior knowledge for the generation. In Rec-RBM and Rec-GAN what the authors do is to generate first some samples with the recombination approach and then train the RBM and GAN models. This is done to mitigate the lack of data, i.e., to do a first step of data augmentation. The recombination model maintains the most statistical properties in general, so providing the combination with RBM and GAN generally improves the quality of the data generated afterwards, compared to only using the initial training sample.

Overall, WGAN and CRBMs are the best performing models and they can be further enhanced with an initial generation step with a better informed model with prior expert information embedded. With this approach we can overcome lack of data up to some extent and provide models like RBMs more data to produce more accurate synthetic data.

5.3 Existing Tools and Software Libraries

In this section we present a summary of the available tools and software found for synthetic data generation. Notice that, as we focus on providing an open ecosystem for synthetic data generation we are not focusing on already available solutions that are provided as a service, such as Gretel [30].

5.3.1 Images

Existing tools offer researchers and practitioners in the field of medical imaging the ability to leverage pretrained generative models for synthesizing realistic and diverse medical images. In our exploratory work, we have found some relevant tools that will allow easy deployment of generative models:

- **Medigan** [486]: Python-based tool specifically designed for medical image synthesis, providing a user-friendly interface for generating synthetic datasets. It offers a selection of pretrained generative models, allowing researchers and practitioners to generate diverse and realistic medical images for various clinical tasks. This library integrates popular deep learning architectures, such as Generative Adversarial Network (GAN) and Variational Autoencoder (VAE). These pre-trained models can be easily accessed through the MedGAN library, eliminating the need for users to build the models from scratch. Furthermore, this Python library also provides access to images present in public datasets.
- **MONAI** [487](Medical Open Network for AI): is a PyTorch-based framework dedicated to medical image analysis. It provides a comprehensive set of tools and utilities, including pretrained generative models, to facilitate synthetic medical image generation for different applications, such as image translation, data augmentation, and domain adaptation. MONAI's flexibility and extensibility make it a powerful tool for synthetic medical image generation, along with other medical imaging tasks.

While open-source tools have greatly contributed to the advancement of medical imaging, tools specifically dedicated to medical image synthesis, such as MediGAN and MONAI, are relatively scarce compared to other fields like medical imaging segmentation [488, 489, 490].

5.3.2 Time series

Unfortunately, in the case of time series the number of available frameworks for generating new data is quite scarce. There are no common tools that have spread across the research community. Nevertheless, we have compiled the following list of tools and frameworks after a brief search on the Internet[31]. Leaves as future work a further and deeper investigation of each of them to check whether they could be of real utility.

- A generator of synthetic time series by the *Nike* company. The framework is written in Python and it allows to create fake traces with distinct seasonalities, levels and trends. It could be useful to test our methods at first with dummy examples. It is available through this link at a GitHub repository.
- A package of the R language named as *gratis* that provides efficient algorithms for generating synthetic time series with diverse and controllable characteristics. Similarly to the previous one, seems it only contains simple and basic methods. In the same way, it could be useful for the initial phase of the project or as a method to test the effects of new evaluation approaches. The package internal implementation is available through this link at a GitHub repository.
- It is a public and extensive compilation of publications concerning the general domain of time series including forecasting, classification and anomaly detection. It also includes a list of the available surveys. It seems up-to-date and, thus, of great interest. It is available through this link at a GitHub repository.

[30] https://gretel.ai

[31] Notice that whenever we use the term of *generation of synthetic time series* we refer to the creation of new traces from scratch. When there is no intention to resemble to a previous known set of time series.

- A benchmarking framework for the generation of time series by the *datacebo* company (not synthetic). Looks like it includes deep learning models aside of classical techniques and a suitable set of evaluation approaches. It is available through this link at a GitHub repository.
- The implementation of a collection of methods for the generation of time series for the specific case of data augmentation (not synthetic). It belongs to one of the surveys cited in the 5.2 Section [469]. Unluckily, it exclusively contains basic methods. It is available through this link at a GitHub repository.

5.3.3 Genomics

In Genomics there are already open source synthetic data generators that implement different kinds of approaches. We provide a short list of three different types:

- **HAPGEN2**: Case control dataset at SNP simulator. It simulates haplotypes by making use of a recombination rate map at fine scale to ensure that the generated data has same Linkage Disequilibrium patterns of the reference data provided. HAPGEN2 can be used as standalone software or as an R package and it is publicly available here.
- **Coalescent simulators**: This kind of simulators generate the ancestry based on the data with the coalescent theory, i.e. assumption of no recombination, no natural selection, no gene flow and no population structure. This is done to treat each SNV equally likely. These simulators try to find common alleles in the sample to derive the ancestor. There are many softwares available like Msms. There is also https://cran.r-project.org/web/packages/coala/index.html which serves as interface to *ms, msms and scrm* to easily specify a model for simulation and conduct the experiments.
- The already mentioned methods can be found published as research code in repositories like GitHub. For example, all the methods from the work of Yelmen et. al [482] are available in their GitHub along with the generated genomes. An alternative implementation of the Wasserstein GAN focused on Population Genetic Alignments can be found in the following link.

5.4 Evaluation and research gaps

Evaluating how good or realistic is a generated data sample compared to the original data and measuring how novel it is (or how non re-identifiable it is compared to the original data) is complex. Therefore, in this section we list the metrics found for each data type for the review performed. Metrics are extremely relevant in every modeling process, but are especially important in data generation as there is no clear gold standard for this field. In the following sections we will see the metrics along with their definitions. Also as a conclusion, we list the research gaps extracted from this review.

5.4.1 Metrics and evaluation

5.4.1.1 Images

Metrics to evaluate image generation techniques is a fundamental aspect of the research. In recent years, several metrics have been developed to quantitatively measure the performance of image generation models. The invention of said metrics is complex, but they play a vital role in guiding the development and comparison of different generative models.

Metrics can be divided in different categories: overall image quality without reference and overall image quality with respect to ground truth.

The first group of measures focus on evaluating the diversity, visual fidelity, and distribution properties of the generated images without the need for labels:

1. Human observer: Refers to the evaluation of human observers. Human evaluation provides subjective judgements and insights that reflect human perception, preferences and aesthetic considerations. By collecting judgements from multiple human observers, statistical analysis can be performed, and can serve as benchmarks for image synthesis problems.
2. Inception score [491]: A measure of the quality and diversity of generated images, based on the activation patterns of a pre-trained inception model.
3. Fréchet inception score [492]: A value which measures the distance between the distributions of features extracted from real and generated images, based on the activation patterns of a pre-trained inception model.
4. Wasserstein distance [493]: A measurement of the distance between two probability distributions, defined as the minimum amount of work required to transform one distribution into the other.
5. KLD [494] (Kullback–Leibler divergence): A measure of the difference between two probability distributions, often used to compare the similarity of the distributions, with a smaller KL divergence indicating a greater similarity.
6. Perceptual loss [495]: A metric of the distance between generated, and real high-level features extracted by pre-trained neural networks.

Metrics in the second category evaluate the quality of generated images by comparing them to a ground truth or reference image:

1. Mean Absolute Error (MAE): A measure of the average magnitude of the errors between the predicted and actual values
2. Mean Squared error (MSE): A measure of the average squared difference between the predicted and actual values.
3. Peak Signal-to-Noise Ratio (PSNR): A measure of the quality of an image or video, based on the ratio between the maximum possible power of a signal and the power of the noise that distorts the signal.
4. Structural Similarity (SSIM) [496]: A measure of the similarity between two images based on their structural information, taking into account luminance, contrast, and structure.
5. Area Under the Curve (AUC): A measure of the performance of a binary classifier, calculated as the area under the receiver operating characteristic curve.
6. Visual Information fidelity (VIF) [497]: A measure that quantifies the Shannon information that is shared between the reference and the distorted image.
7. Universal Quality Index (UQI) [498]: quality of an image can be quantified using the correlation between the original and restored images.
8. Learned Perceptual Image Patch Similarity (LPIPS) [499]: An evaluation metric that measures the distance between two images in a perceptual space based on the activation of a deep CNN.
9. Metrics in image quality analysis by auxiliary task: If the datasets contain labels, this metrics measure how well the generated data performs in auxiliary tasks such as classification, detection or segmentation. To do so, they use pre-trained models for each selected auxiliary task and compare the performance of real data versus synthetic data.

As seen in Tables 16, 17 and 18, most works still use traditional metrics such as MAE, MSE or PSNR that are shallow and do not correlate directly to the human expert evaluation. Finding the best metric to quantitatively evaluate synthetic data is still an active field of research.

5.4.1.2 Time series

Similarly to the image generation techniques, the importance of a proper evaluation when working with time series is crucial. The majority of contributions in the literature assess the correctness of the generated data directly on the final evaluation of the specific use case. For example, in the case of performing data augmentation in a classification set up, the classification metric would be used to check whether it increases when using the original data along the generated samples. In this way, the evaluation methodology in these cases depends entirely on the final use case, which it is outside the scope of this work.

Taxonomy. Nevertheless, there exist a limited set of contributions that judge the quality of the generated traces alien to the final use case. We have summarized in a new taxonomy all the found metrics in Table 20. Notice that the majority of metrics are present as well in the domain of image data, making visible the sharing of knowledge between these two worlds.

Family	Domain	Examples
Direct comparison	Raw	Pearson Correlation Coefficient (PCC)
		Percent Root Mean Square Difference (PRD)
		Mean Squared error (MSE)
		Root Mean Squared Error (RMSE)
		Mean Absolute Error (MAE)
		Dynamic Time Warping (DTW)
		Abstract and Inception Score (IS)
		Fréchet Inception Score (FI)
Distribution comparison		Structural Similarity Index (SSI)
		Maximum Mean Discrepancy (MMD)
		Kullback–Leibler Divergence (KLD)
		Wasserstein distance

Table 20 – Taxonomy of of evaluation metrics for time series generation

Our taxonomy mainly divides the metrics in two families: *direct comparison* and *distribution comparison*. The metrics that perform a *direct comparison* are meant to compare between only two distinct time series. This comparison can be made taking the raw values of the series (*raw comparison*) or comparing both traces in a more abstract layer. This abstract layer is obtained extracting the internal latent features from a deep learning model that has been already trained with similar data. It is said that this mechanism obtains results more similar to the human perspective.

From all the exposed metrics of this family, only the Dynamic Time Warping (DTW) measure is native of time series data. Indeed, it is widely used in the domain, appearing in many contributions in the literature, being used for plenty of different purposes. The DTW metric overcomes the many issues that the Euclidean distance encounters in the case of time series data (equivalent to average error functions like MSE, RMSE, ...). The DTW finds the best fit in the temporal dimension between the two time series that are being compared without altering the order of the time points. In that manner, traces that are very similar but are slightly shifted will result in high similarity scores. On the contrary, applying the Euclidean distance would result in a quite low scores. It is worth mentioning the work from Le Guen and Thome[500] that proposed a smooth relaxation of the DTW metric for making it derivable. In that way, it could be used as the loss function by deep learning models in their training phase, solving the problems of training with the MSE measure and its equivalents.

On the other hand, the evaluation metrics that perform *distribution comparisons* can work with full length datasets. In addition, they can take into account the diversity and heterogeneity of the data. This can solve the mode collapsed issue, very common in GANs architectures, where the generator learns to generate a unique sample that is indistinguishable from the real samples. These kinds of problems are not correctly measure by

direct comparison metrics, because they do not consider the global perspective.

In general, there exist a gap in the literature regarding the effects and behaviour of each evaluation metric in the generation of new time series. Each contribution uses its own set of preferred metrics, a common consensus is missing.

5.4.1.3 Genomics

In order to show the utility of the genomic synthetic generated data, what it is usually done is to compare both the real data and the generated data using well known properties and metrics in the field of genetics, in order to measure the utility of the data. As found in the previous sections, mainly the metrics can be split into direct comparison and distribution comparison. Also, as in previous sections, there is no clear consensus on which are the appropriate sets of metrics that are needed to be used to test if the synthetic data is of use. In this section we list the metrics found:

- Wasserstein distance: Direct comparison. This metric that has already been introduced in the previous sections is also used in genomics.
- Euclidean Genetic Distance: Direct comparison. Measure of divergence between populations, focusing on the mutations. 0 means no difference in a particular place of the genome. This distance can be latter plotted to find patterns.
- Fixation index: Distribution comparison. Summary statistic comparison. Compares the allele variability of a sub-population with the global population, where being 0 translates to completely equal and being 1 translates to completely different.
- Linkage Disequilibrium: Distribution comparison. Metric that measures how different parts (locus/loci) of the allele sequences are associated. It is a metric of independence of the alleles. In other words, it measures the correlation of different parts of the DNA, which represents up to some extent the structure of the DNA in terms of combinations.
- Major Allele Frequency: Distribution comparison. Summary statistic used in population genetics. Useful to differentiate between common and rare variants in the population. It quantifies the most common allele frequency of each SNP. The distribution of the frequencies can be compared between the real sample and the generated sample with the Kolmogorov-Smirnov test.
- Site Frequency Spectrum: Distribution comparison. Summary statistic that can be seen as a density plot of the minor allele frequencies. Again, both real and synthetic data can be compared with Kolmogorov-Smirnov test to evaluate if the allele distribution probabilities are similar.
- Heterozygosity: Distribution comparison. Common in population statistics. Condition of having two different alleles at the same location. In population statistics, lower percentage of heterozygosity means lower diversity in the population. The distribution of heterozygosity can be again compared with Kolmogorov-Smirnov test.

5.4.2 Next steps in Synthetic Data Generation and Privacy

One of the principal concerns regarding the generation of health data is the exposure of sensitive information from the original samples. Health data is categorised as personal data by the General Data Protection Regulation (GDPR) and is subject to meticulous controls for assuring the privacy of the patient individuals. It is essential to prove that no possible reidentification or extraction of confidential information is possible before making publicly available data that has been generated from private health datasets.

From our review of the literature, we have noticed that this is an active area of research that has become more important in the past couple of years. In the examined contributions, we have identified three different approaches to tackle this concern:

- Direct comparison between samples: This methodology consists in first using the generation approach to sample multiples sets of synthetic time series. Subsequently, the distance computation between all samples of the training and testing sets with each one of the generated traces is performed. A simple threshold mechanism with precision and recall metrics is used to check the level of privacy. The goal is to obtain false positives (i.e. a particular record is incorrectly identified as a member of the training set) or true negatives (i.e. a particular record is correctly claim to not be in the training set) and to avoid true positives (i.e. a particular record is correctly identified as a member of the training set). The threshold is specific to the use case and to the convenience of the user or health institution[477].
- Distribution comparison between datasets: This approach consists in checking whether the distribution of distances between the synthetic samples and the training set is equivalent to the distribution of distances between the synthetic samples and the testing set. If both distributions cannot be demonstrated to be equal, symbolizes that the generation approach has ended memorizing instead of generalizing, maintaining identifiable data from the original training samples[474].
- Algorithms changes: This methodology consists in modifying the generation approach itself to theoretically disable the generation of reidentificalable samples. The most common approach is to use the so-called differential privacy. Differential privacy, as introduced already in the document, assures that the generated samples will change very slightly if a sample is removed from the training set. In that way, is very difficult to check whether a sample was used or not in the training set based only in the generated samples. This is usually performed adding noise in the internals of the algorithm[474, 501]. However, the added noise affects the quality of the final samples, the goal is to find a balanced approach between anonymity and quality of the synthetic data.

5.4.3 Related State-of-the-Art Gaps

Finally, based on the section analysis, in table 21 some preliminary State-of-the-Art Gaps have been identified.

Challenge Gap ID	Description	Flows	Related SECURED Component(s)
SoTA-GAP-09	Transformers have not been tested as the backend of Diffusion Models	Data	Synthetic Data Generator

SoTA-GAP-10	The work on genomic data generation is scarce and there is no modern framework to do it. Synthetic data generation for genomic data provides a more secure alternative to share de-identified data so that it can boost medical research. However, the alternative not come for free. As shown by Oprisanu et al. [404], the utility of this data in popular statistical analysis can be affected. Furthermore, the models themselves are not free from attacks as can also happen in the other data types, e.g. Membership Inference attacks. Finally, models that have higher utility metrics tend to have a significant reduction in privacy, showing that there is currently a trade-off between both [404]. On the other hand, in this data type the GAN mode collapse happens more often due to the low probability of rare alleles. Therefore, there is a need to study privacy preserving methods when training models for genetic data	Data	Synthetic Data Generator
SoTA-GAP-11	There is no clear comparison of the methods across of the image modalities in image generation. In imaging there are particular gaps in the literature. For example, usually CNNs have been used as backbone for diffusion models, but transformers could also be useful for this task. Moreover, diffusion models are prone to memorize the data [459], and this can also be the case of GANs. However notice that one of the possible solutions is differential privacy techniques. Further study of the effect of privacy preserving techniques needs to be done in this regards	Data	Synthetic Data Generator
SoTA-GAP-12	There is no standard solution and evaluation for data leakage in models that generate data	Data	Synthetic Data Generator
SoTA-GAP-17	In every data type we have not found a comprehensive comparison of all the methods. For example, in imaging, each of the generative methods are application specific and have been designed for a particular imaging modality: X-ray, MRI, Digital tissue image, etc. Therefore, each of the methods have different use cases and often different evaluation metrics, making it difficult to make a direct comparison across methods. Therefore, there is a need of a common framework that can compare the methods in a similar way	Data	Synthetic Data Generator
SoTA-GAP-18	Mixing Different methods is needed. Modern data generation techniques can benefit from classical techniques to enlarge the training corpus. This is the case for example of the recombination approach in Genetics, which mixes sequences to create a “descendant” of the original data. This method alone produces data that is easy to re-identify, however using it as a first step to a more privacy robust technique can enhance further the utility of the process as the modern technique will have more data to train on.	Data	Synthetic Data Generator

SoTA-GAP-19	In the case of time series, there are few works on this field in general and, again, no comparison between methods is available. There are, however, general domain comparisons for classic approaches [469] and for more modern approaches like GANs [471]. In this regard, there is a need of an empirical comparison of the medical traces with both classical and new Deep Learning methods.	Data	Synthetic Data Generator

Table 21 – Synthetic Data Generation main State-Of-the-Art Gaps

6 Health Data anonymisation

The various actors involved in the health sector, such as patients, caregivers, nurses, doctors, hospitals, researchers, pharmacists, health authorities and regulators, are generating a huge amount of data every second. Furthermore, the spread of telemedicine and the use of connected medical devices for monitoring patients in hospitals, ambulances and at home, and the growing demand of wellbeing wearable devices, generate a massive amount of health data. Big data analytics techniques can analyse this huge amount of data, helping to improve treatments, take faster medical decisions, prevent diseases, reduce cost of medical care, and, basically, improve the quality of life of the general population [502]. To protect and preserve the privacy of these sensitive data is a main challenge, when health data are shared with third parties for analytic purposes. The fulfilment of the regulations such as the GDPR [503], Data Governance Act[32] or the Data Act[33], must be assured in this kind of processes. Additionally, this sensitive information can be compromised by privacy threats such as user re-identification, linkability and inference [504]. Privacy-preserving techniques (PPTs) such as data anonymisation, generalization, perturbation or cryptography, are applied on health datasets for protecting health data, avoiding, or mitigating user re-identification. Following the EC thought that "an effective anonymisation solution prevents all parties from singling out an individual in a dataset, from linking two records within a dataset (or between two separate datasets) and from inferring any information in such dataset" [504], the SECURED project is facing these challenges aiming to preserve health data privacy, prevent successful anonymisation attacks, meet the requirements of current legislation and maintain the balance between data privacy and utility of these data for research purposes. This section describes the state of the art of anonymisation and de-anonymisation techniques and tools, performing an evaluation of the anonymisation techniques, which are more suitable for providing advanced anonymisation tools with sufficiently strong privacy guarantees.

6.1 Advanced Anonymisation Techniques

The anonymisation process involves turning personal or sensitive information (e.g., health data) to anonymous information, by removing personal identification information from a dataset through the application of anonymisation techniques, which preserve the privacy of the data subjects. The European Union regulation (GDPR) states that anonymous data is "information which does not relate to an identified or identifiable natural person or to personal data rendered anonymous in such a manner that the data subject is not or no longer identifiable" [503]. In this way, the anonymous data are not considered personal data and the GDPR is not applicable (Recital 26 [503]). There exist some misunderstandings related to this anonymisation process as the Spanish Data Protection Agency states [505], because it is not always possible to reduce the risk of re-identification and keep the utility of the health dataset for analytics processing. Also, anonymisation of a dataset is not forever as explained below in section 6.2. Recent studies suggest that several methods provide acceptable levels of privacy maintaining the predictive performance. Carvalho and Moniz [506] indicate that the application of PPTs combining generalisation, suppression and noise on large datasets, guarantee a user's high privacy level, a low risk of re-identification, but have an impact on the performance of prediction. Later on, Carvalho et al. [507] confirm that the application of different PPTs and the adequate parameterization on large and varied datasets allow to achieve reliable levels of predictive performance. They analysed the effectiveness of different PPTs in terms of re-identification risk and predictive performance, based on the statistical properties of the attributes and concluded that there is a trade-of between the risk of re-identification and predictive performance. Thus, the anonymisation process is not perfect and a trade-off between privacy and utility is confirmed (Figure 6).

The objective of this anonymisation task is to provide the tools to reduce the risk of re-identification of the health data as much as possible, keeping the utility of the data. The anonymisation models and the data anonymisation techniques used for accomplishing anonymisation, are applied on different types of datasets (structured and unstructured) and must respect the data usefulness and truthfulness, while the data user privacy is preserved [509]. In terms of privacy, four different types of attributes can be distinguished in a dataset [510][511]:

[32] https://digital-strategy.ec.europa.eu/en/policies/data-governance-act
[33] https://www.europarl.europa.eu/RegData/etudes/BRIE/2022/733681/EPRS_BRI(2022)733681_EN.pdf

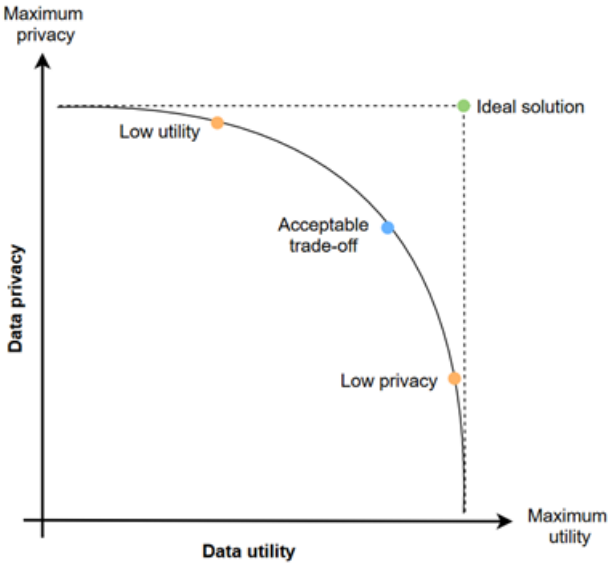

Figure 6 – Trade-off between data privacy and utility [508]

- **Identifying attributes** uniquely identify a person (e.g., name and surname, social security number). These attributes will be removed from the dataset.
- **Quasi-identifying attributes** do not uniquely identify a person, but in combination with other quasi-identifying attributes can be linked with a person (e.g., age, gender, birthdate, postcode). These attributes will be transformed by using anonymisation techniques such as aggregation or generalization.
- **Sensitive attributes** include private information of a person (e.g., disease, medical treatment) protected by law, and need to be kept private but are necessary for analytics purposes. These attributes will not be modified, but must not be linked to that person.
- **Insensitive attributes** can be public and will not be modified.

The basic privacy models commonly used in the medical domain are k-anonymity, ℓ-diversity, and t-closeness:

- k-anonymity [512] is a generalization model, which applies some modifications on the values of quasi-identifying attributes included in a dataset (i.e., transforms the value into a less specific one). k-anonymity groups at least k individuals with the same value in a called Equivalence Class (EC). The privacy requirement imposed by k-anonymity implies that "any released information should be indistinguishably related to no less than a certain number (k) of respondents" [513]. It means that each individual will be indistinguishable from $k-1$ individuals. Applying this model, the probability of re-identifying an individual is equal to or less than $1/k$. Thus, the higher the k, the lower the probability of identification, but when k is too high the utility of the data decreases due to information loss [512], [514], [502]. Although this model protects against identity disclosure, it does not protect against attribute disclosure.
- ℓ-diversity model [515] emerges to cover the k-anonymity limitations, for protecting sensitive attributes. As an individual could be identified in a dataset with low-frequency values, ℓ-diversity assures that at least ℓ distinct values must exist for each ℓ group/EC and sensitive attribute [515]. Since this model is vulnerable to probabilistic inference or corruption attacks, derived ℓ-diversity models such as recursive (c, ℓ)-diversity model or independent ℓ-diversity principle have been adopted and developed [516], avoiding data disclosure.
- t-closeness model [517] was proposed for overcoming the limitation of both the k-anonymity and the l-diversity models for improving the user privacy. In this case "the distribution of a sensitive attribute in any EC is close to the distribution of the attribute in the overall table (i.e., the distance between the two distributions should be no more than a threshold t)" [517]. Unfortunately, the increase of user privacy is linked to a reduction of the data utility.

- ϵ-Differential Privacy (DP) model [518] helps to improve the user data privacy by adding controlled random noise to a large dataset applying mathematical mechanisms (Laplace noise addition). The final dataset anonymised after interactive queries to a database, maintains accurate information for data analysis keeping the user privacy. Parameter ϵ is called privacy budget parameter and indicates the noise added; the smaller value of ϵ, the higher privacy protection. Recently, large companies are using DP for protecting microdata sets. Also, DP is applied in ML for enhancing privacy. An opposite view is suggested by Blanco-Justicia et al. [519].
- δ-dependency model is protection model for XML [520]. As prior privacy models for XML do not provide a proper privacy protection, Landerg et al. developed this model, which is based on the dissection method, i.e., separating quasi-identifying data from sensitive information, and a new privacy property, namely δ-dependency, which considers the hierarchical nature of sensitive data [520].

The following data anonymisation techniques [521][502] have been used to meet the requirements of the described privacy models and can be applied, among others, for anonymising health electronic records and can also be applied on the SECURED project:

- **Generalization** replaces the value of a quasi-identifying attribute by another less specific value, making the data less identifiable. This technique is more adequate for large datasets by using a set of ranges (discretisation), generalization hierarchies or recoding (global or local). The following can be identified as subgroups of generalization [522]:
  - **Global re-coding** groups the values into a broader category. If attributes are continuous, discretisation may be applied.
  - **Top-and bottom coding** is similar to global re-coding, but only applied to ordinal categorical attributes or continuous attributes: values above or below a certain threshold are re-coded.
- **Suppression** implies deleting values of an attribute in a dataset, either a column or a row, thus making it very difficult to recover the information, hence avoiding re-identification. This technique works well when used with ML models, but its utility drops when applied on big datasets.
- **Character replacement or data masking** technique replaces the value of an attribute by a missing value (NA) or special character (*, ?).
- **Perturbation** techniques alter the attributes' values in a dataset creating uncertainty on the original values. The level of perturbation must be controlled in order to diminish the impact on the utility of data. Perturbative methods include:
  - **Noise addition:** Addition or subtraction of small values to the actual value of an attribute, for protecting continuous variables, avoiding linkability.
  - **Shuffling and swapping:** Can be used on ordinal and continuous variables, where the values of the attributes are randomly interchanged. As it can be easily reversible, it must be combined with other anonymisation techniques.
- **Micro-aggregation** is an anonymisation technique leveraging k-anonymity model, creating groups of data with at least k similar records and swapping the entire cluster by its average value. This technique diminishes the loss of data when generalization, suppression or perturbation is used.

Besides the PPTs, **synthetic data generation** is a technique that leverages AI and uses ML, for creating simulated data files, which are indistinguishable from the real data, maintaining all their characteristics. They are used for model training and validation in ML. With this approach, it is not possible re-identify the user, preserving the user privacy. More details on this technique are provided in section 5.

The combination of these models with the techniques and methods described above, helps to enhance the preservation of the user data privacy [523]. Namely, the use of k-anonymity facilitates the trade-off between utility and privacy [524]. To determine which techniques are appropriate to preserve the data privacy, it is

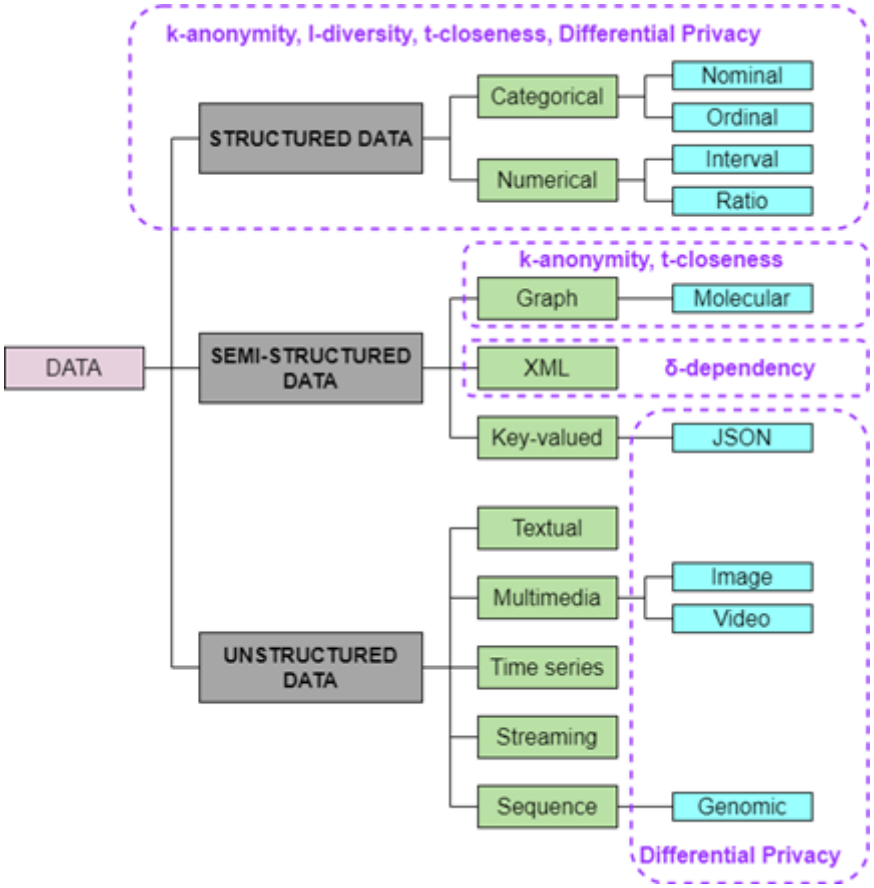

Figure 7 – Data taxonomy based on the structure of data and the appropriate PPT to use on them

fundamental to distinguish between the data formats to be protected. Depending on the structure of the data, there is:

- **Structured** data defined by a data model, included in a relational databases or spreadsheets, containing numbers, dates, strings, among other data types.
- **Semi-structured** data contains structured an unstructured data, e.g., emails contains sender or recipient as structured data and the message as unstructured data.
- **Unstructured** data without a particular organisation, such as text, images, videos or stream data.

Cunha et al. [525] suggest a data taxonomy for mapping the types of data and the appropriate PPT to use. Figure 7 shows the adaptation of this data taxonomy focused on the health domain mapping the anonymisation techniques can be used for each data type. Basically, the structured data can be anonymised by using k-anonymity privacy model and its derivations (l-diversity, t-closeness) and DP. DP can be useful for anonymising unstructured data and Key-value data. For anonymised semi-structured data such as XML and graph data, δ-dependency [520] and t-closeness [526], can be used, respectively. In the health domain, the electronic health records can take different forms. On one hand, datasets contain structured data, e.g., diagnosis or the code related; in this case k-anonymity and variants can be applied. On the other hand, medical images or patients' report are unstructured data, where DP fits better. De Capitani et al. [527] analyse the use of k-anonymity and extensions (ℓ-Diversity and t-closeness), indicating their validity for preserving user privacy in different scenarios including big data analytics. In this regard, k-anonymous solutions ensure scalability (considering volume of data and speed of computation), but the user of k-anonymised datasets from different sources can be an issue. Also, the combination of k-anonymity and DP can improve the user privacy and the utility [528]. As anonymisation is not perfect and k-anonymity and DP have some limitations protecting personal information and disclosing identity, Yamamoto et al. [529] suggest an innovative method named (ϵ, k)-Randomized anonymisation satisfying both k-anonymity and DP. Using real biomedical datasets they apply k-anonymisation and randomized

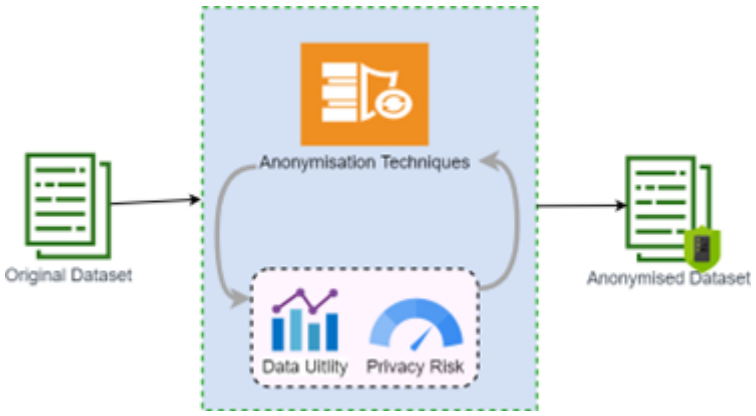

Figure 8 – Re-evaluation of anonymisation process based on the data utility and the privacy risk

response in sequence. Recently, a k-anonymity privacy protection algorithm has been proposed by Su et al. [530] for protecting privacy of multi-dimensional sensitive data, to effectively defend against skewness and similarity attacks. De Pascale et al. [531] provide the KGEN approach based on the k-anonymity model. This is a scalable, practical and data-intensive approach using genetic algorithms, dealing with large datasets. Although this approach has some applicability limitations the initial results are promising.

As described above, anonymisation techniques can be used for preserving privacy of medical information maintaining the utility of the data for later analytics. But the different data sources and the data heterogeneity limit the use of these data for cross-border data exchange between different platforms. To increase the effectiveness of these electronic health records, before anonymisation, it is necessary to harmonise the data into standards facilitating interoperability. There are several standards in the healthcare domain, supporting interoperability at syntactic and semantic level, such as HL7/FHIR, OPEN EHR or SNOMED, among others. There are also European initiatives for the digital transformation in the healthcare sector, such as the OPEN DEI project, working on interoperability and privacy aspects in the health domain.

The sensitive information collected in the SECURED project must avoid privacy breaches. The described anonymisation solutions are commonly used for preventing privacy leakage, but the risk of re-identification persists. Anonymised data can suffer attacks compromising the user privacy [502]:

- Background knowledge attack: the attacker knows a quasi-identifying attribute, identifying the user finally.
- Linkage attack: an untrusted data collector or an external malicious actor can exploit the quasi-identifying attributes and public datasets to identify a user.
- Attribute disclosure attack: based on the quasi-identifying attributes, the attacker can obtain sensitive information.
- Membership disclosure attack: The intruder deduces the presence of a person in a dataset.

Anonymisation can be reverted by applying de-anonymisation techniques that leverage powerful computing resources and new technologies or linking anonymous data with additional datasets, obtained from data breaches. Some aspects related to the assessment on anonymisation and de-anonymisation techniques are provided in the next section 6.2.

Besides the selection of the appropriate PPT, the process of performing the data anonymisation is very important as well; it includes determining the privacy leakage risk, the data structure, type of attributes and evaluates the data utility in an iterative process [508]. Figure 8 illustrates the process for re-evaluation of anonymisation based on the data utility and the privacy risk.

Additionally, Andrew et al. [532] presented a new privacy-preserving data collection protocol for anonymised sensitive health data without the participation of third parties or private communication channels. They applied the anonymisation techniques (suppression for nominal and numeric attributes, top-and-bottom coding for numeric, noise for float, rounding for numeric and global recoding for integer according to data features, evaluating

individually their impact [531]. Table 22 summarizes the limitations and the risk of re-identification for some of the described anonymisation solutions [502].

Anonymisation solution	**Allows re-identification**	**Limitations**
k-anonymity	No. Protect against identity and attribute disclosure attacks	Privacy is compromised when an attacker has high background knowledge or there is a poor variety of values in a group of sensitive attributes
ℓ-diversity	No. Protect against attribute disclosure attack, protecting sensitive attributes	Does not protect against membership disclosure attacks. If data are very imbalanced is difficult to create ℓ-diverse dataset
t-closeness	Better protection on sensitive attributes than ℓ-diversity.	If data are very imbalanced is difficult to create a t-close dataset
Differential privacy	No. Provides strong privacy guarantee	The noise added must be increased when multiple queries are made, for avoiding tracker attack. Thus, the number of queries must be limited. Low utility on microdata sets
Suppression	No. Useful when used with ML	Drop utility in big datasets. Must be applied together with other anonymisation techniques for avoiding privacy breaches and mitigating attacks
Generalization	Yes/No. Reduce linkability	Affect the utility of the dataset reducing granularity. Must be applied together with other anonymisation techniques for avoiding privacy breaches and mitigating attacks
Noise addition	Yes. Appropriate to protect continuous variables	Level of noise can affect the utility and privacy. Must be applied together with other anonymisation techniques for avoiding privacy breaches and mitigating attacks
Shuffling and swapping	Yes. Useful for analysing only one attribute	Must be applied together with other anonymisation techniques for avoiding privacy breaches and mitigating attacks
Character replacement	Yes. Applied on identifiers attributes	Heavily decrease analytic utility. Must be applied together with other anonymisation techniques for avoiding privacy breaches and mitigating attacks
Perturbation	Yes. Improve utility	Must be applied together with other anonymisation techniques for avoiding privacy breaches and mitigating attacks
Microaggregation	Yes. Appropriate for continuous variables	Can affect the computation of some measures sensitive to outliers

Table 22 – Limitations and risk of re-identification for anonymisation techniques

6.2 De-anonymisation Attacks

De-anonymisation and re-identification of anonymized datasets have been areas of active research due to the increasing concerns regarding privacy and data protection. When datasets relating to individuals are shared or published, there is a possibility that personal information may be inferred from the datasets, even when the datasets have been anonymized. To ascertain this risk, research has focused on three main types of attacks: re-identification attacks, membership inference attacks, and attribute inference attacks.

- **Re-identification Attacks:** Researchers have been focusing on developing sophisticated re-identification attacks to unveil the identities of individuals within anonymized datasets. These attacks exploit various vulnerabilities, such as background knowledge, auxiliary data, and external data sources, to link anonymized records to real individuals.
- **Membership Inference Attacks:** Membership inference attacks aim to determine whether a specific individual's data is present in an anonymized dataset. Researchers have been exploring different approaches, such as ML-based techniques and statistical analysis, to infer membership from released data.
- **Attribute Inference Attacks:** Attribute inference attacks involve predicting sensitive attributes of individuals from anonymized datasets. For example, by combining multiple quasi-identifiers (e.g., age, gender, occupation), an attacker may infer additional information about an individual that was not originally disclosed.

There are several examples of highly successful re-identification attacks in the literature, and some became so well known that they were featured on news headlines. In 2006, Netflix released a dataset as part of the Netflix Prize competition, which contained anonymized movie ratings from users. Researchers Arvind Narayanan and Vitaly Shmatikov demonstrated a successful re-identification attack on the dataset [533]. In particular, Narayanan and Shmatikov showed that by combining the Netflix dataset with publicly available movie ratings from the Internet Movie Database (IMDb), they could identify individual users with high accuracy. Similarly, in 2006, AOL released a dataset containing anonymized search queries of their users. The intention was to support research in search behavior analysis. However, the dataset was found to be vulnerable to re-identification attacks. Journalists at the New York Times analyzed the released dataset and were able to identify individuals by linking their search queries to known or personally identifiable information [534].

The main reason why re-identification is possible, and in many cases easy, is because information relating to individuals is highly unique, and therefore highly identifying, even when not directly an identifier (we refer the reader to the discussion on quasi-identifiers above). An example of this is the seminal work by De Montjoye et al. [535], who studied a low-resolution dataset containing fifteen months of human mobility data for one and a half million individuals, and found that human mobility traces are highly unique. The dataset contained the location of an individual at a coarse temporal granularity (hourly) and with a spatial resolution equal to that given by a GSM carrier's antennas (significantly lower than that given, for instance, by GPS). Their results indicate that four spatio-temporal points are enough to uniquely identify 95% of the individuals. By further reducing the resolution of the dataset, the authors were able to find a formula for the uniqueness of human mobility traces given their resolution and the available outside information. This formula shows that the uniqueness of mobility traces decays approximately as the 1/10 power of their resolution. Hence, even coarse datasets provide little anonymity. De Montjoye et al. repeated the experiments in a 2015 article [536], where they studied three months of credit card records for 1.1 million people. Again, their findings indicate that four spatiotemporal points are enough to uniquely reidentify 90% of individuals. Additionally, knowing the price of a transaction increases the risk of reidentification by 22%, on average. These findings represent fundamental constraints to an individual's privacy and have important implications for the design of technologies, frameworks and regulations aimed at protecting the privacy of individuals through anonymisation. In particular, they show that even anonymized datasets that provide coarse information at any or all of the dimensions, provide little anonymity.

Medical and physiological data, resembling biometric data, are even more uniquely linked to a single individual, and, therefore, are identifying data, which makes any anonymisation more vulnerable to attacks [537]. Ravindra and Grama [538] focus, for instance, on neuroimaging datasets, and present a de-anonymisation attack rooted

in the innate uniqueness of the structure and function of the human brain. Worryingly, their attack reveals not only the identity of an individual, but also the efficacy with which they can perform cognitive tasks. Their attack relies on matrix analyses techniques that are used to extract discriminating features in neuroimages and is effective in the de-anonymisation of publicly available databases. Even less data-rich sources of information can prove very difficult to anonymize: El Emam and Kosseim, in two linked articles that appeared in IEEE Security & Privacy [539, 540], discuss de-anonymisation risks to patients of prescription data, from Canadian and US perspectives, where the sale or transfer of prescription data from pharmacies to commercial data brokers, the processing of the data to analyze physicians' prescribing patterns, and the subsequent sale of these prescribing patterns to pharmaceutical companies are common. The propensity of people to employ the Internet as a diagnostic tool (searching information about symptoms, diseases and possible remedies) also presents a unique security risk. In the US, services such as WebMD and HealthBoards provide health news, advice, and expertise, allowing users to post publicly visible helth-related questions, and offering physician-led responses. Ji et al. [541] expose the fragility of the privacy of those who use online health forums through a proof-of-concept attack, where they successfully link 347 out of 2805 WebMD users to real-world people, finding the full names, medical and health information, birthdates, phone numbers, and other sensitive information for most of the re-identified users.

Membership Inference Attacks (MIAs) focus on determining whether a specific individual's data is present in a given dataset, even when the dataset is anonymized or publicly released. The primary objective of a MIA is to identify whether a specific record or data point in a released dataset corresponds to an individual whose information was used in the creation of the dataset. These attacks normally exploit statistical properties and patterns in the released data to make inferences about membership status, and often rely on subtle leakage of information present in the dataset. Attackers often leverage machine learning techniques, such as classification algorithms or black-box model querying, to analyze the dataset and predict membership/non-membership based on patterns, correlations, or features present in the data. While MIAs can be performed on anonymized datasets, their most common application domain is the attack of ML models [542]: by querying the model and observing statistical differences or patterns, attackers can infer whether a specific record was part of the original dataset used for training a model or if it belongs to an external individual not included in the dataset. As ML/DL has attracted broad interest in healthcare and medical communities, the generation of models based on sensitive patient data (such as images of scans) has become commonplace. However, research into the privacy attacks on deep networks trained for medical applications, show that inference-attack algorithms can be used by malicious parties to reconstruct images and text records, simply by using information obtained from queries to the model. Wu et al. evaluate two inference-attack models, namely, attribute inference and model inversion, and show that they can reconstruct real-world medical images and clinical reports with high fidelity [543].

Attribute inference attacks (AIAs) aim to infer sensitive attributes or properties of individuals from anonymized or aggregated datasets. These attacks exploit statistical patterns and relationships present in the data to make inferences about sensitive information (attributes or properties of individuals) that was not originally disclosed. For example, attackers may aim to deduce information such as medical conditions, financial status, political preferences, or personal traits by analyzing patterns and correlations in the data. AIAs leverage statistical analysis, machine learning techniques, or domain knowledge to infer sensitive attributes. By exploring patterns, associations, or dependencies in the released data, attackers make probabilistic judgments about the presence or absence of specific attributes for individuals in the dataset. AIAs often rely on auxiliary information sources to enhance the accuracy of their inferences. These sources can include external datasets, public records, social media profiles, or background knowledge about the population. By combining information from different sources, attackers can amplify their inference capabilities. Main sources of data that can be vulnerable to AIAs come from social media [544, 545], but even seemingly innocuous datasets can often be subject to AIAs: recently, an AIA was used against videogame players statistics [546].

De-anonymisation and re-identification attacks pose significant risks when it comes to healthcare data. Health data is inherently sensitive, containing personal and potentially identifying information that, if exposed, can have severe consequences for individuals. There are a number of key risks associated with de-anonymisation and re-identification attacks on health data:

- **Privacy Breach**: De-anonymisation and re-identification attacks can lead to a breach of individuals' privacy. By linking anonymized health data to real identities, attackers can expose sensitive health conditions, treatment history, medication usage, and other personal information. This breach of privacy can have serious emotional, social, and even financial repercussions for individuals.
- **Stigma and Discrimination:** Re-identification attacks on health data can reveal sensitive information about an individual's health conditions, including stigmatized conditions such as mental health disorders, sexually transmitted infections, or genetic predispositions to certain diseases. This information, if exposed, can lead to social stigma, discrimination, and even negative impacts on personal and professional relationships.
- **Targeted Exploitation:** De-anonymized health data can be a valuable target for malicious actors. By linking health information to specific individuals, attackers can engage in various forms of targeted exploitation, such as blackmail, identity theft, insurance fraud, or targeted advertising of pharmaceutical products or treatments.
- **Secondary Use and Data Linkage:** Re-identification attacks can enable the linkage of health data with other datasets, amplifying the potential risks. Combining health data with other personal data sources, such as social media profiles or financial records, can provide a comprehensive and intrusive view of an individual's life, enabling further privacy violations and potential harm.
- **Trust Erosion:** Privacy breaches and the risk of re-identification can erode public trust in healthcare systems, research initiatives, and data sharing practices. This lack of trust can deter individuals from participating in research studies, sharing their health information, or seeking appropriate medical care, ultimately hindering medical advancements and public health efforts.

To mitigate these risks, it is crucial to implement robust privacy protection measures when handling health data. This includes employing strong anonymisation techniques, which have been proven to be secure against known attacks. As is often the case in cyber security, research on de-anonymisation and re-identification attacks benefits data security by uncovering vulnerabilities and risks associated with anonymized datasets, thereby enabling the development of robust privacy-preserving mechanisms and defenses [547]. Through studying these attacks, researchers gain insights into the limitations of anonymisation techniques, identify potential weaknesses in data handling processes, and devise strategies to mitigate the risk of re-identification. This research helps in enhancing the security of sensitive data, improving privacy-preserving algorithms, establishing stronger anonymisation standards, and promoting responsible data practices, contributing to a more secure and privacy-conscious cybersecurity ecosystem.

6.3 Existing Techniques and Tools

This section provides an overview of the different tools and techniques devoted to data anonymisation (section 6.3.1) and data de-anonymisation (section 6.3.2). Also, a short introduction to interoperability standard tools is provided.

6.3.1 Anonymisation techniques and tools

Recently, several commercial and open-source tools have been developed for protecting sensitive data. The software presented in this section, includes open-source tools covering the models and techniques described in section 6.1, applied on the health domain in different projects and studies. Open-source tools maintained, at least during the last two years, have been considered [548].

Amnesia [549]: Amnesia is an online anonymisation tool for anonymising personal and sensitive data included in a dataset (structured data). Amnesia supports k-anonymity and km-anonymity privacy models, using generalization or suppression techniques. It provides a semi-automated anonymisation process for structured and unstructured data (set-value data). It accepts input datasets files in csv format. Some of the source code is

based on ARX [550]. It is able to anonymise sensitive metadata information from DICOM images. It is written in Java providing a ReST API, documentation, and a web-based Graphical User Interface (GUI). A GitHub repository[34] is available. It can be deployed on Windows, Mac OS or Linux operating systems

Anonimatron [551]: Anonimatron is an open-source, extendable data anonymisation tool. This tool anonymises structured database and files, can de-personalize or anonymize the data by replacing every different value in the database by a synonym. Anonimatron supports different data bases, is easy to configure and able to generate fake data (e.g., email addresses, names or unique identifiers). It is written in Java and running on Windows, Mac OS or Linux operating systems. It is also available as a library. It is available to developers through a GitHub repository[35].

ARX [550]: "ARX is a comprehensive open-source software for anonymizing sensitive personal data. It supports a wide variety of (1) privacy and risk models, (2) methods for transforming data and (3) methods for analysing the usefulness of output data." It provides privacy models, e.g., k-anonymity, ℓ-diversity, t-closeness, k-map, δ-disclosure, differential privacy, among others. It allows for combining these models with anonymisation techniques such as generalization, aggregation, random sampling, microaggregation, top and bottom-coding or suppression. This generic anonymisation tool, has been widely used in the health domain for anonymising health datasets (structured data) in several studies and projects ([502], [552], among others). ARX includes a desktop application, is written in Java, and it provides an API and broad documentation. A public GitHub repository[36] is available and regular software updates are issued, which is very useful for developers.

DANS [552]: The Data anonymisation Service (DANS) is an anonymisation tool, developed by Atos in the context of the medical data exchange demonstrator of the CyberSec4Europe[37] H2020 project. DANS is based on the open-source libraries provided by the ARX tool [553]. It accepts input datasets files in csv or xlsx format. DANS makes it possible to mitigate tracking and user re-identification by anonymizing sensitive personal data, leveraging k-anonymity and l-diversity privacy models, which enable the application of some privacy criteria over a particular dataset, protecting biomedical data against data disclosure. DANS is a modular solution (webapp and server side) offering an easy-to-use user interface facilitating the anonymisation process to low-skilled privacy users. To this end, additional privacy models and new features (such as differential privacy, and utility and privacy risk), and GUI improvements, are envisaged to be included in the DANS service. DANS tool is offered in two flavours to be utilized by the data providers:

- A Java library to be integrated in the data provider legacy systems. Also, this option allows the use of PPTs on the Internet of Things (IoT).
- An anonymisation service to be deployed on the data provider premises or in a trusted third party, exposed as a RESTful API. In this case, a web-based GUI is provided to the user for facilitating the anonymisation process, improving the user experience. It has been deployed and validated by health end-users [554].

μ-ANT [555]: μ-ANT is a microaggregation-based tool for protecting structured dataset, fulfilling k-anonymity and t-closeness. It is a standalone application written in Java accepting input files in csv format, and can be deployed on Windows, MacOS and Linux operation systems. A public GitHub repository[38] is provided.

PrioPrivacy [556] is a desktop application implemented in Java as an extension of ARX tool [550]. It leverages the k-anonymity model prioritising quasi-identifier attributes. It is intended to anonymise structured data and it provides a GitHub repository[39]

sdcTools [557]: sdcTools is software developed for Statistical Disclosure Control (SDC). Also, it can anonymise microdata. This free software is supported by Eurostat[40]. A GitHub repository[41] is maintained and it contains the ARGUS software comprised by two modules, namely mu-argus and tau-argus, including a Java interface, and the sdcMicro tool:

[34] https://github.com/dTsitsigkos/Amnesiaa
[35] https://github.com/realrolfje/anonimatron/tree/master
[36] https://github.com/arx-deidentifier/arx
[37] https://cybersec4europe.eu/
[38] https://github.com/CrisesUrv/microaggregation-based_anonymisation_tool
[39] https://github.com/alex-bampoulidis/prioprivacy
[40] https://ec.europa.eu/eurostat
[41] https://github.com/sdcTools

- **µARGUS (mu-argus)** [558]: mu-Argus is an open-source software devoted to creating safe micro-data files. It is written in Java using anonymisation techniques such as global recoding, top and bottom coding and local suppression. These methods are applied iteratively in a manual way. mu-argus accepts input datasets in csv and SPSS format. It provides a GUI and it can be deployed on Windows, MacOS and Linux operating systems. It can also be used to generate synthetic data.
- **τ-ARGUS (tau-argus)** [559]: tau-argus is an open-source software designed to protect statistical tables.
- **sdcMicro** [560]: sdcMicro is an open-source software for anonymising microdata files. It is written in R, C and C++. This tool uses k-anonymity (and derivations e.g., l-diversity) and global recoding, top and bottom coding, microaggregation, swapping and suppression, as anonymisation techniques, which can be applied manually in an iterative way. It provides a GUI to users for statistical disclosure control, showing details on individual risk, information loss and data utility. It can be deployed on Windows, MacOS and Linux operating systems.

There are other open-source anonymisation software tools, e.g., UTD Anonymisation toolbox[42], CAT[43], OpenAnonymizer[44]; however, they provide restricted privacy models or techniques and currently are not maintained. Also, there are professional tools such as aircloak[45] or Privacy Analytics Eclipse[46] used in the health domain, however we focus on open-source tools, as they better fit the SECURED objectives.

Interoperability standard tools and process: The heterogeneity of Electronic Health Record (EHR) data is a drawback for sharing clinical records between different hospitals or doctors in case of emergency or the patient move from one city to another or even moving abroad. Also, the need of using clinical records for research purpose implies that interoperability and preserving patient privacy is a challenge. The harmonization to standards such as HL7/FHIR, OPEN EHR or SNOMED is needed. Different initiatives have been conducted in this direction.

The **MODELHealth** project [561] obtains health data from the hospital databases harmonizing these data to HL7/FHIR standard and then apply the k-anonimity privacy model for data anonymisation, demonstrating that an adequate harmonization and anonymisation can be performed, while preserving data privacy.

The **HAPI FHIR**[47] open-source solution, is an implementation of the HL7-FHIR standard written in Java, for facilitating the interoperability. It comprises a client, including an Android client, and a server module providing a REST API.

At the end of 2022 a new privacy standard has been released, namely the ISO/IEC 27559 Information security, cybersecurity and privacy protection – Privacy enhancing data de-identification framework [562], providing "a framework or identifying and mitigating re-identification risks and risks associated with the lifecycle of de-identified data". This standard is based on the ISO/IEC 20889 [563] focused on the de-identification techniques applied to structured datasets, establishing a standardised terminology, a description of the de-identification techniques and how they can reduce the risk of re-identification. These standards will be considered during the development of the anonymisation tools.

[42] http://www.cs.utdallas.edu/dspl/cgi-bin/toolbox/index.php?go=home
[43] https://sourceforge.net/projects/anony-toolkit/
[44] https://sourceforge.net/projects/openanonymizer/
[45] https://aircloak.com
[46] https://privacy-analytics.com/eclipse-software/
[47] https://hapifhir.io/

6.3.2 De-anonymisation techniques and tools

Contrary to anonymisation techniques, for which general and data-agnostic open-source implementations are relatively frequently available, de-anonymisation attacks are often specific to a given data type, and, in many cases, dataset. De-anonymization techniques are tailored to specific data types or datasets due to the distinct characteristics, structures, and vulnerabilities associated with each type of data. The representation and format of data vary across types, such as structured or unstructured data, requiring de-anonymization techniques to account for these differences. Statistical properties specific to each data type, such as temporal patterns in time-series data or structural properties in graph data, play a crucial role in the de-anonymization process. Contextual factors, background knowledge, and the level of data granularity or aggregation also impact the choice of de-anonymization techniques. Additionally, domain-specific considerations, such as financial or genomic data, require specialized approaches to address unique privacy risks. Consequently, de-anonymization techniques are tailored to the specific attributes and characteristics of the data type or dataset in order to achieve effective re-identification. For reference, however, we list here some techniques and frameworks which have demonstrated a relatively higher level of generality.

De-Health [541]: De-Health is an online health data De-Anonymization (DA) framework aimed at identifying individuals that post health-related questions on online medical fora such as WebMD and HealthBoards.

Hidden Markov Model (HMM) techniques [564]: An HMM is a statistical tool used in modelling sequential observations (visible) that probabilistically depend on a hidden sequence of events (hidden states). As such, it can be particularly useful for de-anonymisation. Attacks based on the HMM include Forward de-anonymiser and the Kullback-Leibler Divergence de-anonymiser [564].

6.4 Comparisons and Evaluation of anonymisation Approaches and Tools

Table 23 [525] [565] [548] presents the comparison of the different anonymisation open-source tools described in section 6.3.1, considering several aspects related to the privacy models and the techniques provided, the type of data to anonymise, the utility and risk evaluation, the existence of user interface and API, programming language employed, last update and the complexity to use.

Tool/SW	**Privacy model/ technique**	**Data type**	**Evaluation**		**GUI/ Web App**	**API**	**Language**	**OS**	**Last Update**
			Priv	Util					
Amnesia	k-anonymity km-anonymity	Structured (tabular) Unstructured (set-value)	x	x	x/x	-	Java JavaScript	Windows Mac OS Linux	2022
Anonimatron	Replacement	Structured (tabular)	-	-	x/-	-	Java	Windows Mac OS Linux	2021
ARX	k-anonymity l-diversity t-closeness k-map δ-disclosure DP/ Generalization Suppression Microaggregation	Structured (tabular)	x	x	x/-	x	Java	Desktop app	2022

DANS	k-anonymity l-diversity t-closeness/ Generalization Suppression DP*	Structured (tabular)	x	x	x/x	x	Java JavaScript	Windows Linux	2022
µ-ARGUS	Noise addition Suppression	Structured (microdata)	x	-	x/-	-	Java C++	Windows Mac OS Linux	2021
sdcMicro	k-anonymity Noise addition Suppression Shuffling	Structured (microdata)	x	x	x/-	-	R	Windows Mac OS Linux	2021
PrioPrivacy	k-anonymity	Structured	x	x	x/-	-	Java	Desktop app	2021
µ-ANT	k-anonymity t-closeness	Structured	x	x	-/-	x	Java	Windows Mac OS Linux	2020

* Envisaged to be provided during the SECURED project

Table 23 – Comparison between described anonymisation tools.

The applied methodology, the selected privacy model and the technology included in a tool for anonymising health data, must take into account the data type, the expected privacy and utility level to achieve, the information loss during the process, as well as the possible adversarial behaviour and attacks. Haber et al. [548] recommend the use of "ARX for automated anonymisation of relational data and Amnesia for automated anonymisation of set-valued data, and sdcMicro as a library and tool for mostly manual anonymisation processes". Although DP limits the number of queries and the utility decreases when it is applied on microdata, it is of increased interest, as it provides a stronger privacy guarantee, independently of the prior knowledge of the attacker and avoiding linkage attacks. Therefore, DP is well-received by the research community. Also, big tech companies apply DP in their processes for ensitive information protection. Also, DP can be used in FL and the generation of synthetic data. The DANS tool (based on ARX) as a modular solution, is a very good candidate for adoption as an anonymisation tool in the SECURED project, as it covers a wide range of privacy models (k-anonymity, ℓ-diversity, t-closeness, differential privacy) to be applied on health data generated by the different pilots, and also, utility and privacy risk features to evaluate the risk of re-identification. The modular architecture allows to include new models, techniques and features needed for improving the anonymisation process. Additionally, a GUI and a web app which ease the anonymisation process to non-technical people improve the user experience of the data providers. Thus, the selection of DANS tool is based on the broad range of privacy models, techniques and evaluation features that this solution will provide, maintaining the utility of the data, while privacy is sufficiently protected.

6.4.1 Related State-of-the-Art Gaps

Based on the literature review [566][525][567] the following table provides the challenges, gaps and future directions need to be addressed on the anonymisation field. Some of them will be addressed during the development of SECURED project. Regarding gaps and challenges detected:

Finally, based on the section analysis, in Table 24 some preliminary State-of-the-Art Gaps have been identified.

Challenge Gap ID	Description	Flows	Related SECURED Component(s)
SoTA-GAP-20	It is necessary to find the privacy risk and data utility trade-off for different health data types. There is none publicly available tool able to implement and evaluate anonymization solutions for heterogeneous data types in the health domain (microdata, big data, free-text data, images, transaction data).	Data	Data Transformation Engine, Anonymization Service & Toolset
SoTA-GAP-21	There is a lack of a unified tool that is able to automatically suggest an anonymization solution depending on the data type.	Data	Data Transformation Engine, Anonymization Service & Toolset, Anonymization Decision Support
SoTA-GAP-22	Lack of good anonymisation methods for text documents.	Data	Data Transformation Engine, Anonymization Service & Toolset
SoTA-GAP-23	It is necessary to apply the existing methods in real health scenarios	Data	Data Transformation Engine, Anonymization Service & Toolset
SoTA-GAP-24	It seems that there is a lack of mature privacy mechanisms for applying on collection time, considering privacy, utility and efficiency.	Data	Data Transformation Engine, Anonymization Service & Toolset
SoTA-GAP-25	There is a lack of standardized and universal definition of privacy and standard methods to compare the existing anonymisation solutions.	Data	Data Transformation Engine, Anonymization Service & Toolset

Table 24 – Anonymization and De-Anonymization main State-Of-the-Art Gaps

Regarding the future directions there are some research aspects to be considered for improving the anonymisation tools and the privacy preserving process:

1. Research on vulnerability to different threats and attacks.
2. Research on new methods and improved algorithms of anonymisation to be applied on health domain.
3. Research on the use of cryptographic algorithms for anonymisation

7 Preliminary SECURED Components and Technical Requirements

7.1 User/Technical Requirement Collection Methodology

The best practice to promote service availability is to eliminate the obstacles that impede it in the first place as well as encourage sustainability-driven solutions and innovation in general [568]. However, in order to identify these obstacles and design optimal solutions, it is necessary to take one step back and address the whole problem in a more user-centric manner. The notion of a User Journey (UJ) or a Customer Journey (CJ) as the majority of the academia tends to call it, is a relatively new idea in system design which focuses on the entire User Experience (UX) [569], constantly identified by more and more scholars [570] as the optimal method of putting together intuitive, easy to use platforms with simple stories which increase end user engagement and consequently their overall experience. But then the next question emerges; what is UX? User Experience (UX) is defined as a person's subjective response, interpretation and consequent interaction with a product, system or service. It is directly linked with the person's perception of utility, ease of use and efficiency, therefore can be further divided into several response types like cognitive, emotional, behavioral, sensory, and social [571, 572]. As service providers gain momentum in real-world, large-scale economies and markets, the service industry attempts to address challenges related to user-centric, UX-boosting design with the optimal goal being no other than to elevate service sustainability [573] via satisfied and loyal end-users. However, despite the number and variety of studies on sustainability in services, there is little study on tools that may improve sustainable service design [574] and this is where the overall User Journey technique comes into play.

7.1.1 User Journey Approach

7.1.1.1 Origins of the User Journey

Back in the 1990s as the importance of the service sector grew, corporations turned into service operation optimization as a method to maintain their competitive advantage. Researchers reached the conclusion that the more satisfied customers feel about their experience in the service operations system, the more competitiveness the system possesses, as stated in [575] and consequently customer satisfaction became a major indicator of service operation sustainability [576]. After identifying low-quality services, providers attempted to remediate some of their fundamental flaws, but this turned out to be a huge challenge since there was little to none systematic qualification to ensure that user demands were properly treated, in a holistic, logical and scalable manner. The first attempt to address this limitation was made by Shostack [577], who created a service blueprint scheme depicting the broader concept of service operations. The service blueprint pinpoints customer interactions during the service operation processes and is used to split activities between the front office, where customers receive concrete evidence of the service, and the back office which is more or less hidden, outside the customer's view. One of the huge benefits of the service blueprint approach is its ability to simplify problem solving through timely failure point identification, while at the same time pinpoint opportunities and methods to enhance user perceptions [578]. However, service blueprint also has its limitations, mostly due to its design that remains a "conventional work-flow concept dominated flowchart" [579], unable to focus on the entire service experience of the customer or the service operations problems. In essence, the service blueprint reveals the failure of not providing researchers and practitioners with accurate and detailed information concerning the customer service experience, remaining provider- rather than customer-oriented [580]. The major shift of the service blueprint model toward a customer-oriented tool that visually describes the concept of service operation was carried out in 1999 by Tseng, Qinhai, and Su [579] which introduced the Customer Journey (CJ) framework by creating an innovative tool for service operations improvement by objectively mapping the service experience of customers. This was the first introduction of the CJ term in the literature.

7.1.1.2 Definition Evolution

As stated in [581] since the first references to CJ made by Tseng et al. [579], many definitions as well as new perspectives have emerged. Yet, the term itself remained the same until Marquez et al. [582] replaced it with User Journey (UJ). The specific approach seems more appealing for platform designers and framework architects, since it is not just paying customers those which interact with the platform/framework but a broader set of active users. However, the term Customer Journey (CJ) never lost its momentum against the User Journey (UJ) one, therefore for the scope of the specific analysis the former will be mostly utilized. Prior to 2010, scholars considered UJ as just the contact between the user and the service during the purchasing process [583]. Authors in [584] expanded this idea by stating that UJ can be divided into a sequence of events which customers follow to discover, interact and ultimately select firms, products, and services. This newly introduced interactive approach led researchers to create new methods for simplifying the rather complicated process of UJ analysis, which in turn ended in the creation of three distinct UJ phases by Lemon and Verhoef [569], namely "the pre-purchase, purchase, and post-purchase stage". This became the bedrock on top of which the contemporary definition of stages was later introduced. Moreover, the term Customer Experience (CX) also appeared in the definition of UJ, given the fact that the two terms are not only related but somehow interdependent. Currently, it became clear that it is essential to analyze CX in order to properly conceive UJ [585], while at the same time psychological factors linking UJ with emotional aspects were also introduced [575]. This approach strengthened the interrelation between CX and UJ, making Rudkowski et al. [586] to state that "the past fifty years of research has contributed to a holistic understanding of CX as a decision-making process or journey".

7.1.1.3 Key Characteristics

As researchers attempted to approach UJ in a holistic manner, the actual definition became much more complex and the following new terms were incorporated: stages, touchpoints, and personas [586, 587], each at a different phase. More specific, Kranzbühler et al. [587] build onto the existing theoretical framework for the touchpoints [569, 575], to establish (i) satisfying (ii) dissatisfying and (iii) neutral touchpoints [587], as well as online–offline ones [586]. Backed up by an explanatory definition, this approach highlighted the need for strict contact points between users and services. Yet, it was only until recently that the aforementioned, strictly defined sets of terms were utilized in the UJ definition and are now considered as its de-facto characteristics.

Stages. Tracking the individual contact points between the service and the user, is vital for understanding the overall user behavior which consequently provides insights regarding the experience. These contact points are known as "touchpoints" and may be used to identify cognitive, emotional, behavioral, sensory, and social responses, as described by Lemon and Verhoef [569]. These responses, which occur during the User Journey, when combined, map the overall customer experience in a holistic manner. It is therefore possible to define stages as the snapshots of a specific Customer Journey, which also contain the contact points between the user and the service as well as the generated responses in each contact point. However, it must be clarified that in a User Journey, the number of stages and as a result the number of touchpoints and experiences are not precise, but directly dependent on the type of the journey. In essence, the stages are differentiated according to journey type; for instance, for the travel and tourism industry where users are essentially customers, Gretzel et al. [588] and Wang et al. [589] divided the User Journey (used under the term Customer Journey) stages into the pre-trip phase, the en-route and on-site phase, and the post-trip phase. Similarly, when designing online marketing applications, directly considering users as potential customers, Lemon and Verhoef [569], through their extensive analysis of the CX, refer to the User Journey stages as pre-purchase, purchase, and post-purchase. However, the broader scope of system design dictates that users should be treated slightly different than customers and therefore their stages must be linked only with the user/service engagement flow and will be consequently mentioned as such for SECURED project.

Touchpoints. Touchpoints were first described in the scientific literature as encounters between providers and customers. Lockwood and Jones [590] described these encounters as interactive variables, specifically the "personal characteristics, perceptions of each other, social competence, and needs and objectives" between customers and providers. In the 1990s, researchers highlighted the social view of such encounters with respect to service providers, contact personnel, and customers [591]. They focused on the quality factors that affect said encounters during the service experience stages. However, the best definition of touchpoints is that of a direct or indirect contact [592, 593] where users interact with the service/product delivered to them via online platforms or other methods of personal interactions [594, 595]. As stated in the previous section, users form an experience at each touchpoint [596], which is then aggregated into the User Journey in order to generate the total CX [569, 597].

Personas. Personas are descriptive models of archetypal users derived from user research. They constitute an amalgam of multiple individuals with similar goals, motivations and behaviors. To properly represent the widest possible variety of users in any product or service scenario, personas need to be generated based on goals and behaviors rather than demographics or market segments [581]. To encourage realism and further increase user engagement, each persona could be potentially provided with a realistic name, a photo and some form of demographically-obtained data. Authors in [598] make the distinction between data-driven and assumption-driven personas. The former are formulated based on the principles of user research therefore their validity is extremely high and accurate. However, when there is little time to collect and analyze data, assumption-based personas are often utilized to ensure who the user might be, which are their likely goals and motivations and sometimes even predict their behavioral patterns. The introduction of personas in the design of a product or service can be a powerful tool to understand and visualize user goals, motivations and overall behavior. The authors in [592] investigated the suitability of design practices when it comes to user data format acceptance by designers. Preference to well-designed, visually stimulating, flexible, open-ended and easy to use methods was proven as well, however, in the same work authors concluded that similar approaches may not be suitable for presenting detailed technical information since focusing on archetypal users may lead to loss of generality for the rest of the population. Thus said, it appears that as a project moves from the requirements and concept generation stages to the product development stage, personas may need to be supplemented by more specific data regarding the user capabilities. Especially for medical applications, additional limitations should be taken into consideration. For instance, older users are often treated by designers as a homogeneous group, whilst in reality this is far from accurate. Age combined with life experience and physical capability limitations can be a differentiating factor which should be taken into consideration, as designers fail to understand the detrimental effect of aging to the physical and cognitive ability of users. In most cases, Personas can be aggregated in larger groups called Sets, which can be utilized to address broader attributes of a whole category of users, thus highlighting lifestyle diversity. This addresses the misconception that simple personas are adequate to represent a large enough portion of the population, an approach that sometimes leads to poorly designed systems, with limited usability. Data-driven personas remain of significant value and engaging, assumption driven personas tend to provide a persuasive and compelling vision of users that tackles potential scarcity of real knowledge on user needs. One should always remember that at its inception, the persona process was born from the necessity to include end user needs within the software applications of digital products, as well as to add detail of user requirements within the design process. Yet, the underlying principles make their application suitable to just about any field of design.

7.1.2 User Journey Mapping Technique

7.1.2.1 Practical Application of User Journey

The theoretical background of User Journeys led to the formulation of a methodology for depicting user behavior in a systematic manner, the User Journey Mapping (UJM) technique. This is more or less the practical application of the User Journey, originally described by Crosier and Handford [599] as a simple method for market research. UJM was only used for marked research until the scientific community "rediscovered" it, modify it accordingly and started using it in contemporary system design. In this brave new era of User Journey Mapping, many definitions have been provided, yet all tend to converge into proposing that UJM consists of the visual depiction [582, 598, 600] or visual representation [575, 592, 601] of a User Journey. Even though scholars agree on how exactly UJM is visually linked with User Journey, there are different approaches regarding its context and consequently which are the prime elements that any perplexed system analysis should begin with. Authors in [581] introduced three main categories in an attempt to include all existing definitions, that address User Journey Mapping as

- a function of Customer Experience (CX), where User Journey Mapping is only a depiction of the service delivery process from the user perspective [575], only focusing on the critical factors with a direct affect to its overall experience [592];
- an aggregation of touchpoints, where User Journey Mapping is just a simplistic presentation of the touchpoints used by users to interact with the service/system [598, 600];
- a direct, one-on-one User Journey representation and nothing more, placing the system designer in charge of the entire decision-making process [602], the presentation of stages a user is able to navigate into and the proper definition of the corresponding touchpoints.

There is little to no doubt that User Journey Mapping is an invaluable tool for assessing user behavior [582, 599, 601] as well as for service enhancement from the ground up. Due to its inherent affinity with User Journey, it allows system designers to gain significant insight on user motivations and behaviors, while in the same time tracks user responses to specific services [599]. By tracking predefined touchpoints, interaction channels and system functions, user experience is revealed, thus allowing practitioners to essentially "walk in the user's footsteps" [582]. Additionally, User Journey Mapping also boosts service delivery design as it allows architects to visualize it properly, map and properly decode all the steps required to perform a given task, capture detailed traces of user/system interaction and interpret negative or positive emotions [603]. As stated in [604], User Journey Mapping is a tool that greatly enhances the design and assessment of UX in a holistic manner.

7.1.2.2 User Journey Mapping Guidelines and Best Practices in SECURED

Under the auspices of SECURED Project, we will follow the approach which considers the User Journey Mapping as a visual representation of the overall experience a user has when interacting with a platform, a framework or a service. Without loss of generality, the specific definition allows us to focus on the pragmatic system design but from a more user-centric perspective. The main goal remains the same: tell the story of the user interaction across all touchpoints, reveal design flaws, technical and/or implementation issues as well as UX-related inefficiencies. Real-world examples and past experience in system design led to a gradual approach containing well-defined steps for creating a User Journey map. More specifically:

Step 1: Set clear objectives for the user journey map. There is no recipe for every system nor a "one-size-fits-all" approach. It is of paramount importance for a user journey map to have crystal clear objectives to begin with. This will provide some perspective on the problems the system designer tries to solve, as well as ease up the result extraction process, which will remain focused and crisp.

Step 2: Identify users and define their actual goals Identify the audience for a service, a platform or a framework greatly eases its design. There are vast differences in systems focusing on novice rather that highly

experienced users, each with varying results in UX as a whole. Taking some time to make proper assumptions regarding the personas of the user journey map is a good start and will allow the inclusion or exclusion of corner cases and scenarios of limited value for a user group.

Step 3: Identify all possible user touchpoints User touchpoints differ for each use case, platform, service or framework architecture. It is therefore essential to determine them by analyzing user/service interaction. The starting point is to always consider the needs of the target audience and consequently decide initial touchpoints they could potentially use. Pretend you are the user and track down all possible interaction one may have. Review user logs (if any) and categorize identified touchpoints for further analysis and consideration. Last but not least, touchpoints evolve in parallel with a service/platform, it is therefore important to constantly update/refine them for further increasing user experience.

Step 4: Identify user actions for every stage of the journey This one is straightforward and revolves around user actions and specifically what is a user doing in each step of a predefined path inside the overall User Journey. Checking a system from a user perspective as they navigate the process of using the provided service/platform is always a useful exercise because it helps designers understand where things can improve or where the experience is suboptimal. For best results, a user journey map should also note the channels in which these interactions happen to ensure that users are properly engaged.

Step 5: Identify potential changes which may compromise the overall flow, technical obstacles or pain points Where are users running into problems? What's stopping them from getting the best overall experience? Answering these questions allows system designers to identify improvement opportunities for augmenting user experience throughout the journey. Identifying whether users run into problems during a certain process, scenario, or service leads to address the stumbling blocks which compromise UX and lead to a seamless User Journey.

Step 6: Identify opportunities for improvement Constant iterations of fixes for specific issues, bugs, experience pain points and compromised services are extremely important and will boost functionality as a whole. A properly crafted user journey map facilitates improvements in both short and long runs.

In a nutshell, the scope of User Journey Mapping is to reveal if user goals and expectations are met, identify optimal solutions to existing issues and improve the overall user experience for a specific service, platform or framework. When designing a user journey map it is essential to focus on the bigger picture, try to understand what users want to achieve, which are their goals and most importantly: have a proper definition for success. Last but not least, try to remain platform-agnostic, since the main scope is to define the actions, users make to interact with the framework, rather than to create lists of specific functions with no additional value.

7.2 Preliminary SECURED Architecture and Component Identification

In order to properly identify the SECURED architecture components, it is of paramount importance to revisit the project's concept. In SECURED, our intention is to offer a one stop collaboration hub able to provide a secure and trusted environment for decentralized, cooperative processing of health data through SMPC techniques as well as generation of new, synthetic data and anonymisation assessment to health data providers and users. Our vision is to facilitate the broad adoption of health datasets across Europe by making the interconnection between EU health data hubs, the health data analytics research community, health application innovators (like Healthcare SMEs) as well as end users. Apart from an SMPC and anonymisation framework (with appropriate tools and services), the proposed collaboration hub will provide the means to engage its members in the EU health data community through proper training and well as synthetic data to stem health data analysis research, medical education and an increase of the associated datasets volume and considerably reduce their bias. The SECURED vision is to kick start an EU cross-border health data collaboration ecosystem for data providers, data researchers and innovators that will be able to produce new AI-based data analytics solutions and stem innovation. The overall concept is depicted in Figure 9.

In addition, the SECURED Description of Actions (DoA) revealed that certain tools as well as specialized services will be available to all users upon registering to the aforementioned collaboration hub. More specifically:

SECURED Toolbox

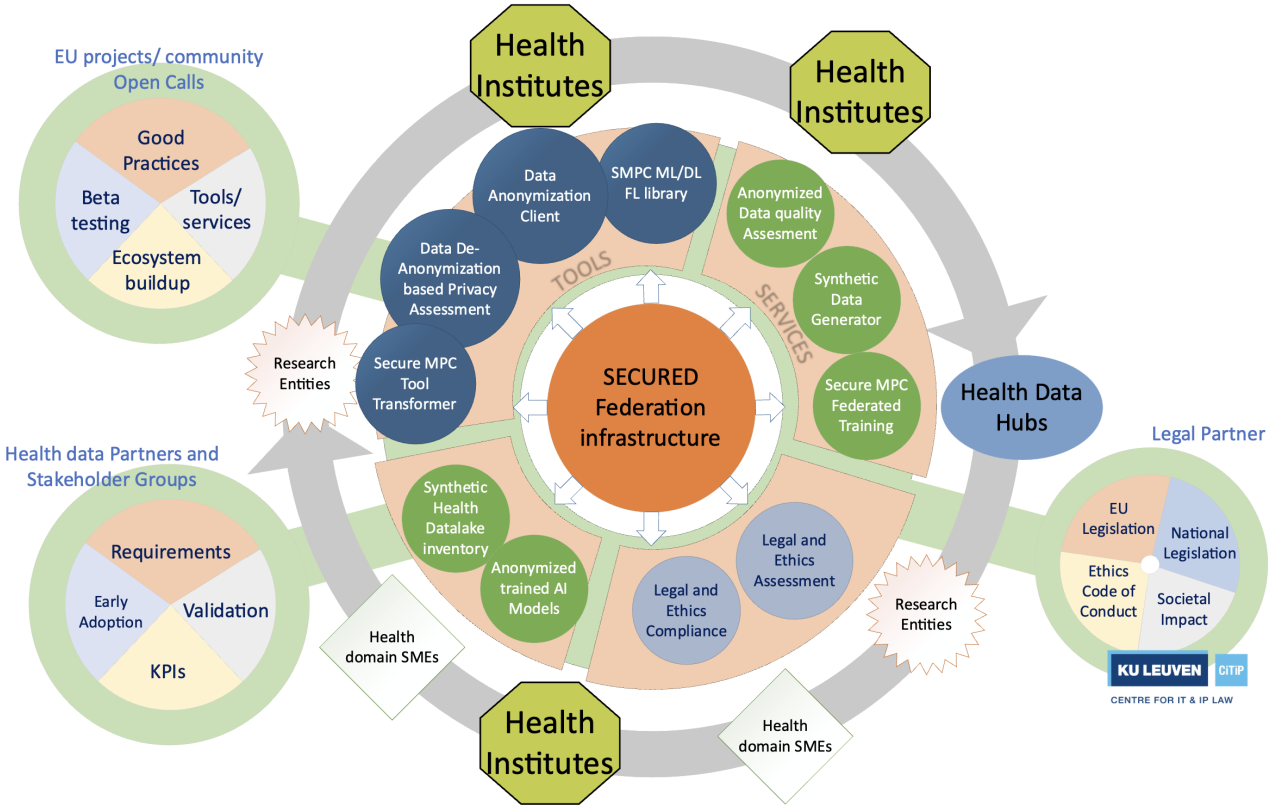

Figure 9 – SECURED Federation Infrastructure concept

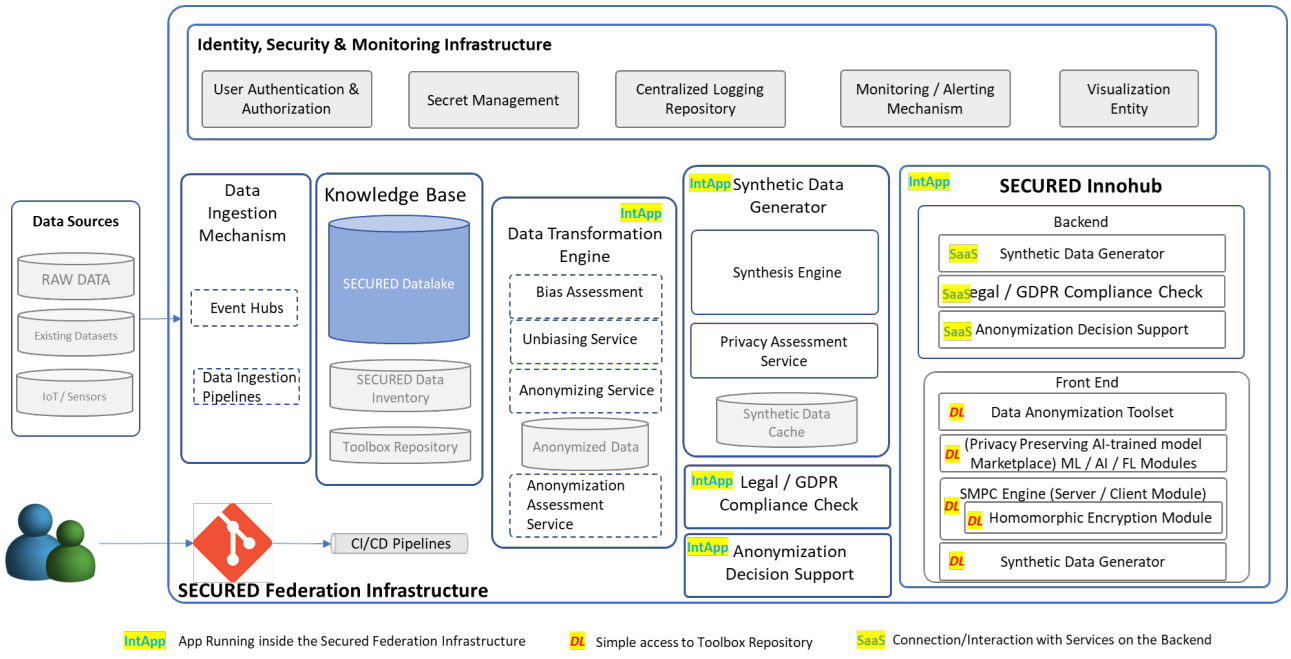

Figure 10 – Preliminary SECURED Architecture

- **SMPC Hardware-assisted software library**, for facilitating the execution of highly scalable SMPC solutions through cloud-based SMPC and/or dedicated hardware accelerated components for heterogenous MPSoC systems (that include multiple CPU cores, GPUs and FPGA fabric)
- **SMPC Transformation**, a tool for analysing existing AI-based Data Analytics solution, identify the components that can be made SMPC compliant and using the SMPC software library to transform the existing solution into a SECURED compliant tool that can operate collaboratively within a SECURED cluster using that private datasets of the cluster's parties and the SECURED Federation
- **Anonymisation tool**, which can be used at the member's premises in order to anonymize private datasets and AI models before sharing the with other parties
- **Anonymisation assessment tool**, for assessing the level of anonymity that is achieved in an anonymized dataset
- **Dataset Bias quality assessment**, a tool for identifying how biased is the dataset and produce a bias score that will afterwards be associated with the dataset
- **Dataset Unbiasing tool**, which can be used to enhance a biased dataset with synthetic data in order to reduce the bias score

SECURED Services

- **Synthetic Data Generator**, delivering "synthetic data-as-a-service" support to registered users, through a direct link with the corresponding service operating at the SECURED infrastructure backend.
- **Anonymisation decision support**, a dedicated assistance for choosing the appropriate anonymization technique that best fits the characteristics of a health dataset.
- **Cross-border data processing legal/GDPR compliance**, a dataset-specific legal framework to ensure dataset compliance with EU legislation per case.
- **Privacy-preserving AI-trained model "marketplace"**, a repository of anonymized, unbiased, trained models that have been produced by the SECURED federation.

It is therefore evident that the overall SECURED architecture is a highly perplexed amalgam of interconnected services, which will be henceforth termed as the *SECURED Federation Infrastructure*.

7.2.1 SECURED Federation Infrastructure

The broader SECURED architecture contains several interconnected services which cooperate to provide the overall platform functionality. Purely from an architectural standpoint, these services are contained inside the SECURED Federation Infrastructure, more or less an umbrella-entity that allows us to define the boundaries of the provided solution as well as to properly identify the necessary communication interfaces and flows that need to be implemented. All services inside the SECURED Federation Infrastructure have an inherent interaction ability only limited by networking configuration, simply for security reasons.

7.2.2 Identity, Security and Monitoring Infrastructure

Users trying to access the Security Federation Infrastructure need to be authenticated and monitored, while unauthorized user access must be prevented. The Identity, Security and Monitoring Infrastructure is the architectural node responsible for federated user authentication, delegated fine-grained authorization ensuring role-based access to corresponding services, user management and last but not least communication channel encryption. In addition, the node incorporates the centralized logging repository for aggregating logs from the distributed system, a dedicated monitoring/alerting mechanism and a visualization entity for transforming system-oriented information into a more user-friendly format

7.2.3 Data Ingestion Module

The specific entity is responsible for accurate, timely and error-free data transfer from all external data sources to the data repositories and/or data transformation functions of the SECURED Federation Infrastructure. The external data sources range from non-structured (DICOM images, .jpeg files or .documents), semi-structured (.csv, logs, .json and xml files, all considered as loosely typed data formats) and fully structured (i.e. large datasets retrieved from relational databases with a pre-defined schema and strong structure) synchronous datasets to asynchronous streams which may require dedicated parsers for proper identification and ingestion.

7.2.4 SECURED Knowledge Base

The SECURED Knowledge Base will act as the major data storage module of the broader SECURED Federation Infrastructure. Due to the vastly heterogeneous data that the specific module needs to handle, it must integrate several data storage solutions, each with a predefined scope and functionality.

7.2.4.1 SECURED Data Lake

The definition of Data Lake is that of a centralized repository designed to store, process, and secure large amounts of structured, semistructured, and unstructured data, with the inherent ability to store data in its native format and process any variety of it, ignoring size limits. A data lake provides a scalable and secure platform that allows analysts to: ingest any data from any system at any speed—even if the data comes from on-premises, cloud, or edge-computing systems; store any type or volume of data in full fidelity; process data in real time or batch mode; and analyze data using a wide variety of programming languages as relevant tools. The SECURED Knowledge Base will incorporate a Data Lake to act as its main data repository, thus exploiting its storage and processing abilities for every aspect of the project.

7.2.4.2 SECURED Data Inventory

In order to boost performance and avoid constant Data Lake parsing, the SECURED Knowledge Base will also integrate a metadata-optimized database, the SECURED Data Inventory. This component is meant to keep track of the datasets that have been registered in the SECURED Innohub and provide the authorized users the means to reach them from the dataset respective repositories. While the Data Inventory is not providing anonymized, privacy-preserving datasets, it provides the path in order to reach them.

7.2.4.3 Toolbox Repository

The Toolbox repository provides the storage location in the SECURED Federation Infrastructure where all the SECURED tools are kept. The repository uses CI/CD pipelines in order to keep the tools that are provided by the SECURED partners and allow the easy deployment of such tools in the SECURED Innohub users. The tools that the repository hosts are originally described in the project's Description of Actions (DoA) and can be further grouped into the Secure Multi-Party Computation (SMPC) Engine that consists of the SMPC Hardware assisted software library and the SMPC Transformation, the Data Anonymization Toolset that includes the Anonymization tool, the Anonymization Assessment tool, the Dataset Bias quality Assessment tool and the Dataset Unbiasing tool as well as tools that can be used for training and updating ML/DL and FL models in a privacy-preserving manner (ML/DL and FL Modules). Note that the SMPC Engine includes a dedicated module for HE (the HE module) with all the hardware assisted software libraries for performing Homomorphic Encryption.

7.2.5 Data Transformation Engine

The Data Transformation Engine operates as a Platform-as-a-Service within the SECURED Federated Infrastructure and utilizes the Data Anonymization Toolbox-as-a-service for the SECURED Innohub users. Such users can register and upload an anonymized health dataset (using the Data Ingestion Mechanism and the Knowledge Base components) and then deploy the Data Transformation Engine on it. Then the Engine is able to perform bias Assessment and Anonymization Assessment and if bias and anonymization vulnerabilities are discovered then perform an unbiasing and (re)Anonymization to remove them. The Data Transformation Engine, as expected, has an internal data storage where intermediate results of the above described process are temporarely stored.

7.2.6 Synthetic Data Generator

The Synthetic Data Generator is the component that will be in charge of producing new data when required. This component is data-driven, therefore it will require an external preparation, e.g. machine learning training. Afterwards, the component will be available to use. This external preparation can be done outside the system without requiring internal resources, but there is also the possibility of training it inside the SECURED Innohub platform if the required components are in place, e.g. Federated Learning Framework and associated computing resources. This component will be made of smaller components, one dedicated to each data type and modality. Note that the Synthetic Data Generator can be used as a platform-as-a-service or can be downloaded as a tool and executed locally at an end user's premises.

7.2.7 Anonymisation Decision Support

Given the broad variety of anonymization techniques and the fact that each one of the operates optimally on specific types of data, the Anonymization Decision Support component offers guidelines on a given dataset regarding the anonymization approach that should be followed on this dataset. The Anonymization Decision Support is a platform-as-a-service (an application running inside the SECURED Federation Infrastructure) where a registered SECURED Innohub user provides as an input information related to a specific dataset and the service suggests the optimal techniques to be used for anonymizing the dataset and/or for performing anonymization assessment. These techniques are, as expected, supported in the Data Anonymization Toolset and the Data Tranformation Engine. The Anonymization Decision Support component can be offered also as a tool to be downloaded and executed locally at an end user's premises.

7.2.8 Legal Compliance Check

This component is offered as a service of the SECURED Innohub and is providing support on the legal aspects related to a privacy sensitive dataset. Since there are different regulations in each EU member state as well as the EU as a whole (e.g. the GDPR) on citizen's data privacy, when a given dataset through the SECURED solution is used across countries within EU, the users should be aware of all relevant privacy related regulations (that may be different from country to country) that are applicable to such a dataset.

7.2.9 SECURED Innohub

The SECURED Innohub will act as the user entry-point to further exploit the SECURED Federation Infrastructure functionality. The specific module will allow access to authorized users only and will contain links for downloading (i) the latest version of Data Anonymization Toolset, retrieved directly from the Toolbox Repository (ii) the SMPC Engine modules, (iii) the HE module and (iv) ML/DL/FL modules. Upon downloading these software bundles, users will be able to execute them on prem, following the detailed instructions delivered by the developers of each module. In addition, the SECURED Innohub will act as the gateway for the SaaS solutions of the SECURED Federation Infrastructure, such as the Synthetic Data Generator, the Anonymization Decision Support and the legal/GDPR Compliance Check.

7.2.10 Auxiliary Modules

7.2.10.1 Codebase Repository

A codebase is any single repo (in a centralized revision control system like Subversion), or any set of repos who share a root commit (in a decentralized revision control system like Git). One codebase may maps to many deploys, but there is always a one-to-one correlation between the codebase and the app. The codebase remains the same across all deploys, although different versions may be active in each deploy. For example, a developer has some commits not yet deployed to staging; staging has some commits not yet deployed to production. But they all share the same codebase, thus making them identifiable as different deploys of the same app. Thus said, there is the necessity for a dedicated repository for the overall codebase of SECURED Federation Infrastructure, mapped by the specific module.

7.2.10.2 CI/CD Pipelines

A continuous integration/continuous delivery (CI/CD) pipeline is a framework that emphasizes iterative, reliable code delivery processes for agile DevOps teams. It involves a workflow encompassing continuous integration, testing, delivery, and continuous delivery/deployment practices. The pipeline arranges these methods into a unified process for developing high-quality software. Test and build automation is key to a CI/CD pipeline, which helps developers identify potential code flaws early in the software development lifecycle (SDLC). It is then easier to push code changes to various environments and release the software to production. Automated tests can assess crucial aspects ranging from application performance to security. In addition to testing and quality control, automation is useful throughout the different phases of a CI/CD pipeline. It helps produce more reliable software and enables faster, more secure releases.

7.2.10.3 Container Registry

Container images are standalone packages of software that can be used to quickly build and run containerized applications and their dependencies. These software packages form the basis of any contemporary cloud-native ecosystem. Used with container engines like Docker, container images transform the way that software is developed and delivered. But without a way to organize and share container images, they won't be nearly so useful. A container registry is a place to store container images for use in application development—especially cloud native development on microservices and containerized applications.

7.3 Preliminary Use-Case Descriptions with regards to the SECURED Architecture

While the main goal of this deliverable is not to capture and describe in detail the user requirements and use-cases of the SECURED project, user requirements are a necessary part of the user journey/process mapping mechanism that is adopted in the project in order to extract technical requirements. Thus, T4.1 has been following closely the activities of T5.1 where user requirements have been collected and extracted so as to utilize the collected input for the T4.1 activities and the D4.1 deliverable. However, since the full procedure and actions of user requirement collection is part of the T5.1 activities, in D4.1 deliverable we don't provide analytic information on the T5.1 work but rather focus on results of the task that are relevant to T4.1 and D4.1 activities. Given that T5.1 will deliver the final outcomes in M18, in D4.1 (that is delivered at M6) we can only consider preliminary user requirements to extract technical requirements of the SECURED solution. However, the work done up until M6 and the collected inputs from the SECURED pilots/use-case providers is deemed sufficient to extract realistic technical requirements of the SECURED solution. Eventually, when the final user requirements are provided, T4.1 in close collaboration with T5.1 will be able to revise any technical requirements that become obsolete by M18 in the D4.2 deliverable.

The interaction with the use-case provider/pilot partners consisted of several offline email exchange and a series of bilateral online teleconferences (one per use case partner/pilot) where we applied the first stages of the user journey approach and identified involved users per pilot, basic processes that the pilots want to be implemented and eventually the end goals of each use-case. The consolidated input from the bilateral teleconferences were processed and were used as a starting point for the first end user SECURED workshop, a physical meeting that took place on the 13th of June in Barcelona (hosted by the partner BSC and led by the T5.1 task leader EMC). The second phase of user requirement collection, the association with the SECURED preliminary Architecture and the alignment of the architecture to the involved use-case providers/pilot was extracted from the discussions made in the 1st end user SECURED workshop. In the following subsections we provide the extracted outcomes of the above activities from the viewpoint of T4.1, i.e. from the SECURED architecture and SECURED technical requirements perspective. More specifically, in subsubsection 7.3.1, Core user requirements that can be applied to all use-cases (including consortium use-case providers/pilots and any external open-call participants) are briefly provided. Beyond the above requirements, in subsubsection 7.3.2 the extracted outcomes of the 1st end user SECURED workshop regarding each use-case partner are provided

and more specific use-case specifications/requirements are described. In this subsubsection we also provide information on the components of the preliminary SECURED architecture that each use-case is expected to use.

7.3.1 Core user requirements

User requirements are defined by the end-users of a platform, framework or component and express how the specific element is expected to perform, strictly from the user's perspective. It is rather obvious that user requirements provide information that can be treated as the baseline for further specification, design and verification of any attribute of the corresponding element. During the design phase of the SECURED Architecture, an attempt to identify core user needs was carried out, through a dedicated workshop where user feedback was collected. Initially, we divided the overall framework we were trying to design into distinct components, for which the users were directly interviewed. These distinct components provide a bird's-eye view of the system and are presented in the following paragraphs.

7.3.1.1 Login Requirements

When users try to interact with a service or a platform, their first action is to establish connection and issues requests toward the Login module, namely the software component responsible for user authentication. Its main purpose is to ensure that only users with access rights will be able to join the platform and interact with the available content. The login module typically presents a login screen or interface where users are prompted to enter their username and password. Then the module validates the provided credentials against a pre-defined database of authorized users and grants or denies access based on the result of this authentication process. Since login modules are commonly used in applications, operating systems and software where access control is required, users are accustomed with their role and have a really clear idea of how such modules must behave, together with their essential functionality, all listed in Table 25.

Code name	Description
CUR_1	Must provide a single login page for all roles
CUR_2	Must allow new users to register in the platform and automatically grant access based on their respective roles
CUR_3	The platform should provide a new user initiation mechanism which must also support invitation by existing users. New, registered users may have access only to public materials
CUR_4	A platform user should be able to change his/her password
CUR_5	The platform should provide "forgot password" functionality
CUR_6	The service should provide federated authentication
CUR_7	The service should provide role-based access
CUR_8	The service should handle SECURED-specific roles. The service should provide overview of users based on their login name, access rights, roles and last access date. Any roles or role groups should be able to assigned to or revoked from a user
CUR_9	In general, users should only need to verify their identity during the initial process

Table 25 – Login Requirements

7.3.1.2 Data-Transfer Requirements

Data transfer refers to the collection, replication, and transmission of data from a specific organization, platform, framework or software component to another. Between larger systems and organizations, the term "data transfer" is directly linked with the concept of secure enterprise data sharing between business partners. Because the data is moving beyond the enterprise perimeter, care must be taken to secure the data. The major user requirements regarding data transfer, as defined under the auspices of the SECURED framework are listed Table 26.

Code name	Description
CUR_10	All communication channels must be encrypted / secure
CUR_11	User must be able to instantly drop communication channel if it is considered compromised
CUR_12	A registered user must be able to load privately-owned data to the system and should be able to define access policy/rights to the data, without any additional authentication

Table 26 – Data Transfer Requirements

7.3.1.3 Data-Storage Requirements

Data storage can be defined as the recording of information in a convenient medium, which allows its retrieval, reproduction and availability. In the digital age, the term refers to the use of recording media to retain data using computers or other devices, with the most prevalent forms of data storage being file storage, block storage and object storage, each suitable for different purposes. The SECURED framework will incorporate a series of such solutions to ensure sensitive information handling, between all participating entities. As such, there are certain expectations by users regarding the overall SECURED infrastructure storage modules, and they are presented in Table 27.

Code name	Description
CUR_13	The SECURED framework must support the storage of Unbiased, Anonymous, Synthetic Data in a large scale
CUR_14	The SECURED framework must have a dedicated module for storing Unbiased, Anonymous, AI Models
CUR_15	A comprehensive inventory of Health Data sources (Knowledge Graph and Data catalogue) must be available as part of the SECURED framework.
CUR_16	Authorized Users must be allowed to parse/download Datasets / AI Models, following the Legal limitations of their region/country as well as the relevant legislation (some registered Geolocation information must be also available).
CUR_17	The SECURED framework may store/retrieve information regarding User roles.
CUR_18	The SECURED framework must support Hybrid data integration from various sources.
CUR_19	The SECURED framework must provide mechanisms for granular dataset access control
CUR_20	The SECURED framework must support/accommodate data access requests from registered users

Table 27 – Data Storage Requirements

7.3.1.4 User-Interface (UI) Requirements

A user interface can be defined as the space where interactions between humans and machines occur. The goal of this interaction is to allow effective operation and control of the machine from the human end, while the machine simultaneously feeds back information that aids the operators' decision-making process. Generally, the goal of user interface design is to produce a user interface that makes it easy, enjoyable and efficient to operate a machine in the way which produces the best possible results. This generally means that the operator needs to provide minimal input to achieve the desired output, and also that the machine minimizes undesired outputs to the user. Modern-day users are familiar with user interfaces and compared to previous generations of users, their expectations are significantly raised. The obtained feedback indicated that for the design of the user interface of the SECURED Platform, the requirements listed in Table 28 should be taken into consideration:

Code name	Description
CUR_21	The Content must be displayed and presented properly
CUR_22	The UI must be really easy to Navigate
CUR_23	The interface must be simple but not simplistic. Information must be presented without omitting critical elements or being out of context
CUR_24	The UI must be responsive
CUR_25	The platform must include a mechanism for users to provide necessary feedback
CUR_26	The UI must have a purposeful layout, allowing user guidance to obtain the necessary information
CUR_27	The UI must have a strategical usage of colour and texture
CUR_28	The UI must provide relevant and up-to-date help information
CUR_29	The UI must follow a User-centric approach

Table 28 – User Interface

7.3.1.5 Overall SECURED Platform Requirements

System requirements is a statement that identifies the functionality that is needed by a system in order to satisfy the customer's requirements. Failure to meet system requirements may result in installation, performance and most definitely customer experience issues. The former may prevent a device or application from getting installed, whereas the latter may cause a product to malfunction or perform below expectation or even to hang or crash. As for the third category, it is of paramount importance since it may compromise user satisfaction, and consequently lead to a smaller customer base for a product, platform or service. Under the auspices of SECURED Project, users provided a detailed list of expectations when it comes to the core platform's functionality, the features it needs to support as well as the performance standards it needs to maintain in an end-to-end manner. The requirements that need to be developed are evaluated and presented in Table 29.

Code name	Description
CUR_30	It must be able to query the Data Storage block for gathering information (Dataset information obtained through API calls)
CUR_31	It should be able to query the Analytics block (APIs) for gathering analytics/measurements
CUR_32	It must be able to retrieve Geolocation Information by accessing User Registry (Active Directory) or whichever repository the User Data/Info is stored
CUR_33	It must generate recommendations at the occurrence of datasets that could be included/downloaded/used based on the geolocation/use case of the specific user
CUR_34	It must log events that are considered relevant to end-user application's purposes (i.e. report generation, legal regulations which led to a specific suggestion)
CUR_35	It must push notifications in case policies/regulations/legislation changes to users that may have already downloaded datasets which no longer should utilize
CUR_36	It must expose and API for accessing events
CUR_37	It should allow the specification of customized rules for the attributes it monitors, and correlates user needs/requirements, while in the same time preserves legal compliance
CUR_38	It should allow to define a logical group of rules for simplified management and visualization purposes
CUR_39	It should send notification via email
Core_40	It should assist the end user in creating new rule instances by offering an appropriate API, providing a tentative rule instance filled in with default values (e.g. threshold values) as well as parameter values customized based on the user query.

Table 29 – Platform Requirements

7.3.2 Use-Case adaptations of the SECURED preliminary Architecture

7.3.2.1 Use-Case 1: Real-Time tumor classification

Use-case 1 is focused on the privacy-preserving processing of health data needed at the surgery operating room performed in a untrusted (from a privacy perspective) High Performance Computing (HPC) site. More specifically, during surgery functional Ultra-Sound (fUS) data collected live from a patient undergoing brain surgery for tumor extraction need to be combined (and processed) with MRI data that have been collected by the hospital radiology department on a previous day. Due to the complexity of the computation, the volume of involved data and the need for fast results implies that the processing of the involved data cannot be done within the operating room but it is performed offsite in the EMC HPC Cluster (denoted as CUBE). Note that the computation result is expected in less that 5 min from the provided input in order to be useful/accurate for the doctor/surgeon during surgery. Furthermore, since the MRI data is collected by another hospital department (the radiology department), due to privacy policies, they are not to be shared outside the department apart from being just visualized for the involved doctors. So, while the operating room and the CUBE HPC cluster belong to the same department and they can exchange data (i.e. fUS data in the use-case 1), MRI data cannot be shared and thus processed by the CUBE HPC cluster. The above described overall use-case as well as the addition that can be introduced using the SECURED solution are presented in Figure 11.

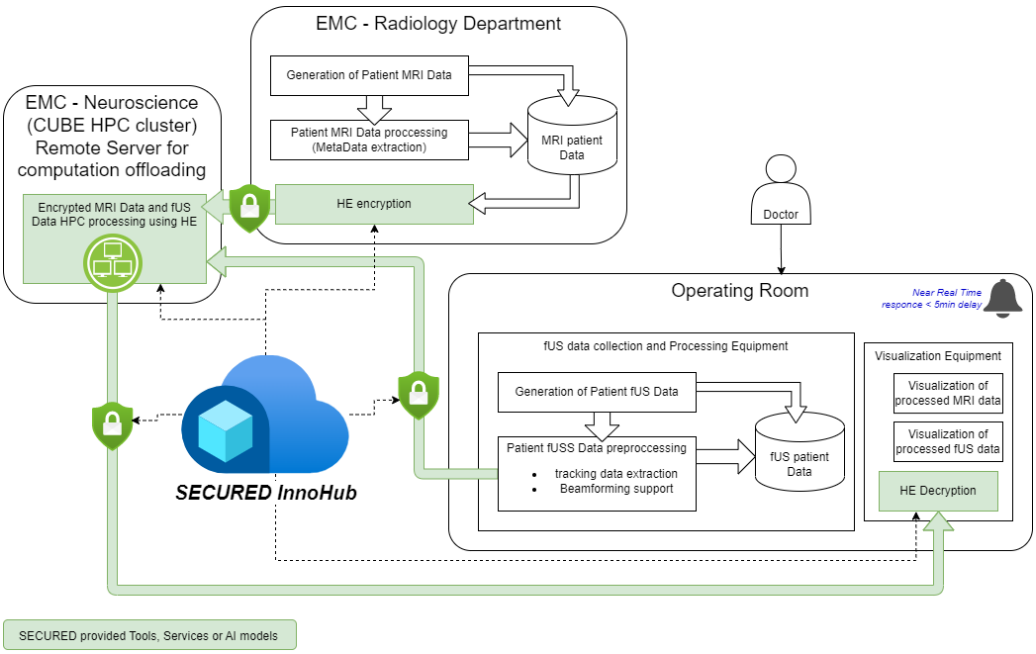

Figure 11 – An overview of Use-Case 1 and its interaction with the SECURED solution

Initially, the patient visits the radiology department of the hospital and gets necessary MRI scans a few days before the surgery. These patient data are stored in a local database and are processed using AI techniques in order to extract valuable features for the actual surgery (as well as to reduce the size of the MRI data to be used). Then, using the SECURED SMPC Engine and more specifically the HE module the MRI data (and metadata) are encrypted using HE in order to be send for processing in the CUBE HPC cluster. The above process is not time critical and does not have near real time constrains. During surgery, the operating room has conceptually two types of equipment involved in the use-case 1: the fUS data collection and processing equipment and the Health data visualization equipment. For both types of equipment their main user is the doctors/surgeons performing the surgery at the Operating Room. The fUS data collection and processing equipment is used in order to collect fUS data during surgery and are preprocessed to extract tracking information and also provide support for image beamforming [605]. The preprocessing outcomes are then stored locally and are forwarded through a secure channel to the CUBE HPC cluster for further processing in combination with the homomorphically encrypted MRI data. The outcome of this HE processing is then forwarded to the Visualization Equipment of the Operating Room where they are HE decrypted and visualized in monitors inside the Operating Room. The above discussed Operating Room processes are time-critical and require near-real time response (the computation results should be available in less the 5 minutes from taking the fUS measurements).

The above use-case 1 analysis as expected, involves several components of the SECURED architecture. Those components are presented in Figure 12 where we highlight in green dashed boxes the utilized SECURED architecture components. SECURED components that are not needed in use-case 1 are grayed out in the figure.

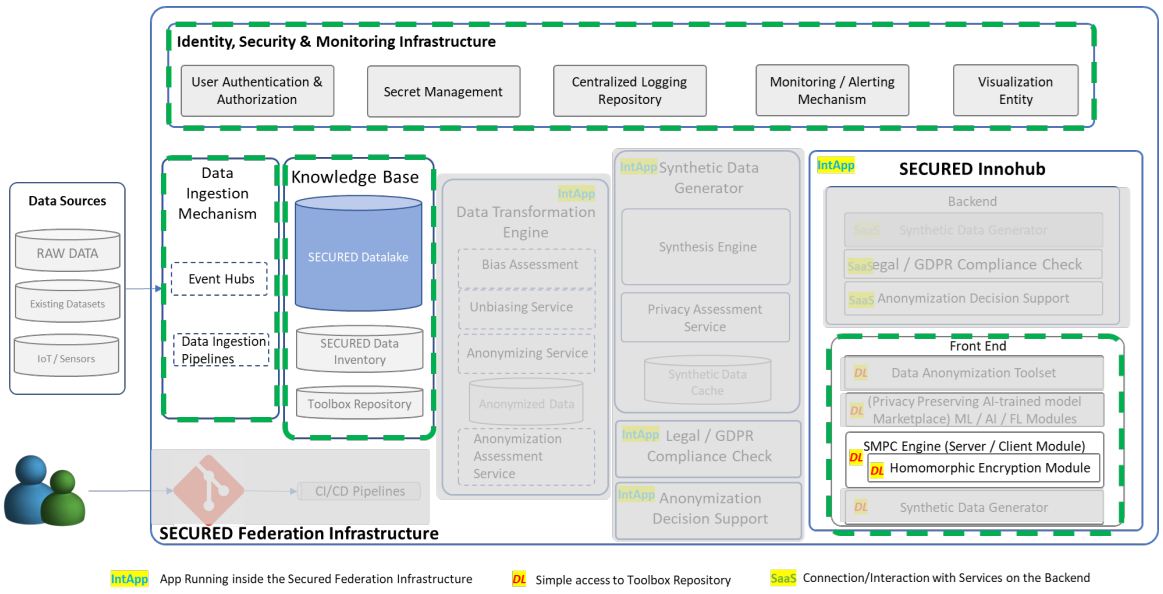

Figure 12 – SECURED component (in dashed green) utilized by Use-Case 1

7.3.2.2 Use-Case 2: Telemonitoring for children

Use-case 2 is focused on detecting anomalous characteristics of health timeseries data collected through various health monitoring sensors using ML/DL. The anomaly detection can be performed within a hospital premises but also at a child patient's house where its health is monitored after leaving the hospital. To realize the use-case we can assume that there is a training phase to create the proper ML/DL model and an inference phase where the created ML/DL model is used to detect anomalies. As depicted in Figure 13, viewing the use-case from one hospital perspective (denoted as Hospital A), we assume that this hospital requires the assistance of other hospitals to train ML/DL models. That is, Hospital A aims to utilize data from other hospitals (eg. Hospital B and Hospital C) in order for all of them to collectively train ML/DL models that eventually all of them will use internally. From a technology perspective this constitutes a typical SMPC scenario. Given the above specifics, in use-case 2 we can also consider that some of the hospitals wants to enhance their health data pool by including synthetically generated data apart from the real health data. In Figure 13, a multiple hospital ML/DL assisted patient tele-monitoring system is presented. Each hospital has installed proper sensors on patients to collect a broad range of timeseries health data including ECG, heart rate, oxygen saturation, respiratory rate etc. The collected data are stored locally at each hospital premises and are viewed in a hospital client application (in Figure 13 the client application of Hospital A is visible only). The data are not directly associated with the patient (i.e they are anonymized) but only with the patient ID and the bed number. Apart from making the data visible to a doctor or nurse (users of client application) the hospital wishes to make available to its users the capability to identify early on anomalies that may be associated with possible patient health issues. Thus ML/DL for anomaly detection is needed in the hospital client application for inference of such anomalies. To train however such ML/DL models, the hospital needs high data volumes and requires the assistance of other hospitals. ML/DL model training needs to be made using entities (various hospital health data hubs) that are not legally allowed to share health data. To achieve this, each hospital could download and use as a tool (on premise) the SECURED data Anonymization toolset in order to evaluate the anonymity and bias of their dataset and then setup the SECURED Secure Multiparty Computation Engine in order to collaboratively compute/train a common ML/DL model. The process will be repeated for all involved hospitals and eventually all the hospitals will use the SECURED SMPC to train ML/DL models using their own training dataset as well as the training datasets (while maintaining privacy using SMPC/HE) from all other the hospitals. The end outcome will be a common, for all hospitals, ML/DL trained model that is going to be registered to the SECURED Innohub and be used for inference by the client application of each hospital.

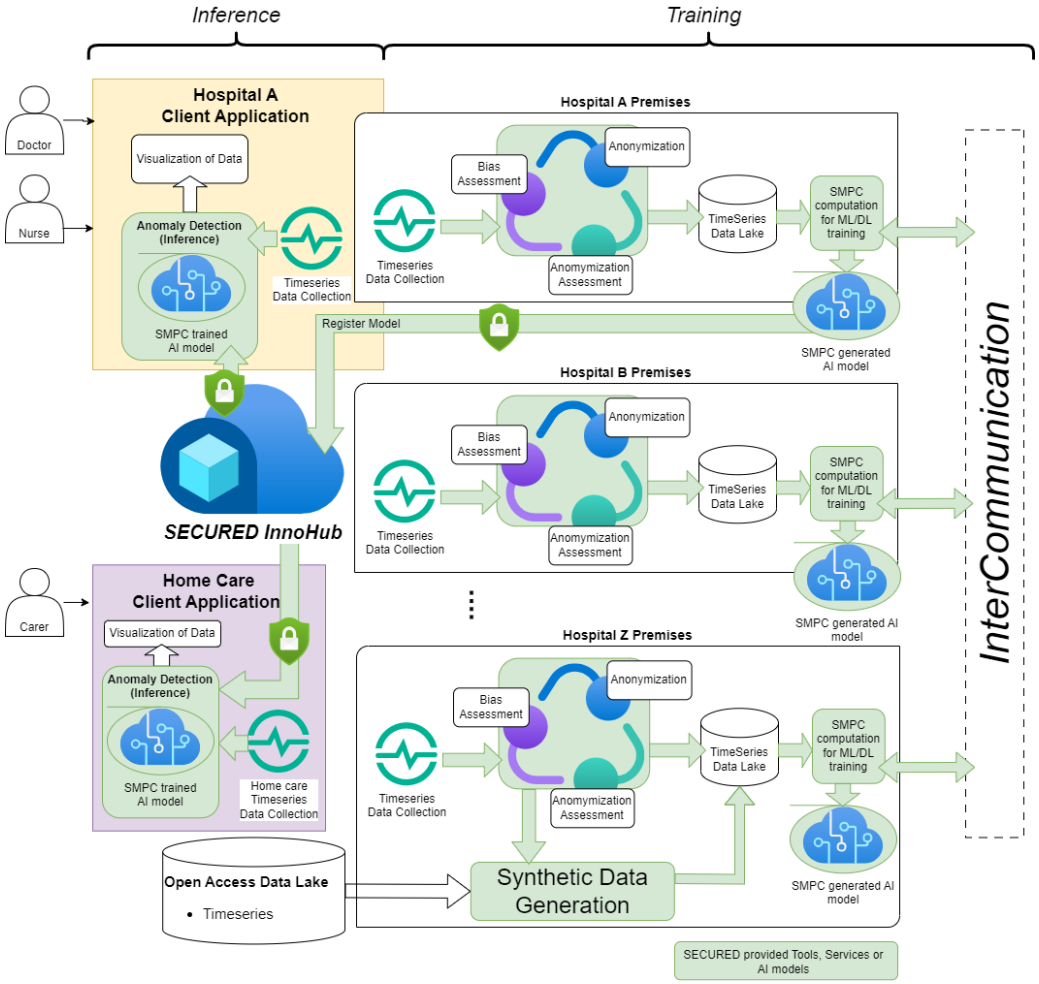

Figure 13 – An overview of Use-Case 2 and its interaction with the SECURED solution

This use-case can be further extended by adopting more advanced feature of the SECURED solution like privacy-preserving Federated Learning for continuous training/update of the ML/DL models. In such a scenario, that still closely resembles the one in Figure 13, one of the hospitals can act as a FL server while the other hospitals can act as clients. In such a variation, the SMPC engine is used in combination with the SECURED FL modules to implement the ML/DL training and update.

In addition to the above, the Figure 13 use-case includes anomaly detection based inference for home-care telemonitoring. In such a scenario, we assume that a child is monitored at his/her home though various installed medical sensors and medical data are collected to an aggregation point at the home premises. The data are then fed into a Home Care Client Application that identifies anomalies and visualizes the data and ML/DL results. To achieve that, the Home Care client application registers to the SECURED Innohub and downloads the already trained ML/DL anomaly detection models (by the various hospitals scenario of use-case 2 discussed in the previous paragraphs) that are already included in the SECURED Innohub.

The above use-case 2 analysis as expected, involves several components of the SECURED architecture. Those components are presented in Figure 14 where we highlight in green dashed boxes the utilized SECURED architecture components. SECURED components that are not needed in use-case 2 are grayed out in the figure.

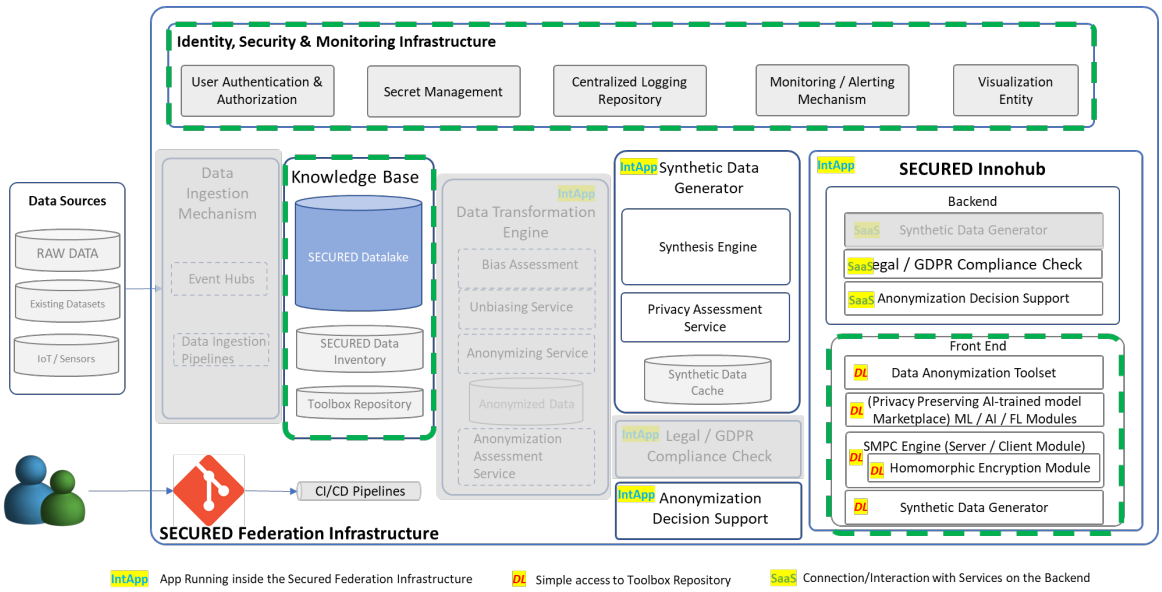

Figure 14 – SECURED component (in dashed green) utilized by Use-Case 2

7.3.2.3 Use-Case 3: Synthetic-data generation for education

Use-case 3 is highly focused on the synthetic health data generation for educational purposes. The use-case considers three different modalities of health data:

- imaging modality with 2D/3D mammography images, MRI scans, ultrasound scans, colon, liver, lung Whole Slide Image (WSI) etc.;
- timeseries modality with heart rate, ECG, CTG timeseries;
- EHR modality with text related and tabular data.

The synthetic data generation process should consider data from the use-case partner private Health Data Lake as well as health data from third party open access data lakes. The synthetic data generation process should make sure that the data that are used as input to this process are properly anonymized and are unbiased. Similarly, the synthetically generated data should also be anonymous and unbiased. The currently identified involved users in the use-case are the administrators of the system, the educators, the evaluators (usually doctors) that assess the quality of the synthetic data and the students.

The use-case can use the SECURED solution in two different ways. The first way the use-case provider employs the SECURED Innohub relevant services as presented in Figure 15. Thus, in this approach the synthetic data generation takes place off-premise and is provided as a service to the use-case provider. However, the data to be transferred to the SECURED Innohub in order to be fed to the synthetic data generation service, should be properly anonymized on-premise (they cannot leave their private data lake otherwise). Thus, the proper usage of the SECURED solution in the above described approach, as depicted in Figure 15, is the following. Firstly, the use-case provider (the admin user) is downloading on-premise, the Data Anonymization toolset which operates as a Data Transformation Engine but offline, on use-case provider premises. This toolbox will locally assess the privacy status (bias, resistance to de-anonymization) of the private, in-house data sets (for all the expected modalities) and perform re-anonymization if needed as well as bias assessment/un-biasing. Eventually, when the process has been successfully completed (the datasets are properly anonymized (i.e low risk of anoynmization vulnerabilities discovered through the SECURED anonymization assessment) and are unbiased), the online phase will initiate. In this phase, the anonymized data are uploaded to the SECURED Innohub (eventually they are also made available for the relevant research community with appropriate access rights, determined by the legal documents as described in D1.2) along with any relevant open access data set. The provided inputs are ingested by the synthetic data generator that performs synthetic data generation and privacy assessment. The latter, provides the same functionality as the Data Anonymization toolset (i.e

anonymization and bias assessment and re-anonymization if needed). Finally, the anonymized results are registered in the SECURED Knowledge Base and are provided to the use-case provider.

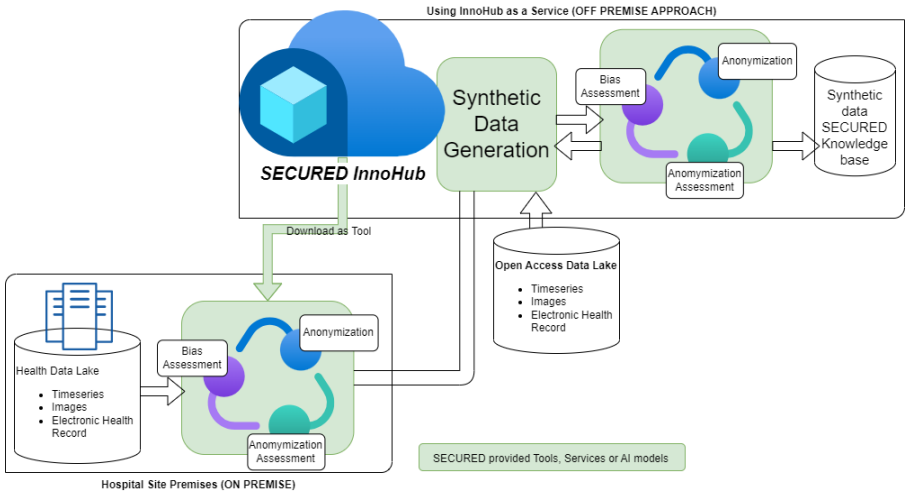

Figure 15 – An overview of Use-Case 3 when off-premise computation is involved and its interaction with the SECURED solution

The second way that SECURED solution can be employed in use-case 3 is by downloading on the use-case provider premises all the necessary tools from the SECURED Innohub in order to perform synthetic data generation locally, as depicted in Figure 16. Initially, the use-case provider (the admin user) downloads and installs on-premise the Data Anonymization toolset in order to anonymize, assess the anonymization and bias of the dataset and potentially re-anonymize them after the synthetic data generation. Similarly, the use-case provider downloads and installs the synthetic data generator engine as a tool. Eventually, the local health data are fed into the locally deployed synthetic data generation engine along with external open access datasets and the synthetically generated data are stored locally on the use-case provider premises after passing through the Data Anonymization toolset to evaluate and enhance (if needed) their anonymity or remove any existing bias. The SECURED Innohub Knowledge base must be updated in order to register the synthetically generated data.

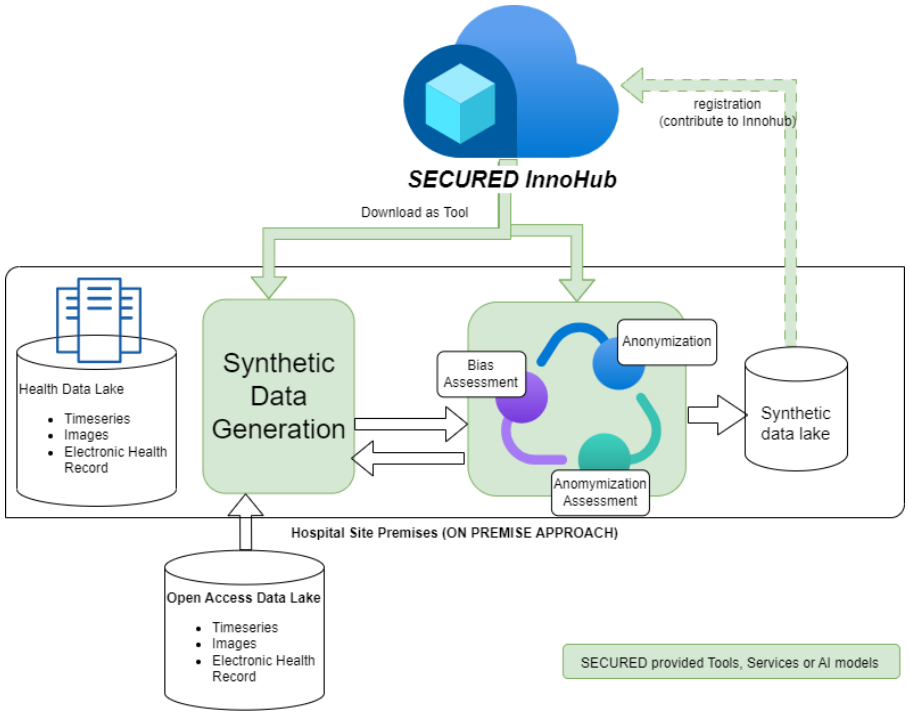

Figure 16 – An overview of Use-Case 3 where all computations are performed on-premise and its interaction with the SECURED solution

Based on the above analysis of the use-case and the use-case interactions with the SECURED solution, in Figure 17 we highlight in green dashed boxes the SECURED architecture components that are needed for

use-case 3. SECURED components that are not needed in use-case 3 are grayed out in the figure.

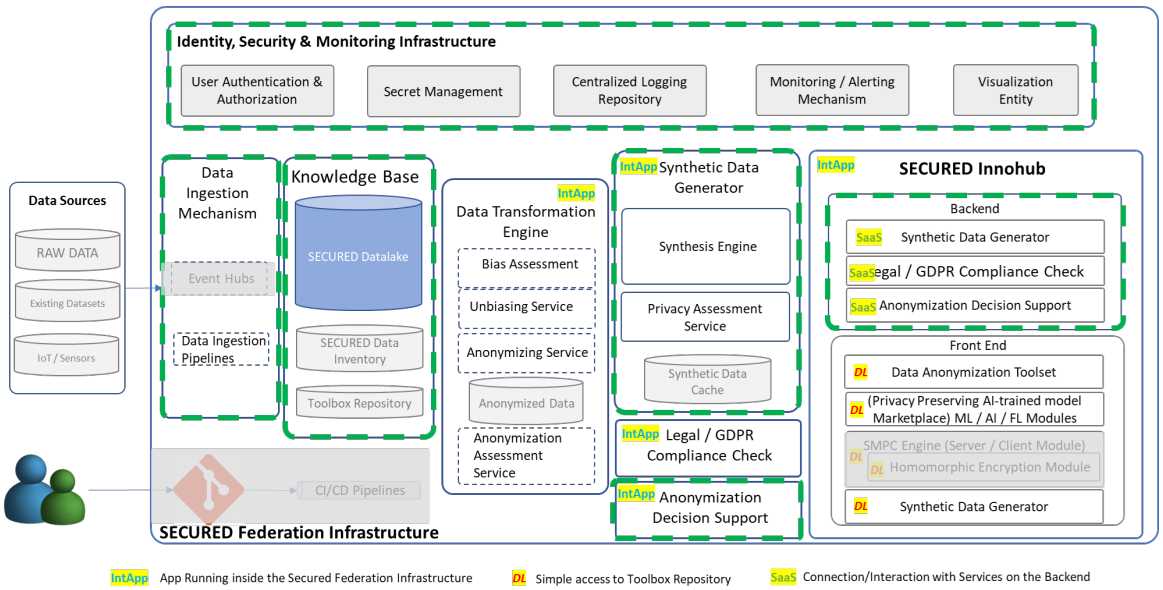

Figure 17 – SECURED component (in dashed green) utilized by Use-Case 3

7.3.2.4 Use-Case 4:Access to genomic data

Use-case 4 is focused on the privacy enabled processing of genomic data with the EHR of a patient. Practically, as depicted in Figure 18, the use-case assumes that the hospital has securely stored genomic data for a large pool of people and that the same or another hospital has stored the EHR of some of the people in the genomic data lake. The use-case assumes that a doctor using some client application (on a local Personal Computer or a mobile phone) is able to access and update a patient's EHR (that exists in the EHR data lake) and eventually identify the polygenic risk of this patient for various types of cancer. Eventually, there are two factors that need to be evaluated to extract the correct polygenic risk, that is the genetic risk (the baseline risk) and the environmental impact to this baseline risk as this is captured by a patient's recorded health data (in the EHR). The processing of the two data flows is performed by the Onco AI solution that extracts the baseline genetic risk from the genomic data and using an AI-based mechanism combines this information with a patient's EHR to provide the adapted, updated cancer risk of ovarian cancer considering the patient's environmental characteristics. This processing (i.e. Onco AI) by combining data from two different sources that don't want to fully disclose data introduces the need for privacy-preserving processing and thus it involves the realization of some SMPC or HE scheme. Note that processing is performed off hospital premise and thus the data (genome and EHR have to be provided in privacy-preserving manner. Eventually, the outcome of the processing need to reach the client application of the patient's doctor where they will be visualized in order for the doctor to be able to explain the results to the patient in order to together decide the best course of action to battle cancer.

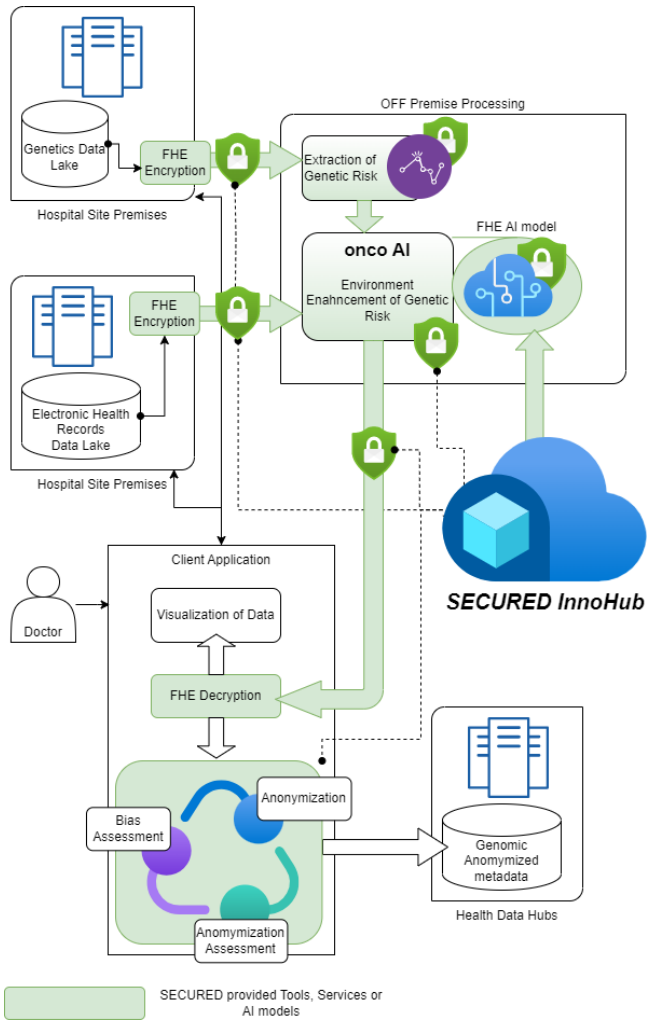

Figure 18 – An overview of Use-Case 4 and its interaction with the SECURED solution

In Figure 18, a preliminary SECURED solution enhanced adaptation of the use-case is provided. The involved data of the use-case, i.e. the genome data and the EHR data are encrypted using the SECURED SMPC engine that include the necessary HE modules to handle the data encryption at the source (in each hospital or data lake). The data leave their respective data lakes after been encrypted with HE and reach their processing point without leaking any confidential/private information. At the processing point, using again the SECURED SMPC engine (and the HE module) the data are encrypted in two phases. In phase 1, the extraction of the genetic risk is made only from the encrypted genomic data. In phase 2 the SMPC/HE-enabled onco AI tool is used in order to align the encrypted genetic risk into the environmentally enabled risk for cancer. All data inputs, intermediate values and outputs are encrypted during the overall risk extraction process. Since the onco AI is using ML or DL models, such trained and FHE enabled models can be provided from the SECURED Innohub and the offered SECURED knowledge base. Eventually, the final outcome is transmitted to the doctor client application, it is decrypted using the SECURED SMPC engine (the FHE module) and is visualized by the doctor. Additionally, the doctor may decide that some of the provided, decrypted results should be made available to the research community as long as they are properly anonymized. This process can be done using the SECURED Data transformation engine or the relevant toolset which consists of an annonymization mechanism, a bias assessment mechanism and the anonymization assessment mechanism. Eventually, failure to comply with the anonymization assessment criteria will initiate a new anonymization round with different configuration that eventually after anonymization assessment will provide the necessary anonymity status that is required by the use-case provider. In Figure 19 we highlight in green dashed boxes the SECURED architecture components that are needed for Use-case 4 based on the above provided description. SECURED components that are not needed in use-case 4 are grayed out in the figure.

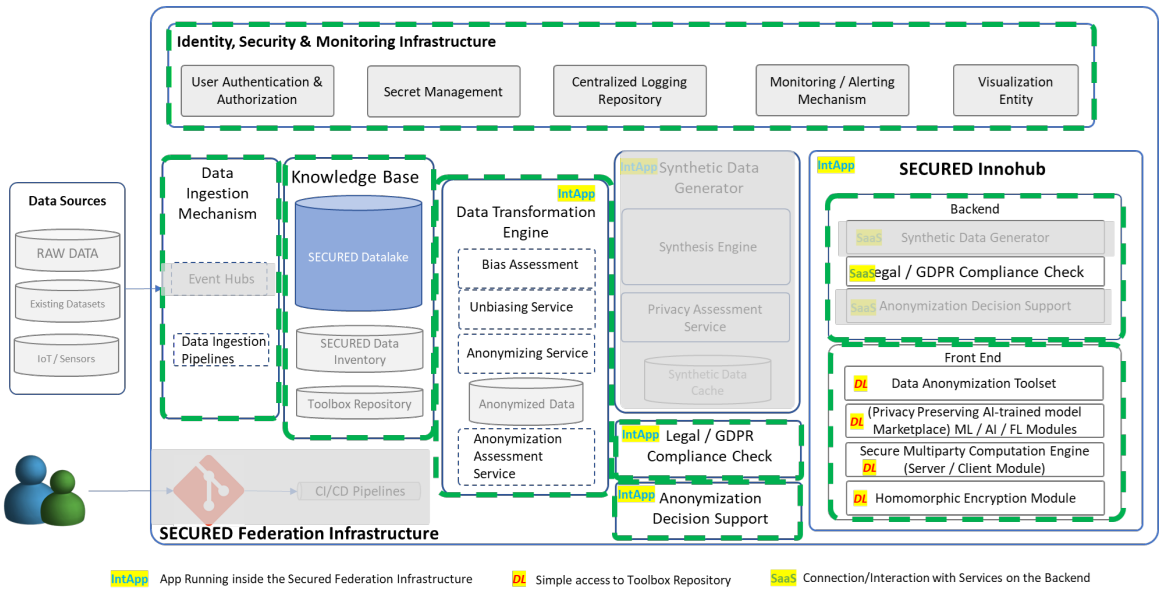

Figure 19 – SECURED component (in dashed green) utilized by Use-Case 4

7.4 Technical requirements of Preliminary SECURED Architecture components

This section outlines the technical requirements extracted from the preliminary activities of T4.1, the user journey and process mapping that has been performed following the guidelines of subsubsection 7.1.2.2, the interaction with the use-case providers and their respective use-case as this is described in subsection 7.3.2. Furthermore, we have taken into consideration the analysis in Sections 3 to 6 of this State-of-the-Art document and the SoTA Gaps that have been documented there. The above have been viewed under the SECURED project objectives as those have been presented in the DoA as well as the identified preliminary SECURED architecture components (and the Architecture as a whole) presented at Figure 10. Apart from the above, we have also taken into consideration the dominant integration technologies on the market that can help us provide an integrated SECURED solution (also depicted in the preliminary SECURED Architecture) and have included them in the technical requirement specifications.

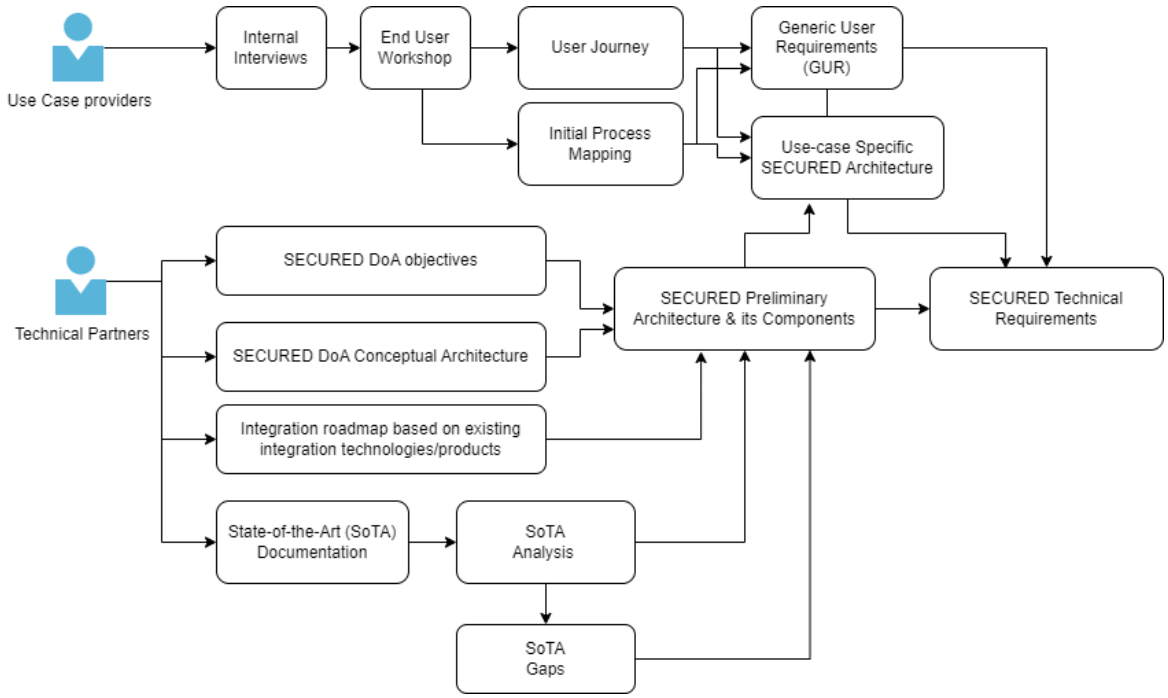

Figure 20 – The SECURED Technical Requirements Documentation Approach

The overall approach that has been followed is illustrated in Figure 20. Using the described approach in the Figure, we aim to ensure that all technical requirements of the project are well-defined and aligned with the needs of the stakeholders and end users, while also being feasible and achievable given the current state of the technology and State of the Art research. The Technical requirements are presented in the following subsections per Preliminary Architecture component or group of components as those are presented in Figure 10 and are described in subsection 7.2. Each Technical Requirement is encoded using the legend of the following Table (Table 30).

Template Field	Description
ID	A unique ID, in the form REQ-TASK-CATEGORY-PRIORITY-# Example: “REQ-PLAT-SEC-M-001”
Task	**TASK = DATA|DPROC|AIxL|PLAT|LEG|SHW|DEV|PI** DATA : Data Handling / Synthetic Data Generation / (De)Anonymization DPROC: SMPC / HE AI-xL : Artificial Intelligence / Federated Learning PLAT: Platform/Deployment / Architecture Requirements LEG : Legal Compliance / Trust guarantees and risk analysis SHW : Specialized Hardware Requirement DEV: Specialized System & Software Development PI: Physical Infrastructure (COTS)
Requirement Type	**ReqType = FUNC|NFUNC** Define if this is a Functional or Non-Functional Requirement

Category	**TYPE = AVL|USE|REL|SEC|PERF|COMP|MAINT | PORT | PRIV** AVL: Availability USE: Usability REL: Reliability SEC: Security PERF: Performance/Efficiency COMP: Compatibility MAINT: Maintainability PORT: Portability PRIV: Privacy
Priority (MoSCoW equivalent)	This allow to identify the priority of the requirements. It can be updated in the future iterations of the Requirements: M: Must-have. **Mandatory** requirement. S: Should-have. **Desirable** requirement. O: Could-have. **Optional** requirement. P: Will-not-have. **Possible** future enhancement
Short Title	A meaningful and not too long title that characterizes the requirement.
Description	General description, a brief text explaining the requirement, including the objectives where necessary.
Associated Core User Requirement (CUR)	Enumerate the associated Core user requirements that have been presented in subsection 7.3.1

Table 30 – Technical Requirements Encoding Legend

7.4.1 Technical Requirements for Platform Deployment, underlying Infrastructure and module Positioning

REQ-PLAT-PORT-M-01	**Short Name:**	Integration of experimenter's complimentary components			
Description	The platform should support the integration of virtual or physical infrastructure component brought by the experimenters during the open call, which will allow them to conduct the relevant experimentation and validate KPI related objectives. This may be permitted under certain conditions and safety regulations and is strongly dependent on the actual functionality provided by the component. Components which may compromise the SECURED Hub security cannot be allowed to be deployed.				
Priority	Mandatory	**Type**	Functional	**CUR**	30 - 40

REQ-PLAT-USE-D-02	**Short Name:**	Application/Service deployment at the Edge			
Description	When required by Application/Service, Edge Infrastructure should be available for deployment (Edge can be on premise, on local/private cloud environments or potentially public cloud). Edge Infrastructure should support deployment of containerized Application/Services (i.e. Docker containers) using industry standard tools such as Terraform.				
Priority	Desirable	**Type**	Functional	**CUR**	13 - 15

REQ-PLAT-USE-M-03	**Short Name:**	Application/Service deployment infrastructure support		
Description	Compute infrastructure used for Deployment must provide APIs and Tools to Deploy and orchestrate function virtualisation via virtual machines or containers and virtual networking			
Priority	Mandatory	**Type**	Functional	**CUR** 13 - 15

REQ-PLAT-REL-M-04	**Short Name:**	Application/Service Private deployment support		
Description	Downloaded Application/Service binaries/artifacts, should be able to be easily instantiated in Private Cloud Environments (K8s clusters) using standardized (K8s) software/tools			
Priority	Mandatory	**Type**	Functional	**CUR** 13 - 15, 30, 31

REQ-PLAT-REL-M-05	**Short Name:**	Application/Service Public deployment support		
Description	Downloaded Application/Service binaries/artifacts, should be able to be easily instantiated in Public Cloud Environments (K8s clusters) using standardized (K8s) software/tools			
Priority	Mandatory	**Type**	Functional	**CUR** 13 - 15, 30, 31

REQ-PLAT-AVL-D-06	**Short Name:**	Cloud native elasticity (scale out/scale in) support		
Description	Application/Service should incorporate standardized scale up/out procedures in containers/VMs in order to support high volumes of traffic and number of users.			
Priority	Desirable	**Type**	Non-Functional	**CUR** 13 - 15, 30, 31, 37

REQ-PLAT-SEC-M-07	**Short Name:**	Security, integrity and reliability of communication links between Application/Service and interconnected entities		
Description	All connectivity links between the Application/Service and the inteconnected architectural components should be secure. The communication link between the Application/Service and the control plane APIs should be protected in terms of confidentiality and integrity.			
Priority	Mandatory	**Type**	Non-Functional	**CUR** 13 - 15, 30, 31, 36, 37

REQ-PLAT-PRIV-M-08	**Short Name:**	End to end encrypted communication		
Description	Communications through the network should be secured, with end-to-end encryption			
Priority	Mandatory	**Type**	Functional	**CUR** 10 - 12

REQ-PLAT-PRIV-M-09	**Short Name:**	Granular User Access support		
Description	The Application/Service should support various levels of authorization (remote user / centralized user / administrator etc.			
Priority	Mandatory	**Type**	Functional	**CUR** 6 - 9

REQ-PLAT-SEC-M-10	**Short Name:**	Strong User Authentication support		
Description	The infrastructure should support mutual strong authentication between exposed APIs and Application/Service (as well as with the corresponding vertical applications).			
Priority	Mandatory	**Type**	Functional	**CUR** 6 - 9, 17

7.4.2 Technical Requirements for delivering contemporary services related with User Authentication & Authorization

REQ-PLAT-SEC-D-11	**Short Name:**	Single-Sign On / Out			
Description	Users Authenticate with the Authentication and Authorization entity rather than individual applications. This means that the infrastructure only has a single entity to deal with login forms, authenticating and storing users. Once logged-in to AA entity, users don't have to login again to access a different application or service inside the SECURED ecosystem. In addition, the AA entity should provide single-sign out, meaning that users only have to logout once to be logged-out of all applications and services.				
Priority	Desirable	**Type**	Functional	**CUR**	6 - 9, 17

REQ-PLAT-SEC-M-12	**Short Name:**	User Federation Support			
Description	The Authentication & Authorization entity must support interconnection to existing LDAP or Active Directory servers. It should also be possible to implement additional providers to integrate users found in other repositories i.e. a relational database				
Priority	Mandatory	**Type**	Functional	**CUR**	6 - 9, 16, 17

REQ-PLAT-SEC-M-13	**Short Name:**	Standard Protocol and Identity Brokering Support			
Description	The Authentication & Authorization entity must be based on standard protocols and provides support for OpenID Connect, OAuth 2.0, and SAML. This will allow user Authentication with existing OpenID Connect or SAML 2.0 Identity Providers. Ideally, the identity brokering should be configurable through the administration console.				
Priority	Mandatory	**Type**	Functional	**CUR**	6 - 12, 16, 17

REQ-PLAT-SEC-D-14	**Short Name:**	Administration Console			
Description	Every configuration action of the Authentication & Authorization entity must be made through the admin console. Administrators must be able to centrally (i) enable and disable various features, (ii) configure identity brokering and user federation, (iii) create and manage applications and services (iv) define fine-grained authorization policies and (v) manage users, including permissions and sessions.				
Priority	Desirable	**Type**	Functional	**CUR**	1 - 9, 13 - 18

REQ-PLAT-SEC-M-15	**Short Name:**	RBAC Authorization Service and Customized Policy Support			
Description	The Authentication & Authorization entity must support Role-based access control (RBAC), in order to restrict network access based on the roles of individual users within the SECURED Infrastructure. To further enhance security the Authentication & Authorization entity must also support fine-grained authorization services, allowing permission management for all services through a central point thus empowering the administrators to define accurately who is allowed to access the various entities through well-defined policies.				
Priority	Mandatory	**Type**	Non-Functional	**CUR**	7, 13 - 18

REQ-PLAT-SEC-P-16	**Short Name:**	Account Management Console / Environment (OPTIONAL)			
Description	The Authentication & Authorization entity must provide an account management console accessible by users for managing their own accounts. Users should be able to update the profile, change passwords, manage sessions and last but not least view the account history. If you've enabled social login or identity brokering users can also link their accounts with additional providers to allow them to authenticate to the same account with different identity providers.				
Priority	Possible	**Type**	Non-Functional	**CUR**	1 - 9

REQ-PLAT-SEC-P-17	**Short Name:**	2(M)Factor Authentication Support (OPTIONAL)			
Description	The Authentication & Authorization entity could support two(multi)-factor authentication to increase security. 2(M)FA should be configurable by the users from the Account Management Console.				
Priority	Possible	**Type**	Functional	**CUR**	1 - 3, 9

7.4.3 Vault and Secrets Management

REQ-PLAT-SEC-M-18	**Short Name:**	Secrets Management			
Description	The Vault must be able to Securely store and tightly control access to tokens, passwords, certificates, API keys, and other secrets. In addition, the Vault must also support K8s-related secret management (i.e KV Secrets Engine, Database Credentials, Kubernetes Secrets)				
Priority	Mandatory	**Type**	Functional	**CUR**	10, 12, 13 - 18, 30, 32

REQ-PLAT-SEC-M-19	**Short Name:**	Key Management			
Description	The Vault must support the creation and control of the encryption keys used for all types of data encryption (data at rest/in transit).				
Priority	Mandatory	**Type**	Functional	**CUR**	10, 30, 32

REQ-PLAT-SEC-M-20	**Short Name:**	Certificate Management			
Description	The Vault must support the provisioning, management, and deployment of public and private Transport Layer Security/Secure Sockets Layer (TLS/SSL) certificates which could be used to secure external/internal connected resources.				
Priority	Mandatory	**Type**	Functional	**CUR**	10, 30, 32

7.4.4 Technical Requirements for delivering a Centralized Logging Repository

REQ-PLAT-COMP-M-21	**Short Name:**	Source-agnostic data ingestion			
Description	The Centralized Logging Repository must have the ability to ingest and securely store data from various resources. Data storage should be isolated from the ingestion mechanism to avoid data loss.				
Priority	Mandatory	**Type**	Functional	**CUR**	30 - 34

REQ-PLAT-MAINT-M-22	**Short Name:**	Metrics Handling support			
Description	Metrics are numerical values that are collected at regular intervals and describe some aspect of the system at a particular time. The Centralized Logging Repository must support Metrics handling. The entity needs to be able to collect numeric data from monitored resources into a time-series database.				
Priority	Mandatory	**Type**	Functional	**CUR**	34

REQ-PLAT-MAINT-D-23	**Short Name:**	Log Ingestion support			
Description	The Centralized Logging Repository must support Log ingestion, as well as being able to organize log/performance data from monitored resources. For properly do so, it must support a variety of data types that have their own structure.				
Priority	Desirable	**Type**	Functional	**CUR**	34

REQ-PLAT-MAINT-D-24	**Short Name:**	Custom Query Support			
Description	The Centralized Logging Repository must support log queries for analyzing ingested logs. Ideally, an API must be exposed giving the ability to other modules/entities to execute queries on stored logs/metrics.				
Priority	Desirable	**Type**	Functional	**CUR**	34

REQ-PLAT-MAINT-D-25	**Short Name:**	Traces Handling support			
Description	The Centralized Logging Repository must support Traces ingestion. A trace represents the end-to-end journey of a request through a distributed system. By viewing a trace, known as distributed tracing, it is possible to track a complete execution path and identify which part of the code is causing issues like errors, latency concerns or resource availability, all elements of paramount importance for a distributed architecture.				
Priority	Desirable	**Type**	Functional	**CUR**	34

7.4.5 Technical Requirements for implementing a Monitoring / Alerting mechanism

REQ-PLAT-MAINT-M-26	**Short Name:**	Centralized Logging Repository query environment / interface			
Description	Must provide an environment to edit and run log queries against data stored in the Centralized Logging Repository				
Priority	Mandatory	**Type**	Functional	**CUR**	37, 39

REQ-PLAT-MAINT-M-27	**Short Name:**	Metric Alerts support			
Description	Metric alerts evaluate resource metrics at regular intervals. Metrics can be platform metrics, custom metrics, or logs converted to metrics via a customized process (i.e. timestamps). Metric alerts should also be triggered via multiple conditions as well as through dynamic thresholds.				
Priority	Mandatory	**Type**	Functional	**CUR**	37, 39

REQ-PLAT-MAINT-D-28	**Short Name:**	Activity Log Alert support			
Description	Activity log alerts are triggered when a new activity log event occurs that matches defined conditions. Resource Health alerts and Service Health alerts are considered activity log alerts that report on your service and resource health.				
Priority	Desirable	**Type**	Functional	**CUR**	37, 39

REQ-PLAT-MAINT-D-29	**Short Name:**	Customized Alert Rules support			
Description	The Monitoring / Alerting Mechanism should support customized alert rules, which monitor stored data and identify signals that indicates anomaly on the specified resource. For instance, an alert rule that captures a signal may check if the signal meets the criteria of the condition. If the conditions are met, an alert is triggered, leading to an associated action together with an update of the state of the alert.				
Priority	Desirable	**Type**	Functional	**CUR**	35, 37, 39, 40

7.4.6 Visualization Entity Technical Requirements

REQ-PLAT-MAINT-M-30	**Short Name:**	Dashboards and Tables			
Description	Visual representation of data through filtered tables and dashboards				
Priority	Mandatory	**Type**	Functional	**CUR**	21 - 29

REQ-PLAT-MAINT-M-31	**Short Name:**	Log Reports			
Description	Predefined reports for visualizing logs per application type				
Priority	Mandatory	**Type**	Functional	**CUR**	21 - 29, 35, 37

REQ-PLAT-MAINT-M-32	**Short Name:**	Analytics Graphs			
Description	Responsible for providing on-demand general statistics related to predefined attributes in a given time interval				

Priority	Mandatory	**Type**	Functional	**CUR**	21 - 29

7.4.7 Technical Requirements for the DATA Transformation Engine

REQ-PLAT-PERF-M-33	**Short Name:**	Accelerated Data Retrieval Mechanism			
Description	The Data Transformation Engine must incorporate a mechanism for loading data from Datalakes faster, similar to MS Polybase				
Priority	Mandatory	**Type**	Functional	**CUR**	13 - 18

REQ-PLAT-DATA-O-34	**Short Name:**	Common Data Integration Pattern Functionality (ETL/ELT)			
Description	The Data Transformation Engine must be able to align with data functionality following both the ELT as well as the ETL operational mode, namely executing (i) Extraction by connecting to sources, copy data to central store, (ii) Transformation, by process the data for analysis / data transformation at scale and (iii) Loading by moving the data to data lake/warehouse or analytics engines for business insights. On the ELT pattern, raw (native) data is itself loaded (stored) before the transformation phase.				
Priority	Optional	**Type**	Functional	**CUR**	13 - 18

REQ-PLAT-MAINT-D-35	**Short Name:**	Data Pipeline Definition and Execution			
Description	The Data Transformation Engine must support the definition/creation/execution of Data manipulating Pipelines. Pipelines are defined as logical groupings of activities that perform one unit of work. To accommodate proper data manipulation, the Data Handling Module must support activities related to Data Movement, Data Transformation and Data Control.				
Priority	Desirable	**Type**	Functional	**CUR**	13 - 18

REQ-PLAT-DATA-D-36	**Short Name:**	Dataset Identification and Handling			
Description	The Data Transformation Engine must be able to identify data structures which provide a SELECT view into data store. This allows pointing to a specific data subset to use in activity for input and output.				
Priority	Desirable	**Type**	Non-Functional	**CUR**	13 - 18

REQ-PLAT-DATA-M-37	**Short Name:**	Data Flow Mapping			
Description	The Data Transformation Engine must be able to create and manage data transformation graphs that work on any size of data. In addition, the module must be capable of building up reusable libraries of data transformation routines as well as executing the specific routines in pipelines (as activity) for scaling purposes.				
Priority	Mandatory	**Type**	Non-Functional	**CUR**	13 - 18

REQ-PLAT-PERF-M-38	**Short Name:**	Integration Runtime			
Description	The Data Transformation Engine must have the ability to map and properly define the COMPUTE infrastructure that is needed for providing fully managed (i) Data flows (transformation) (ii) data movement (movement) (iii) activity dispatch (route to service).				

Priority	Mandatory	**Type**	Non-Functional	**CUR**	13 - 18, 30 - 35

7.4.8 Technical Requirements for implementing the Data Anonymization functionality of the DATA Transformation Engine

REQ-DATA-PRIV-M-39	**Short Name:**	Data Masking Support			
Description	Data masking refers to the disclosure of data with modified values. Data anonymization is done by creating a mirror image of a database and implementing alteration strategies, such as character shuffling, encryption, term, or character substitution.				
Priority	Mandatory	**Type**	Non-Functional	**CUR**	13 - 18, 30 - 35

REQ-DATA-PRIV-M-40	**Short Name:**	Pseudoanonymization Support			
Description	Pseudonymization is a data de-identification tool that substitutes private identifiers with false identifiers or pseudonyms, such as swapping the "John Smith" identifier with the "Mark Spencer" identifier. It maintains statistical precision and data confidentiality, allowing changed data to be used for creation, training, testing, and analysis, while at the same time maintaining data privacy.				
Priority	Mandatory	**Type**	Non-Functional	**CUR**	13 - 15, 18

REQ-DATA-PRIV-M-41	**Short Name:**	Generalization Support			
Description	Generalization involves excluding some data purposely to make it less identifiable. Data may be modified into a series of ranges or a large region with reasonable boundaries. For example, the house number at an address may be deleted, but make sure the name of the lane does not get deleted. The aim is to remove some of the identifiers while maintaining the accuracy of the data.				
Priority	Mandatory	**Type**	Non-Functional	**CUR**	13 - 15, 18

REQ-DATA-PRIV-M-42	**Short Name:**	Data Swapping Support			
Description	Data swapping – often known as permutation and shuffling – rearranges dataset attribute values so that they do not fit the original information. Switching attributes (columns) that include recognizable values, such as date of birth, can make a huge impact on anonymization.				
Priority	Mandatory	**Type**	Non-Functional	**CUR**	13 - 15, 18

REQ-DATA-PRIV-M-43	**Short Name:**	Data Pertubation Support			
Description	Data perturbation modifies the initial dataset marginally by applying round-numbering methods and adding random noise. The set of values must be proportional to the disturbance. A small base can contribute to poor anonymization, while a broad base can reduce a dataset's utility				
Priority	Mandatory	**Type**	Non-Functional	**CUR**	13 - 15, 18

7.4.9 Data Lake Technical Requirements

REQ-PLAT-DATA-M-44	**Short Name:**	Compatibility with the Hadoop Distributed File System (HDFS)			
Description	HDFS is the de-facto, open-source standard for running parallel analytics workloads at consistent high performance. Any adopted Data Lake solution must be compatible with it for being able to benefit from industry-standard features and capabilities.				
Priority	Mandatory	**Type**	Functional	**CUR**	13 - 18, 30, 33

REQ-PLAT-DATA-M-45	**Short Name:**	Compatibility with standardized Analytics Engines via a dedicated Query Layer			
Description	The Data Lake must be compatible with contemporary analytic engines (i.e. Apache Spark) for big data environments and machine learning.				
Priority	Mandatory	**Type**	Non-Functional	**CUR**	13 - 18, 30, 33

REQ-PLAT-DATA-M-46	**Short Name:**	Data Lake solution must be Data Source Agnostic			
Description	The Data Lake must be capable of storing a huge amount of raw data in a structured, semi-structured and/or unstructured storage format. More specific (i) Unstructured data often correspond to text heavy data with structures that are not uniform (ii) Semi-Structured data is typically data with associated metadata tags, giving it some sort of loosely defined structure and (iii) Structured data refer to data with a rigid format, which includes data residing in traditional databases like SQL servers and analysis services cubes.				
Priority	Mandatory	**Type**	Non-Functional	**CUR**	13 - 18, 30, 33

REQ-PLAT-DATA-M-47	**Short Name:**	Native Data Type Support			
Description	Architectural components, their interaction and identified products should support native data types				
Priority	Mandatory	**Type**	Non-Functional	**CUR**	13 - 18, 30, 33

REQ-PLAT-DATA-D-48	**Short Name:**	Data Lake REST API			
Description	The Data Lake must be able to seamlessly communicate/integrate with all existing technologies. The most efficient way to do so, is through a dedicated REST API, available to all interconnected entities. In addition, the specific API must be used for exposing disposable components.				
Priority	Desirable	**Type**	Functional	**CUR**	13 - 18, 30, 33

REQ-PLAT-DATA-M-49	**Short Name:**	Layered and Isolated Architecture			
Description	The overall Data Lake architecture must contain independent mechanisms for managing data discovery, ingestion, storage, administration, quality, transformation, and visualization. Moreover, the Data Lake design should be driven by what is available instead of what is required. The schema and data requirement is not defined until it is queried. Always keep in mind that Unified operations tier, Processing tier, Distillation tier and HDFS are important layers of traditional Data Lake architecture.				
Priority	Mandatory	**Type**	Non-Functional	**CUR**	37

7.4.10 Technical Requirements for defining the Architecture, certain Development quidelines and the Verification process

REQ-DEV-USE-D-50	**Short Name:**	Software Documentation			
Description	The developer must provide an Application/Service/Function blueprint that describes how the Application/Service/Function is built, deployed and communicates with interconnected entities/services/functional blocks.				
Priority	Desirable	**Type**	Other	**CUR**	30 - 40

REQ-DEV-USE-M-51	**Short Name:**	Main Codebase Repository Requirements			
Description	The main development repository of Application/Service/Function should allow (1) creating branches, (2) commit changes, (3) pull and push and (4) creation and cloning to/from secondary repositories. (5) Different member's options should be available (e.g. owner, contributor, collaborator), (6) ticketing/issues tracking system (7) support different status identification (e.g. draft, internal testing, etc.).				
Priority	Mandatory	**Type**	Other	**CUR**	13 - 18, 30 - 40

REQ-DEV-USE-D-52	**Short Name:**	Verification tools virtualization			
Description	Verification tools should be virtualized (e.g. dockerized) to provide a quick execution of an isolated environment				
Priority	Desirable	**Type**	Non-Functional	**CUR**	13 - 15

REQ-DEV-USE-D-53	**Short Name:**	Exclusive verification tests			
Description	The developer should be able to execute verification and smoke tests in a isolated environment for validating code functionality.				
Priority	Desirable	**Type**	Non-Functional	**CUR**	6, 7, 8

REQ-DEV-USE-D-54	**Short Name:**	Open-sourced validation tools			
Description	Validation tools should be open-sourced, unless specific contractual obligations prevents this for some components or parts of the code				
Priority	Desirable	**Type**	Non-Functional	**CUR**	6, 12

REQ-DEV-USE-D-55	**Short Name:**	Validation framework containerization			
Description	Validation framework should be virtualized (e.g. containerized) whenever possible, to provide a quick execution on an isolated environment				
Priority	Desirable	**Type**	Non-Functional	**CUR**	12

REQ-DEV-AVL-M-56	**Short Name:**	Cloud-native compatibility			
Description	The Validation Framework should support the deployment of cloud-native functions				
Priority	Mandatory	**Type**	Non-Functional	**CUR**	30, 31

7.4.11 SECURED Innohub Technical Requirements

REQ-PLAT-USE-M-57	**Short Name:**	SECURED Toolbox / Software Repository			
Description	A dedicated repository must be available to store, search and download certified Applications Tools and Software that are willingly uploaded (compiled version / artifact)				
Priority	Mandatory	**Type**	Functional	**CUR**	30 - 34

7.4.12 Technical Requirements for API Design

REQ-DEV-REL-D-58	**Short Name:**	SECURED REST API for facilitating component interconnection			
Description	SECURED API should follow the principles of Representational State Transfer (REST) RESTful API design is the process of designing an API that follows the principles of Representational State Transfer (REST), which is the most popular API architecture today. In a RESTful architecture, resources are identified by URIs (Uniform Resource Identifiers), and the client interacts with those resources with standard HTTP methods such as GET, POST, PUT, and DELETE.				
Priority	Desirable	**Type**	Functional	**CUR**	30, 31, 36, 40

REQ-DEV-COMP-D-59	**Short Name:**	SECURED API Security Practices			
Description	SECURED API must follow a security-driven design and therefore: (i) implement quotas and throttling to limit the number of requests to a particular service in order to conserve resources and ensure high availability, (ii) integrate appropriate headers to all API responses, (iii) introduce read and write granularity (iv) support input validation, sanitization and encoding at the method level, (v) support logging and monitoring				
Priority	Desirable	**Type**	Non-Functional	**CUR**	30, 31, 36, 40

REQ-DEV-USE-D-60	**Short Name:**	SSECURED API Documentation			
Description	SECURED API Documentation should include (i) Authentication Instructions (ii) Detailed information about endpoints, operations and resources (iii) examples of common requests and responses				
Priority	Desirable	**Type**	Other	**CUR**	30, 31, 36, 40

7.4.13 Technical Requirements for Third Party integration (Open Call)

REQ-DEV-COMP-M-61	**Short Name:**	Extension - SECURED Infrastructure interaction			
Description	New services shall be able to interact with the SECURED Infrastructure though the open-source SECURED REST API that will be specified in the project.				
Priority	Mandatory	**Type**	Functional	**CUR**	30 - 40

REQ-DEV-COMP-M-62	**Short Name:**	Extension - Remote Operational Control

Description	New services should provide means for remote operational control from the corresponding SECURED Infrastructure entity (Controller / CLI / Admin UI)				
Priority	Mandatory	**Type**	Functional	**CUR**	30 - 40

REQ-DEV-COMP-M-63	**Short Name:**	User authorization to experimentation/development data
Description	SECURED Infrastructure should ensure that third party developers/experimenters are authorized to perform tests isolated, fully independent, without the ability to access other experimenter's data(sets).	
Priority	Mandatory	**Type** Functional **CUR** 30 - 40

REQ-DEV-COMP-M-64	**Short Name:**	Extension – SECURED Infrastructure secure communication
Description	The connectivity link/communication channel between all entities/services shall be secure, potentially with end-to-end encryption.	
Priority	Mandatory	**Type** Functional **CUR** 10 - 12, 30 - 40

REQ-DEV-COMP-M-65	**Short Name:**	Extension - User Authentication
Description	New services should support various levels of authorization (i.e. remote user / centralized user / administrator) and identification	
Priority	Mandatory	**Type** Functional **CUR** 16, 30 - 40

REQ-DEV-COMP-M-66	**Short Name:**	QoS Alerting Mechanism
Description	New services should generate alerts if expected/predefined QoS cannot be reached, in order to trigger adaptation/improvement mechanisms on the SECURED Infrastructure side. The QoS should be the output of monitoring values/performance metrics such as latency, throughput, uptime as well as application-specific KPIs	
Priority	Mandatory	**Type** Non-Functional **CUR** 25, 33, 39, 40

REQ-DEV-COMP-M-67	**Short Name:**	Extension – Behavior Monitoring
Description	New services should be authorized to access exposed APIs based on continuous monitoring (behavior, traffic patterns, queries, etc) provided by the SECURED Infrastructure. Monitored access could be utilized to ensure appropriate behavior or to detect potentially malicious actions (i.e. DDoS-type attacks taking advantage of exposed APIs)	
Priority	Mandatory	**Type** Non-Functional **CUR** 30, 31, 36

REQ-DEV-COMP-M-68	**Short Name:**	Testbed-Experimenter collaboration
Description	Specific details about the verification tests (scope, implementation) must be agreed between the experimenter and the testbed owner	
Priority	Mandatory	**Type** Functional **CUR** 40

7.4.14 Technical Requirements on the Data Anonymization Service and Tools

REQ-DATA-PRIV-M-69	**Short Name:**	Anonymization service and tool for different, heterogeneous Health data types			
Description	In SECURED use-cases we adopt several different types of datasets including health image datasets, health timeseries datasets and EHR datasets. Thus the anonymization Services/Anonymization toolbox should be able to anonymize various different data types.				
Priority	Mandatory	**Type**	Functional	**CUR**	10 - 20

REQ-DATA-PRIV-D-70	**Short Name:**	Anonymization of high data volumes			
Description	The Anonymiation process should be able to work with out excessive delays and with out loss of accuracy for high volume of health datasets of various types (also check REQ-DATA-PRIV-M69)				
Priority	Desired	**Type**	Non Functional	**CUR**	10 - 20

REQ-DATA-PRIV-D-71	**Short Name:**	Offered Anonymization to withstand de-anonymization attacks			
Description	The provided Anonymization services/tools should be able to provide high quality anonymized results that when assessed through the Anonymization Assessment service/tools no vulnerabilities (or low risk vulnerabilities) will only be found.				
Priority	Desired	**Type**	Functional	**CUR**	10 - 20

REQ-DATA-MAINT-M-72	**Short Name:**	Provide report/guarantee of anonymization process			
Description	The Anonymization service/tools should be able to log the history of anonymization/de-anonymization rounds that took place as well as anonymity vulnerabilities that have been discovered and potentially mitigated. This log/report should be part of a privacy guarantee that must be associated with the outcome of the Data Transformation service (related to the anonymized and anonymization assessed dataset as well as and bias that has been removed from the dataset).				
Priority	Mandatory	**Type**	Functional	**CUR**	10 - 20

7.4.15 Technical Requirements on the Anonymization Assessment Service and Tools

REQ-DATA-SEC-M-73	**Short Name:**	Assess anonymized dataset for Timeseries Health Data			
Description	Since Health Timeseries Data are been used by several of the SECURED use-case it is mandatory to be able to assess the anonymity of such datasets and discover potential privacy leakage				
Priority	Mandatory	**Type**	Functional	**CUR**	10 - 20

REQ-DATA-SEC-M-74	**Short Name:**	Assess anonymized dataset for Image Health Data

Description	Since various types of Health Image Data are been used by several of the SECURED use-case it is mandatory to be able to assess the anonymity of such datasets and discover potential privacy leakage				
Priority	Mandatory	**Type**	Functional	**CUR**	10 - 20

REQ-DATA-SEC-M-75	**Short Name:**	Assess anonymized dataset for Electronic Health Record Data			
Description	Since EHR Data are been used by several of the SECURED use-case it is mandatory to be able to assess the anonymity of such datasets and discover potential privacy leakage				
Priority	Mandatory	**Type**	Functional	**CUR**	10 - 20

REQ-DATA-SEC-O-76	**Short Name:**	Provide broad-scope de-anonymization techniques			
Description	Based on the deliverable's SoTA, there exist several, very different techniques to assess the anonymity of a dataset. This is highly related to the dataset type, the data volume and the specific characteristics of this dataset. In SECURED, such techniques will be handled under a flexible and generic framework so that the provided solution can cover a broad range of de-anonymization attakcs and countermeasures				
Priority	Optional	**Type**	Functional	**CUR**	10 - 20

REQ-DATA-SEC-M-77	**Short Name:**	Provide report of Anonymization assessment			
Description	The Anonymization assessment service/tools must provide a report that detail the performed de-anonymization attack, their outcome in terms of discovered vulnerabilities as well as potential guidelines for the means to patch such vulnerabilities.				
Priority	Mandatory	**Type**	Functional	**CUR**	10 - 20

7.4.16 Technical Requirements on the Synthetic Data Generation Engine

REQ-SHW-PERF-M-78	**Short Name:**	Access to HPC hardware for efficient synthesis (several CPUs, GPUs...)			
Description	The proposed methodologies are usually quite compute intensive, therefore they need to have good enough hardware to compute.				
Priority	Mandatory	**Type**	Non-Functional	**CUR**	10 - 20

REQ-DATA-DATA-M-79	**Short Name:**	Generate data for different data types and modalities			
Description	The data generator needs to generate data for each type of data type and modality agreed.				
Priority	Mandatory	**Type**	Functional	**CUR**	10 - 20

REQ-DATA-DATA-D-80	**Short Name:**	Data novelty evaluation			
Description	The generated data must not be equal or closely related to the training data. This is needed to avoid reidentification of the original samples and data leaks.				
Priority	Desirable	**Type**	Functional	**CUR**	10 - 20

7.4.17 Technical Requirements on Anonymization Decision Support

REQ-DATA-PRIV-M-81	**Short Name:**	Privacy risk and data utility trade-off mechanisms for different health data types.)			
Description	The core functionality of the Anonymization Decision Support component is to be able to assist users in determining the optimal techniques and tools to anonymize their datasets. Thus, Anonymization Decision Support should provide for a given dataset the privacy risk and that data utility trade-off that the choice of a specific anonymization technique will have as a result to the dataset.				
Priority	Mandatory	**Type**	Functional	**CUR**	28 - 38

REQ-DATA-PRIV-D-82	**Short Name:**	User friendly anonymisation decision support and anonymization tools)			
Description	The complexity of the Anonymization Decision Support, the Data Anonymization Toolset and the Anonymization Service dictates the need for a user-friendly and easily-applicable process of using the tools and the service. This requirement is also applicable to the requirements on Data Anonymization Toolset and the Anonymization Service.				
Priority	Desirable	**Type**	Non-Functional	**CUR**	10 - 20

7.4.18 Technical Requirements on the Secure Multi-Party Computation (SMPC) Engine and Secure Multi-Party Computation (SMPC) Transformation

REQ-DPROC-SEC-M-83	**Short Name:**	Seamless integration of SotA open-source SMPC/HE libraries.)			
Description	There exist several open source SMPC/HE libraries librearies that could be used for the use-cases and objective of SECURED. Incorporation of libraries that best fit the SECURED objectives incl integration with ML/DL tasks needs to be made.				
Priority	Mandatory	**Type**	Non-Functional	**CUR**	10 - 20

REQ-DPROC-SEC-O-84	**Short Name:**	Cost Estimator for MPC/HE protocols.)			
Description	Since each SMPC/HE library has different performance cost, given that the SMPC Engine is closely related to the SMPC transformer component, a mechanism is need to assist the user on choosing the proper library to be used based on the cost that such library will have. This requirement is linked to both the SMPC Enine and SMPC Transformation				
Priority	Optional	**Type**	Non-Functional	**CUR**	10 - 20

REQ-DPROC-PERF-D-85	**Short Name:**	Circuit optimizer for hardware acceleration of HE.)			
Description	Tools exist to optimize circuits for SMPC/HE, but these will not take into account if we have hardware components to do e.g. HE operations that are slow in hardware on commodity CPUs. Hardware Optimizations need to be made in order to improve the performance of an SMPC/HE circuit.				
Priority	Desirable	**Type**	Non-Functional	**CUR**	10 - 20

REQ-DPROC-PERF-D-86	Short Name:	SMPC or HE solutions very fast response time)			
Description	Since use-case 1 has requirements for near-real time the SMPC Engine should be able to provide fast SMPC/HE response when processing encrypted or SMPC transformed data.				
Priority	Desirable	**Type**	Non-Functional	**CUR**	10 - 20

REQ-DPROC-SEC-M-87	Short Name:	Customized, adaptable SMPC Transformation process			
Description	The Transformation process must provide the user with the means to use in a simple and realistic manner the SMPC Engine libraries to generate a SMPC/HE enabled Model. The process for such operation must be made customised/adaptable so that it can allow, as much as possible, the reuse of some library component, mechanisms and techniques.				
Priority	Mandatory	**Type**	Functional	**CUR**	10 - 20

7.4.19 Technical Requirements on the Bias Assessment Service and Tools

REQ-DATA-REL-M-88	Short Name:	Provide accurate bias score for a given dataset.)			
Description	After analyzing a given dataset for biasing the Bias Assessment tool must be able to provide an indicator of the bias level such a dataset has. Thos bias score should be must be realistic and accurate to provide conclusive decisions on how biased a dataset is.				
Priority	Mandatory	**Type**	Functional	**CUR**	10 - 20

REQ-DATA-REL-M-89	Short Name:	Detection of Bias in Timeseries Health Data.)			
Description	Since Health Timeseries Data are been used by several of the SECURED use-case it is mandatory to be able to detect bias on such datasets.				
Priority	Madatory	**Type**	Functional	**CUR**	10 - 20

REQ-DATA-REL-M-90	Short Name:	Detection of Bias in Image Health Data)			
Description	Since various types of Health Image Data are been used by several of the SECURED use-case it is mandatory to be able to detect bias on such datasets.				
Priority	Mandatory	**Type**	Functional	**CUR**	10 - 20

REQ-DATA-REL-M-91	Short Name:	Detection of Bias in Electronic Health Record Data.)			
Description	Since EHR Data are been used by several of the SECURED use-case it is mandatory to be able to detect bias on such datasets.				
Priority	Mandatory	**Type**	Functional	**CUR**	10 - 20

REQ-DATA-REL-M-92	Short Name:	Detection of Bias in Anonymized Datasets.

Description	Given that the bias assessment is part of the Data Transformation Engine and operates in conjunction with the Anonymization Service and toolset, the bias assessment must be able to discover bias in anonymized datasets.				
Priority	Mandatory	**Type**	Functional	**CUR**	10 - 20

REQ-DATA-PRIV-M-93	**Short Name:**	Provide analytic bias assessment reports.
Description	The Bias Assessment service/tools must provide a report that details the performed bias assessment approaches, their outcome in terms of discovered bias as well as potential guidelines for unbiasing the dataset.	
Priority	Mandatory	**Type** Functional **CUR** 10 - 20

7.4.20 Technical Requirements on the Unbiasing Service and Tools

REQ-DATA-PRIV-M-94	**Short Name:**	Unbiasing of Timeseries Health Data.)
Description	Since Health Timeseries Data are been used by several of the SECURED use-case it is mandatory to be able to unbias such datasets.	
Priority	Mandatory	**Type** Functional **CUR** 10 - 20

REQ-DATA-PRIV-M-95	**Short Name:**	Unbiasing of Image Health Data.)
Description	Since various types of Health Image Data are been used by several of the SECURED use-case it is mandatory to be able to unbias such datasets.	
Priority	Madatory	**Type** Functional **CUR** 10 - 20

REQ-DATA-PRIV-M-96	**Short Name:**	Unbiasing of Electronic Health Record Data
Description	Since EHR Data are been used by several of the SECURED use-case it is mandatory to be able to unbias such datasets	
Priority	Mandatory	**Type** Functional **CUR** 10 - 20

REQ-DATA-PRIV-M-97	**Short Name:**	Unbiasing of Anonymized Datasets)
Description	Given that the unbiasing service/toolset is part of the Data Transformation Engine and operates in conjunction with the Anonymization Service and toolset, unbiasing must be performed on anonymized datasets.	
Priority	Mandatory	**Type** Functional **CUR** 10 - 20

REQ-DATA-PRIV-M-98	**Short Name:**	Provide report/guarantee of the Unbiasing process.
Description	The Unbiasing service/tools should be able to log the history of unbiasing rounds that took place as well as the bias that have been discovered and how it was removed. This log/report should be part of a privacy guarantee that must be associated with the outcome of the the Data Transformation service.	
Priority	Mandatory	**Type** Functional **CUR** 10 - 20

8 Conclusions

In this Deliverable we have documented the current State-of-the-Art (SoTA) on the various topics that SECURED project deals with i.e. Secure Multi-Party Computation (SMPC) solutions with a focus on Homomorphic Encryption (HE) and how SMPC/HE libraries can be scaled up using algorithmic and hardware acceleration, anonymization and de-anonymization schemes of health data, bias and unbiasing as well as synthetic data generation of health data. All of these topics have been viewed from the perspective of Machine Learning (ML)/Deep Learning (DL)/Federated Learning (FL) solutions that are related to the SECURED use-cases. For this reason, we also document the SoTA of health data related to ML/DL/FL research. We have extracted preliminary SoTA gaps that constitute useful research points during the SECURED project's lifetime and also provided a critical view of the existing tools/techniques that will help the consortium focus on research that is compatible to the SECURED solution while also yielding research outcomes that are beyond the SoTA. Apart from this reporting, in the deliverable we also documented the interactions with the use-case providers and their outcomes from M1-M6 to extract using the user journey/process mapping methodology, preliminary user-requirements, and a realistic/practical preliminary SECURED architecture. We also described the components of this architecture and link it to the four use-cases of the SECURED project while in parallel making it possible to extend it so as to cover additional use-cases after the completion of the SECURED open call for evaluators. Eventually, we analyzed all the above outcomes (i.e SoTA Gaps, preliminary user-requirements, SECURED architecture components, existing integration technologies) and provided a series of Technical Requirements for each SECURED architecture component that will guide the realization of the whole SECURED solution. It can be concluded that there is a lot a space for new research results to be made out of the project and that the use-cases of the project need to adopt unique views of the SECURED architecture. However, the SECURED architecture including the SECURED Federation Infrastructure and the SECURED Innohub manages to fully support all four use-cases and is generic enough to support additional use-cases. Another interesting conclusion from the performed analysis (especially on the existing SoTA) is that within the consortium we will have to develop tailored to the use-cases Privacy-Enhancing Technologies (PETs) applications that are closely related to the available datasets. However, on the other hand, the core concept (SECURED Innohub), services, Software and Hardware libraries and tools to be used remain generic enough to address any application. Eventually, this principle dictates all provided Technical Requirements (i.e generic solution approach) and the preliminary SECURED architecture as a whole. Finally, it should be noted that D4.1 is a preliminary deliverable of the T4.1 activities so it is expected that some of the described concepts, requirements, and other items in the deliverable will possibly be updated before the end of the task (at M18) given that by that time the T5.1 activities on user requirements will also be concluded (given user requirements and technical requirements are logically linked). All updates as well as the final SECURED Architecture, the interconnection between components and the structure of the exchanged data between components will be provided in the final deliverable (i.e. Deliverable 4.2) of T4.1.

References

[1] A. L. Beam and I. S. Kohane, "Big data and machine learning in health care," *Jama*, vol. 319, no. 13, pp. 1317–1318, 2018.

[2] W. Zhu, C. Liu, W. Fan, and X. Xie, "Deeplung: Deep 3d dual path nets for automated pulmonary nodule detection and classification," in *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2018, pp. 673–681.

[3] P. Afshar, A. Mohammadi, and K. N. Plataniotis, "Brain tumor type classification via capsule networks," in *2018 25th IEEE International Conference on Image Processing (ICIP)*, 2018, pp. 3129–3133.

[4] A. Collins and Y. Yao, *Machine Learning Approaches: Data Integration for Disease Prediction and Prognosis*. Singapore: Springer Singapore, 2018, pp. 137–141. [Online]. Available: https://doi.org/10.1007/978-981-13-1071-3_10

[5] A. Esteva, A. Robicquet, B. Ramsundar, V. Kuleshov, M. DePristo, K. Chou, C. Cui, G. Corrado, S. Thrun, and J. Dean, "A guide to deep learning in healthcare," *Nature medicine*, vol. 25, no. 1, pp. 24–29, 2019.

[6] P. B. Jensen, L. J. Jensen, and S. Brunak, "Mining electronic health records: towards better research applications and clinical care," *Nature Reviews Genetics*, vol. 13, no. 6, pp. 395–405, Jun. 2012, number: 6 Publisher: Nature Publishing Group. [Online]. Available: https://www.nature.com/articles/nrg3208

[7] Z. Wang, A. D. Shah, A. R. Tate, S. Denaxas, J. Shawe-Taylor, and H. Hemingway, "Extracting Diagnoses and Investigation Results from Unstructured Text in Electronic Health Records by Semi-Supervised Machine Learning," *PLOS ONE*, vol. 7, no. 1, p. e30412, Jan. 2012, publisher: Public Library of Science. [Online]. Available: https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0030412

[8] B. Nestor, M. B. A. McDermott, W. Boag, G. Berner, T. Naumann, M. C. Hughes, A. Goldenberg, and M. Ghassemi, "Feature Robustness in Non-stationary Health Records: Caveats to Deployable Model Performance in Common Clinical Machine Learning Tasks," Aug. 2019, arXiv:1908.00690 [cs, stat]. [Online]. Available: http://arxiv.org/abs/1908.00690

[9] Y. Xue, T. Xu, L. Rodney Long, Z. Xue, S. Antani, G. R. Thoma, and X. Huang, "Multimodal Recurrent Model with Attention for Automated Radiology Report Generation," in *Medical Image Computing and Computer Assisted Intervention – MICCAI 2018*, ser. Lecture Notes in Computer Science, A. F. Frangi, J. A. Schnabel, C. Davatzikos, C. Alberola-López, and G. Fichtinger, Eds. Cham: Springer International Publishing, 2018, pp. 457–466.

[10] "Natural Language–based Machine Learning Models for the Annotation of Clinical Radiology Reports," vol. 287. [Online]. Available: https://pubs.rsna.org/doi/abs/10.1148/radiol.2018171093

[11] B. Jing, P. Xie, and E. Xing, "On the automatic generation of medical imaging reports," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Melbourne, Australia: Association for Computational Linguistics, Jul. 2018, pp. 2577–2586. [Online]. Available: https://aclanthology.org/P18-1240

[12] A. Rajkomar, J. Dean, and I. Kohane, "Machine learning in medicine," *New England Journal of Medicine*, vol. 380, no. 14, pp. 1347–1358, 2019.

[13] F. Attal, S. Mohammed, M. Dedabrishvili, F. Chamroukhi, L. Oukhellou, and Y. Amirat, "Physical Human Activity Recognition Using Wearable Sensors," *Sensors*, vol. 15, no. 12, pp. 31 314–31 338, Dec. 2015, number: 12 Publisher: Multidisciplinary Digital Publishing Institute. [Online]. Available: https://www.mdpi.com/1424-8220/15/12/29858

[14] M. Ghassemi, T. Naumann, P. Schulam, A. L. Beam, I. Y. Chen, and R. Ranganath, "A review of challenges and opportunities in machine learning for health," *AMIA Summits on Translational Science Proceedings*, vol. 2020, p. 191, 2020.

[15] P. F. Schulam and S. Saria, "What-if reasoning with counterfactual gaussian processes," *ArXiv*, vol. abs/1703.10651, 2017.

[16] R. C. Sato and G. T. K. Sato, "Probabilistic graphic models applied to identification of diseases," *einstein (São Paulo)*, vol. 13, pp. 330–333, Jun. 2015, publisher: Instituto Israelita de Ensino e Pesquisa Albert Einstein. [Online]. Available: https://www.scielo.br/j/eins/a/NQF79C4d5FRvr5fMgHSLQNQ/?lang=en

[17] C. Glymour, K. Zhang, and P. Spirtes, "Review of Causal Discovery Methods Based on Graphical Models," *Frontiers in Genetics*, vol. 10, 2019. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fgene.2019.00524

[18] A. Qayyum, J. Qadir, M. Bilal, and A. Al-Fuqaha, "Secure and Robust Machine Learning for Healthcare: A Survey," *IEEE Reviews in Biomedical Engineering*, vol. 14, pp. 156–180, 2021, conference Name: IEEE Reviews in Biomedical Engineering.

[19] C. S. Perone, P. Ballester, R. C. Barros, and J. Cohen-Adad, "Unsupervised domain adaptation for medical imaging segmentation with self-ensembling," *NeuroImage*, vol. 194, pp. 1–11, Jul. 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1053811919302034

[20] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, "Membership inference attacks against machine learning models," in *2017 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2017.

[21] M. Fredrikson, S. Jha, and T. Ristenpart, "Model inversion attacks that exploit confidence information and basic countermeasures," in *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, 2015.

[22] L. Zhu, Z. Liu, and S. Han, "Deep leakage from gradients," in *Advances in Neural Information Processing Systems*, 2019.

[23] H. Zhang, J. Gao, and L. Su, "Data Poisoning Attacks Against Outcome Interpretations of Predictive Models," in *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. Virtual Event Singapore: ACM, Aug. 2021, pp. 2165–2173. [Online]. Available: https://dl.acm.org/doi/10.1145/3447548.3467405

[24] A.-K. Dombrowski, M. Alber, C. Anders, M. Ackermann, K.-R. Müller, and P. Kessel, "Explanations can be manipulated and geometry is to blame," in *Advances in Neural Information Processing Systems*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, Eds., vol. 32. Curran Associates, Inc., 2019. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2019/file/bb836c01cdc9120a9c984c525e4b1a4a-Paper.pdf

[25] I. Shumailov, Y. Zhao, D. Bates, N. Papernot, R. D. Mullins, and R. Anderson, "Sponge examples: Energy-latency attacks on neural networks," in *IEEE European Symposium on Security and Privacy, EuroS&P 2021, Vienna, Austria, September 6-10, 2021*. IEEE, 2021, pp. 212–231. [Online]. Available: https://doi.org/10.1109/EuroSP51992.2021.00024

[26] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016, http://www.deeplearningbook.org.

[27] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, p. 1798–1828, aug 2013. [Online]. Available: https://doi.org/10.1109/TPAMI.2013.50

[28] B. Norgeot, B. S. Glicksberg, and A. J. Butte, "A call for deep-learning healthcare," *Nature medicine*, vol. 25, no. 1, pp. 14–15, 2019.

[29] D. Ravì, C. Wong, F. Deligianni, M. Berthelot, J. Andreu-Perez, B. Lo, and G.-Z. Yang, "Deep learning for health informatics," *IEEE journal of biomedical and health informatics*, vol. 21, no. 1, pp. 4–21, 2016.

[30] K. Suzuki, "Overview of deep learning in medical imaging," *Radiological physics and technology*, vol. 10, no. 3, pp. 257–273, 2017.

[31] J.-G. Lee, S. Jun, Y.-W. Cho, H. Lee, G. B. Kim, J. B. Seo, and N. Kim, "Deep learning in medical imaging: general overview," *Korean journal of radiology*, vol. 18, no. 4, pp. 570–584, 2017.

[32] M. I. Razzak, S. Naz, and A. Zaib, "Deep learning for medical image processing: Overview, challenges and the future," *Classification in BioApps: Automation of Decision Making*, pp. 323–350, 2018.

[33] B. Shickel, P. J. Tighe, A. Bihorac, and P. Rashidi, "Deep EHR: A Survey of Recent Advances in Deep Learning Techniques for Electronic Health Record (EHR) Analysis," *IEEE journal of biomedical and health informatics*, vol. 22, no. 5, pp. 1589–1604, Sep. 2018.

[34] A. Rajkomar, E. Oren, K. Chen, A. M. Dai, N. Hajaj, M. Hardt, P. J. Liu, X. Liu, J. Marcus, M. Sun, P. Sundberg, H. Yee, K. Zhang, Y. Zhang, G. Flores, G. E. Duggan, J. Irvine, Q. Le, K. Litsch, A. Mossin, J. Tansuwan, D. Wang, J. Wexler, J. Wilson, D. Ludwig, S. L. Volchenboum, K. Chou, M. Pearson, S. Madabushi, N. H. Shah, A. J. Butte, M. D. Howell, C. Cui, G. S. Corrado, and J. Dean, "Scalable and accurate deep learning with electronic health records," *npj Digital Medicine*, vol. 1, no. 1, pp. 1–10, May 2018, number: 1 Publisher: Nature Publishing Group. [Online]. Available: https://www.nature.com/articles/s41746-018-0029-1

[35] Y. Kassahun, B. Yu, A. T. Tibebu, D. Stoyanov, S. Giannarou, J. H. Metzen, and E. Vander Poorten, "Surgical robotics beyond enhanced dexterity instrumentation: a survey of machine learning techniques and their role in intelligent and autonomous surgical actions," *International Journal of Computer Assisted Radiology and Surgery*, vol. 11, no. 4, pp. 553–568, Apr. 2016.

[36] H. Mayer, F. Gomez, D. Wierstra, I. Nagy, A. Knoll, and J. Schmidhuber, "A system for robotic heart surgery that learns to tie knots using recurrent neural networks," in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2006, pp. 543–548.

[37] T. Zhang, Z. McCarthy, O. Jow, D. Lee, X. Chen, K. Goldberg, and P. Abbeel, "Deep imitation learning for complex manipulation tasks from virtual reality teleoperation," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 5628–5635.

[38] L. R. Medsker and L. Jain, "Recurrent neural networks," *Design and Applications*, vol. 5, no. 64-67, p. 2, 2001.

[39] L. Cheng and M. Tavakoli, "Neural network-based physiological organ motion prediction and robot impedance control for teleoperated beating-heart surgery," *Biomedical Signal Processing and Control*, vol. 66, p. 102423, 2021.

[40] Y. Qin, H. Su, and X. Wang, "From one hand to multiple hands: Imitation learning for dexterous manipulation from single-camera teleoperation," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 10 873–10 881, 2022.

[41] V. Jindal, "Integrating mobile and cloud for ppg signal selection to monitor heart rate during intensive physical exercise," in *2016 IEEE/ACM International Conference on Mobile Software Engineering and Systems (MOBILESoft)*, 2016, pp. 36–37.

[42] D. Tran and D. M. Blei, "Implicit Causal Models for Genome-wide Association Studies," Oct. 2017, arXiv:1710.10742 [cs, q-bio, stat]. [Online]. Available: http://arxiv.org/abs/1710.10742

[43] M. K. K. Leung, A. Delong, B. Alipanahi, and B. J. Frey, "Machine learning in genomic medicine: A review of computational problems and data sets," *Proceedings of the IEEE*, vol. 104, no. 1, pp. 176–197, 2016.

[44] J. T. Dudley, J. Listgarten, O. Stegle, S. E. Brenner, and L. Parts, "Personalized medicine: from genotypes, molecular phenotypes and the quantified self, towards improved medicine," *Pacific Symposium on Biocomputing. Pacific Symposium on Biocomputing*, pp. 342–346, 2015.

[45] C. Zhang, Y. Xie, H. Bai, B. Yu, W. Li, and Y. Gao, "A survey on federated learning," *Knowledge-Based Systems*, vol. 216, p. 106775, 2021.

[46] P. M. Mammen, "Federated learning: Opportunities and challenges," *arXiv preprint arXiv:2101.05428*, 2021.

[47] L. Fu, H. Zhang, G. Gao, H. Wang, M. Zhang, and X. Liu, "Client selection in federated learning: Principles, challenges, and opportunities," *arXiv preprint arXiv:2211.01549*, 2022.

[48] T. Nishio and R. Yonetani, "Client selection for federated learning with heterogeneous resources in mobile edge," in *ICC 2019-2019 IEEE international conference on communications (ICC)*. IEEE, 2019, pp. 1–7.

[49] Y. J. Cho, J. Wang, and G. Joshi, "Client selection in federated learning: Convergence analysis and power-of-choice selection strategies," *arXiv preprint arXiv:2010.01243*, 2020.

[50] F. Lai, X. Zhu, H. V. Madhyastha, and M. Chowdhury, "Oort: Efficient federated learning via guided participant selection." in *OSDI*, 2021, pp. 19–35.

[51] J. Xu and H. Wang, "Client selection and bandwidth allocation in wireless federated learning networks: A long-term perspective," *IEEE Transactions on Wireless Communications*, vol. 20, no. 2, pp. 1188–1200, 2020.

[52] H. T. Nguyen, V. Sehwag, S. Hosseinalipour, C. G. Brinton, M. Chiang, and H. V. Poor, "Fast-convergent federated learning," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 1, pp. 201–218, 2020.

[53] S. Sun, Z. Cao, H. Zhu, and J. Zhao, "A survey of optimization methods from a machine learning perspective," *IEEE transactions on cybernetics*, vol. 50, no. 8, pp. 3668–3681, 2019.

[54] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Artificial intelligence and statistics*. PMLR, 2017, pp. 1273–1282.

[55] M. P. Sah and A. Singh, "Aggregation techniques in federated learning: Comprehensive survey, challenges and opportunities," in *2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE)*. IEEE, 2022, pp. 1962–1967.

[56] S. Ek, F. Portet, P. Lalanda, and G. Vega, "Evaluation of federated learning aggregation algorithms: application to human activity recognition," in *Adjunct proceedings of the 2020 ACM international joint conference on pervasive and ubiquitous computing and proceedings of the 2020 ACM international symposium on wearable computers*, 2020, pp. 638–643.

[57] Z. Liu, J. Guo, W. Yang, J. Fan, K.-Y. Lam, and J. Zhao, "Privacy-preserving aggregation in federated learning: A survey," *IEEE Transactions on Big Data*, 2022.

[58] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," *Proceedings of Machine learning and systems*, vol. 2, pp. 429–450, 2020.

[59] V. Smith, C.-K. Chiang, M. Sanjabi, and A. S. Talwalkar, "Federated multi-task learning," *Advances in neural information processing systems*, vol. 30, 2017.

[60] M. G. Arivazhagan, V. Aggarwal, A. K. Singh, and S. Choudhary, "Federated learning with personalization layers," *arXiv preprint arXiv:1912.00818*, 2019.

[61] H. Wang, M. Yurochkin, Y. Sun, D. Papailiopoulos, and Y. Khazaeni, "Federated learning with matched averaging," *arXiv preprint arXiv:2002.06440*, 2020.

[62] N. Guha, A. Talwalkar, and V. Smith, "One-shot federated learning," *arXiv preprint arXiv:1902.11175*, 2019.

[63] Y. Zhan, J. Zhang, Z. Hong, L. Wu, P. Li, and S. Guo, "A survey of incentive mechanism design for federated learning," *IEEE Transactions on Emerging Topics in Computing*, vol. 10, no. 2, pp. 1035–1044, 2021.

[64] C. Düsing and P. Cimiano, "On the trade-off between benefit and contribution for clients in federated learning in healthcare," in *2022 21st IEEE International Conference on Machine Learning and Applications (ICMLA)*. IEEE, 2022, pp. 1672–1678.

[65] Y. Shi, H. Yu, and C. Leung, "Towards fairness-aware federated learning," *IEEE Transactions on Neural Networks and Learning Systems*, 2023.

[66] L. S. Shapley, "A value for n-person games," *Contributions to the Theory of Games*, 1953.

[67] B. Rozemberczki, L. Watson, P. Bayer, H.-T. Yang, O. Kiss, S. Nilsson, and R. Sarkar, "The shapley value in machine learning," *arXiv preprint arXiv:2202.05594*, 2022.

[68] J. Castro, D. Gómez, and J. Tejada, "Polynomial calculation of the shapley value based on sampling," *Computers & Operations Research*, 2009.

[69] A. Ghorbani and J. Zou, "Data shapley: Equitable valuation of data for machine learning," *arXiv preprint arXiv:1904.02868*, 2019.

[70] P. W. Koh and P. Liang, "Understanding black-box predictions via influence functions," *arXiv preprint arXiv:1703.04730*, 2017.

[71] B. Pejó, G. Biczók, and G. Ács, "Measuring contributions in privacy-preserving federated learning," *ERCIM NEWS*, p. 35, 2021.

[72] B. Pejó, A. Tótth, and G. Biczók, "Quality inference in federated learning with secure aggregation," *IEEE Transactions on Big Data*, 2023.

[73] Q. Li, Z. Wen, Z. Wu, S. Hu, N. Wang, Y. Li, X. Liu, and B. He, "A survey on federated learning systems: vision, hype and reality for data privacy and protection," *IEEE Transactions on Knowledge and Data Engineering*, 2021.

[74] L. Lyu, H. Yu, and Q. Yang, "Threats to federated learning: A survey," *arXiv preprint arXiv:2003.02133*, 2020.

[75] R. Gosselin, L. Vieu, F. Loukil, and A. Benoit, "Privacy and security in federated learning: A survey," *Applied Sciences*, vol. 12, no. 19, p. 9901, 2022.

[76] C. Fung, C. J. Yoon, and I. Beschastnikh, "The limitations of federated learning in sybil settings." in *RAID*, 2020, pp. 301–316.

[77] J. Shi, W. Wan, S. Hu, J. Lu, and L. Y. Zhang, "Challenges and approaches for mitigating byzantine attacks in federated learning," in *2022 IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*. IEEE, 2022, pp. 139–146.

[78] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," *arXiv preprint arXiv:1412.6572*, 2014.

[79] C. Fung, C. J. Yoon, and I. Beschastnikh, "Mitigating sybils in federated learning poisoning," *arXiv preprint arXiv:1808.04866*, 2018.

[80] E. Bagdasaryan, A. Veit, Y. Hua, D. Estrin, and V. Shmatikov, "How to backdoor federated learning," in *International Conference on Artificial Intelligence and Statistics*. PMLR, 2020, pp. 2938–2948.

[81] J. So, B. Güler, and A. S. Avestimehr, "Byzantine-resilient secure federated learning," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 7, pp. 2168–2181, 2020.

[82] N. Carlini, A. Athalye, N. Papernot, W. Brendel, J. Rauber, D. Tsipras, I. Goodfellow, A. Madry, and A. Kurakin, "On evaluating adversarial robustness," *arXiv preprint arXiv:1902.06705*, 2019.

[83] M. Lecuyer, V. Atlidakis, R. Geambasu, D. Hsu, and S. Jana, "Certified robustness to adversarial examples with differential privacy," in *2019 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2019, pp. 656–672.

[84] V. Mothukuri, R. M. Parizi, S. Pouriyeh, Y. Huang, A. Dehghantanha, and G. Srivastava, "A survey on security and privacy of federated learning," *Future Generation Computer Systems*, vol. 115, pp. 619–640, 2021.

[85] M. Fredrikson, S. Jha, and T. Ristenpart, "Model inversion attacks that exploit confidence information and basic countermeasures," in *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security, Denver, CO, USA, October 12-16, 2015*, I. Ray, N. Li, and C. Kruegel, Eds. ACM, 2015, pp. 1322–1333. [Online]. Available: https://doi.org/10.1145/2810103.2813677

[86] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, "Membership inference attacks against machine learning models," in *2017 IEEE Symposium on Security and Privacy, SP 2017, San Jose, CA, USA, May 22-26, 2017*. IEEE Computer Society, 2017, pp. 3–18. [Online]. Available: https://doi.org/10.1109/SP.2017.41

[87] K. Ganju, Q. Wang, W. Yang, C. A. Gunter, and N. Borisov, "Property inference attacks on fully connected neural networks using permutation invariant representations," in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, 2018.

[88] F. Tramèr, F. Zhang, A. Juels, M. K. Reiter, and T. Ristenpart, "Stealing machine learning models via prediction apis," in *25th USENIX Security Symposium, USENIX Security 16, Austin, TX, USA, August 10-12, 2016*, T. Holz and S. Savage, Eds. USENIX Association, 2016, pp. 601–618. [Online]. Available: https://www.usenix.org/conference/usenixsecurity16/technical-sessions/presentation/tramer

[89] B. Pejó and D. Desfontaines, *Guide to Differential Privacy Modifications: A Taxonomy of Variants and Extensions*. Springer Nature, 2022.

[90] G. Ács and C. Castelluccia, "I have a dream!(differentially private smart metering)." in *Information hiding*, vol. 6958. Springer, 2011, pp. 118–132.

[91] N. Rieke, J. Hancox, W. Li, F. Milletari, H. R. Roth, S. Albarqouni, S. Bakas, M. N. Galtier, B. A. Landman, K. Maier-Hein *et al.*, "The future of digital health with federated learning," *NPJ digital medicine*, vol. 3, no. 1, p. 119, 2020.

[92] W. Li, F. Milletarì, D. Xu, N. Rieke, J. Hancox, W. Zhu, M. Baust, Y. Cheng, S. Ourselin, M. J. Cardoso, and A. Feng, "Privacy-preserving federated brain tumour segmentation," in *Machine Learning in Medical Imaging: 10th International Workshop, MLMI 2019, Held in Conjunction with MICCAI 2019, Shenzhen, China, October 13, 2019, Proceedings*. Berlin, Heidelberg: Springer-Verlag, 2019, p. 133–141. [Online]. Available: https://doi.org/10.1007/978-3-030-32692-0_16

[93] D. C. Nguyen, Q.-V. Pham, P. N. Pathirana, M. Ding, A. Seneviratne, Z. Lin, O. Dobre, and W.-J. Hwang, "Federated learning for smart healthcare: A survey," *ACM Computing Surveys (CSUR)*, vol. 55, no. 3, pp. 1–37, 2022.

[94] T. S. Brisimi, R. Chen, T. Mela, A. Olshevsky, I. C. Paschalidis, and W. Shi, "Federated learning of predictive models from federated electronic health records," *International Journal of Medical Informatics*, vol. 112, pp. 59–67, 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S138650561830008X

[95] D. Liu, T. A. Miller, R. Sayeed, and K. D. Mandl, "FADL: federated-autonomous deep learning for distributed electronic health record," *CoRR*, vol. abs/1811.11400, 2018. [Online]. Available: http://arxiv.org/abs/1811.11400

[96] A. G. Roy, S. Siddiqui, S. Pölsterl, N. Navab, and C. Wachinger, "Braintorrent: A peer-to-peer environment for decentralized federated learning," *CoRR*, vol. abs/1905.06731, 2019. [Online]. Available: http://arxiv.org/abs/1905.06731

[97] X. Li, Y. Gu, N. C. Dvornek, L. H. Staib, P. Ventola, and J. S. Duncan, "Multi-site fmri analysis using privacy-preserving federated learning and domain adaptation: ABIDE results," *Medical Image Anal.*, vol. 65, p. 101765, 2020. [Online]. Available: https://doi.org/10.1016/j.media.2020.101765

[98] C. He, M. Annavaram, and S. Avestimehr, "Fednas: Federated deep learning via neural architecture search," *CoRR*, vol. abs/2004.08546, 2020. [Online]. Available: https://arxiv.org/abs/2004.08546

[99] P. Vepakomma, O. Gupta, T. Swedish, and R. Raskar, "Split learning for health: Distributed deep learning without sharing raw patient data," *CoRR*, vol. abs/1812.00564, 2018. [Online]. Available: http://arxiv.org/abs/1812.00564

[100] M. Oldenhof, G. Ács, B. Pejó, A. Schuffenhauer, N. Holway, N. Sturm, A. Dieckmann, O. Fortmeier, E. Boniface, C. Mayer, A. Gohier, P. Schmidtke, R. Niwayama, D. Kopecky, L. H. Mervin, P. C. Rathi, L. Friedrich, A. Formanek, P. Antal, J. Rahaman, A. Zalewski, E. Oluoch, M. Stößel, M. Vanco, D. Endico, F. Gelus, T. de Boisfossé, A. Darbier, A. Nicollet, M. Blottière, M. Telenczuk, V. T. Nguyen, T. Martinez, C. Boillet, K. Moutet, A. Picosson, A. Gasser, I. Djafar, A. Arany, J. Simm, Y. Moreau, O. Engkvist, H. Ceulemans, C. Marini, and M. Galtier, "Industry-scale orchestrated federated learning for drug discovery," *CoRR*, vol. abs/2210.08871, 2022. [Online]. Available: https://doi.org/10.48550/arXiv.2210.08871

[101] M. A. Haeri and K. A. Zweig, "The crucial role of sensitive attributes in fair classification," in *2020 IEEE Symposium Series on Computational Intelligence (SSCI)*, 2020, pp. 2993–3002.

[102] N. Mehrabi, F. Morstatter, N. Saxena, K. Lerman, and A. Galstyan, "A survey on bias and fairness in machine learning," *ACM Comput. Surv.*, vol. 54, no. 6, jul 2021. [Online]. Available: https://doi.org/10.1145/3457607

[103] Y. Shi, H. Yu, and C. Leung, "Towards fairness-aware federated learning," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–17, 2023. [Online]. Available: https://doi.org/10.1109%2Ftnnls.2023.3263594

[104] J. Ogier du Terrail, A. Leopold, C. Joly, C. Béguier, and al., "Federated learning for predicting histological response to neoadjuvant chemotherapy in triple-negative breast cancer," *Nature Medicine*, vol. 29, pp. 135–146, 2023. [Online]. Available: https://doi.org/10.1038/s41591-022-02155-w

[105] A. Abay, E. Chuba, Y. Zhou, N. Baracaldo, and H. Ludwig, "Addressing unique fairness obstacles within federated learning," in *CEUR Workshop*, Feb. 2021, pp. 27–27. [Online]. Available: https://ceur-ws.org/Vol-2812/RDAI-2021_paper_9.pdf

[106] C. Saplicki and M. Bante, "Fairness explained: Definitions and metrics," Nov 2022. [Online]. Available: https://medium.com/ibm-data-ai/fairness-explained-definitions-and-metrics-9690f8e0a4ea

[107] C. Dwork, M. Hardt, T. Pitassi, O. Reingold, and R. Zemel, "Fairness through awareness," in *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*, ser. ITCS '12. New York, NY, USA: Association for Computing Machinery, 2012, p. 214–226. [Online]. Available: https://doi.org/10.1145/2090236.2090255

[108] J. Kleinberg, S. Mullainathan, and M. Raghavan, "Inherent Trade-Offs in the Fair Determination of Risk Scores," in *8th Innovations in Theoretical Computer Science Conference (ITCS 2017)*, ser. Leibniz International Proceedings in Informatics (LIPIcs), C. H. Papadimitriou, Ed., vol. 67. Dagstuhl, Germany: Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2017, pp. 43:1–43:23. [Online]. Available: http://drops.dagstuhl.de/opus/volltexte/2017/8156

[109] Broward County Clerk's Office, Broward County Sherrif's Office, Florida Department of Corrections, ProPublica, "Compas recidivism risk score data and analysis," 2016. [Online]. Available: https://www.propublica.org/datastore/dataset/compas-recidivism-risk-score-data-and-analysis

[110] A. Rudra, "Fairness and algorithms," 2023. [Online]. Available: http://www-student.cse.buffalo.edu/~atri/algo-and-society/support/notes/fairness/index.html

[111] A. Bell, L. Bynum, N. Drushchak, T. Herasymova, L. Rosenblatt, and J. Stoyanovich, "The possibility of fairness: Revisiting the impossibility theorem in practice," 2023.

[112] M. J. Kusner, J. R. Loftus, C. Russell, and R. Silva, "Counterfactual fairness," 2018.

[113] R. Islam, K. N. Keya, S. Pan, A. D. Sarwate, and J. R. Foulds, "Differential fairness: An intersectional framework for fair ai," *Entropy*, vol. 25, no. 4, 2023. [Online]. Available: https://www.mdpi.com/1099-4300/25/4/660

[114] R. K. E. Bellamy, K. Dey, M. Hind, S. C. Hoffman, S. Houde, K. Kannan, P. Lohia, J. Martino, S. Mehta, A. Mojsilovic, S. Nagar, K. N. Ramamurthy, J. T. Richards, D. Saha, P. Sattigeri, M. Singh, K. R. Varshney, and Y. Zhang, "AI fairness 360: An extensible toolkit for detecting, understanding, and mitigating unwanted algorithmic bias," *CoRR*, vol. abs/1810.01943, 2018. [Online]. Available: http://arxiv.org/abs/1810.01943

[115] F. Kamiran and T. Calders, "Data preprocessing techniques for classification without discrimination," *Knowledge and Information Systems*, vol. 33, pp. 1 – 33, 2011.

[116] M. Sugiyama, S. Nakajima, and H. Kashima, "Direct importance estimation with model selection and its application to covariate shift adaptation," 01 2007.

[117] Y. Yu and C. Szepesvari, "Analysis of kernel mean matching under covariate shift," 2012.

[118] M. Feldman, S. Friedler, J. Moeller, C. Scheidegger, and S. Venkatasubramanian, "Certifying and removing disparate impact," 2015.

[119] R. Zemel, Y. Wu, K. Swersky, T. Pitassi, and C. Dwork, "Learning fair representations," in *Proceedings of the 30th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, S. Dasgupta and D. McAllester, Eds., vol. 28, no. 3. Atlanta, Georgia, USA: PMLR, 17–19 Jun 2013, pp. 325–333. [Online]. Available: https://proceedings.mlr.press/v28/zemel13.html

[120] R. Feng, Y. Yang, Y. Lyu, C. Tan, Y. Sun, and C. Wang, "Learning fair representations via an adversarial framework," 2019.

[121] D. Kim, K. Kim, I. Kong, I. Ohn, and Y. Kim, "Learning fair representation with a parametric integral probability metric," 2023.

[122] D. Xu, S. Yuan, L. Zhang, and X. Wu, "Fairgan: Fairness-aware generative adversarial networks," 2018.

[123] F. P. Calmon, D. Wei, K. N. Ramamurthy, and K. R. Varshney, "Optimized data pre-processing for discrimination prevention," 2017.

[124] G. Louppe, M. Kagan, and K. Cranmer, "Learning to pivot with adversarial networks," 2017.

[125] B. H. Zhang, B. Lemoine, and M. Mitchell, "Mitigating unwanted biases with adversarial learning," 2018.

[126] X. Wang and H. Huang, "Approaching machine learning fairness through adversarial network," 2019.

[127] T. Adel, I. Valera, Z. Ghahramani, and A. Weller, "One-network adversarial fairness," in *AAAI Conference on Artificial Intelligence*, 2019.

[128] X. Gao, J. Zhai, S. Ma, C. Shen, Y. Chen, and Q. Wang, "Fairneuron: Improving deep neural network fairness with adversary games on selective neurons," 2022.

[129] E. Raff, J. Sylvester, and S. Mills, "Fair forests: Regularized tree induction to minimize model bias," 2017.

[130] W. Zhang, A. Bifet, X. Zhang, J. C. Weiss, and W. Nejdl, "Farf: A fair and adaptive random forests classifier," 2021.

[131] L. Breiman, "Random forests," *Machine Learning*, vol. 45, pp. 5–32, 2001.

[132] F. Kamiran, A. Karim, and X. Zhang, "Decision theory for discrimination-aware classification," in *2012 IEEE 12th International Conference on Data Mining*, 2012, pp. 924–929.

[133] M. Hardt, E. Price, and N. Srebro, "Equality of opportunity in supervised learning," 2016.

[134] I. Alabdulmohsin and M. Lucic, "A near-optimal algorithm for debiasing trained machine learning models," 2022.

[135] H. Chang and R. Shokri, "Bias propagation in federated learning," in *The Eleventh International Conference on Learning Representations*, 2023. [Online]. Available: https://openreview.net/forum?id=V7CYzdruWdm

[136] A. Abay, Y. Zhou, N. Baracaldo, S. Rajamoni, E. Chuba, and H. Ludwig, "Mitigating bias in federated learning," 2020.

[137] J.-F. Rajotte, S. Mukherjee, C. Robinson, A. Ortiz, C. West, J. L. Ferres, and R. T. Ng, "Reducing bias and increasing utility by federated generative modeling of medical images using a centralized adversary," 2021.

[138] S. Mukherjee, Y. Xu, A. Trivedi, and J. L. Ferres, "privgan: Protecting gans from membership inference attacks at low cost," 2020.

[139] W. Du, D. Xu, X. Wu, and H. Tong, "Fairness-aware agnostic federated learning," in *Proceedings of the 2021 SIAM International Conference on Data Mining (SDM)*. SIAM, 2021, pp. 181–189.

[140] L. Chu, L. Wang, Y. Dong, J. Pei, Z. Zhou, and Y. Zhang, "Fedfair: Training fair models in cross-silo federated learning," *CoRR*, vol. abs/2109.05662, 2021. [Online]. Available: https://arxiv.org/abs/2109.05662

[141] Y. Djebrouni, "Towards bias mitigation in federated learning," in *16th EuroSys Doctoral Workshop*, Rennes, France, Apr. 2022. [Online]. Available: https://hal.science/hal-03639179

[142] M. Duan, D. Liu, X. Chen, Y. Tan, J. Ren, L. Qiao, and L. Liang, "Astraea: Self-balancing federated learning for improving classification accuracy of mobile deep learning applications," in *2019 IEEE 37th International Conference on Computer Design (ICCD)*. IEEE, nov 2019. [Online]. Available: https://doi.org/10.1109%2Ficcd46524.2019.00038

[143] L. Ferraguig, Y. Djebrouni, S. Bouchenak, and V. Marangozova, "Survey of bias mitigation in federated learning," in *ComPAS'2021*, Jul. 2021. [Online]. Available: https://hal.science/hal-03343288

[144] C. Dwork, F. McSherry, K. Nissim, and A. D. Smith, "Calibrating noise to sensitivity in private data analysis," in *Theory of Cryptography, Third Theory of Cryptography Conference, TCC 2006, New York, NY, USA, March 4-7, 2006, Proceedings*, ser. Lecture Notes in Computer Science, S. Halevi and T. Rabin, Eds., vol. 3876. Springer, 2006, pp. 265–284. [Online]. Available: https://doi.org/10.1007/11681878_14

[145] M. A. Pathak, S. Rane, and B. Raj, "Multiparty differential privacy via aggregation of locally trained classifiers," in *Advances in Neural Information Processing Systems 23: 24th Annual Conference on Neural Information Processing Systems 2010. Proceedings of a meeting held 6-9 December 2010, Vancouver, British Columbia, Canada*, J. D. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R. S. Zemel, and A. Culotta, Eds. Curran Associates, Inc., 2010, pp. 1876–1884. [Online]. Available: https://proceedings.neurips.cc/paper/2010/hash/0d0fd7c6e093f7b804fa0150b875b868-Abstract.html

[146] R. Shokri and V. Shmatikov, "Privacy-preserving deep learning," in *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security, Denver, CO, USA, October 12-16, 2015*, I. Ray, N. Li, and C. Kruegel, Eds. ACM, 2015, pp. 1310–1321. [Online]. Available: https://doi.org/10.1145/2810103.2813687

[147] Z. Á. Mann, C. Weinert, D. Chabal, and J. W. Bos, "Towards practical secure neural network inference: The journey so far and the road ahead," *IACR Cryptol. ePrint Arch.*, p. 1483, 2022. [Online]. Available: https://eprint.iacr.org/2022/1483

[148] D. Rathee, M. Rathee, N. Kumar, N. Chandran, D. Gupta, A. Rastogi, and R. Sharma, "Cryptflow2: Practical 2-party secure inference," in *CCS '20: 2020 ACM SIGSAC Conference on Computer and Communications Security, Virtual Event, USA, November 9-13, 2020*, J. Ligatti, X. Ou, J. Katz, and G. Vigna, Eds. ACM, 2020, pp. 325–342. [Online]. Available: https://doi.org/10.1145/3372297.3417274

[149] F. Boemer, R. Cammarota, D. Demmler, T. Schneider, and H. Yalame, "MP2ML: a mixed-protocol machine learning framework for private inference," in *ARES 2020: The 15th International Conference on Availability, Reliability and Security, Virtual Event, Ireland, August 25-28, 2020*, M. Volkamer and C. Wressnegger, Eds. ACM, 2020, pp. 14:1–14:10. [Online]. Available: https://doi.org/10.1145/3407023.3407045

[150] M. S. Riazi, C. Weinert, O. Tkachenko, E. M. Songhori, T. Schneider, and F. Koushanfar, "Chameleon: A hybrid secure computation framework for machine learning applications," in *Proceedings of the 2018 on Asia Conference on Computer and Communications Security, AsiaCCS 2018, Incheon, Republic of Korea, June 04-08, 2018*, J. Kim, G. Ahn, S. Kim, Y. Kim, J. López, and T. Kim, Eds. ACM, 2018, pp. 707–721. [Online]. Available: https://doi.org/10.1145/3196494.3196522

[151] R. Lehmkuhl, P. Mishra, A. Srinivasan, and R. A. Popa, "Muse: Secure inference resilient to malicious clients," in *30th USENIX Security Symposium, USENIX Security 2021, August 11-13, 2021*, M. Bailey and R. Greenstadt, Eds. USENIX Association, 2021, pp. 2201–2218. [Online]. Available: https://www.usenix.org/conference/usenixsecurity21/presentation/lehmkuhl

[152] J. Ye, A. Maddi, S. K. Murakonda, V. Bindschaedler, and R. Shokri, "Enhanced membership inference attacks against machine learning models," in *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security, CCS 2022, Los Angeles, CA, USA, November 7-11, 2022*, H. Yin, A. Stavrou, C. Cremers, and E. Shi, Eds. ACM, 2022, pp. 3093–3106. [Online]. Available: https://doi.org/10.1145/3548606.3560675

[153] P. Mohassel and Y. Zhang, "Secureml: A system for scalable privacy-preserving machine learning," in *2017 IEEE Symposium on Security and Privacy, SP 2017, San Jose, CA, USA, May 22-26, 2017*. IEEE Computer Society, 2017, pp. 19–38. [Online]. Available: https://doi.org/10.1109/SP.2017.12

[154] J. Liu, M. Juuti, Y. Lu, and N. Asokan, "Oblivious neural network predictions via minionn transformations," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, CCS 2017, Dallas, TX, USA, October 30 - November 03, 2017*, B. Thuraisingham, D. Evans, T. Malkin, and D. Xu, Eds. ACM, 2017, pp. 619–631. [Online]. Available: https://doi.org/10.1145/3133956.3134056

[155] P. Mishra, R. Lehmkuhl, A. Srinivasan, W. Zheng, and R. A. Popa, "Delphi: A cryptographic inference service for neural networks," in *29th USENIX Security Symposium, USENIX Security 2020, August 12-14, 2020*, S. Capkun and F. Roesner, Eds. USENIX Association, 2020, pp. 2505–2522. [Online]. Available: https://www.usenix.org/conference/usenixsecurity20/presentation/mishra

[156] S. Wagh, S. Tople, F. Benhamouda, E. Kushilevitz, P. Mittal, and T. Rabin, "Falcon: Honest-majority maliciously secure framework for private deep learning," *Proc. Priv. Enhancing Technol.*, vol. 2021, no. 1, pp. 188–208, 2021. [Online]. Available: https://doi.org/10.2478/popets-2021-0011

[157] D. Demmler, T. Schneider, and M. Zohner, "ABY - A framework for efficient mixed-protocol secure two-party computation," in *22nd Annual Network and Distributed System Security Symposium, NDSS 2015, San Diego, California, USA, February 8-11, 2015*. The Internet Society, 2015. [Online]. Available: https://www.ndss-symposium.org/ndss2015/aby---framework-efficient-mixed-protocol-secure-two-party-computation

[158] P. Mohassel and P. Rindal, "Aby^3: A mixed protocol framework for machine learning," in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security, CCS 2018, Toronto, ON, Canada, October 15-19, 2018*, D. Lie, M. Mannan, M. Backes, and X. Wang, Eds. ACM, 2018, pp. 35–52. [Online]. Available: https://doi.org/10.1145/3243734.3243760

[159] C. Boura, N. Gama, M. Georgieva, and D. Jetchev, "CHIMERA: combining ring-LWE-based fully homomorphic encryption schemes," *J. Math. Cryptol.*, vol. 14, no. 1, pp. 316–338, 2020. [Online]. Available: https://doi.org/10.1515/jmc-2019-0026

[160] C. Juvekar, V. Vaikuntanathan, and A. P. Chandrakasan, "GAZELLE: A low latency framework for secure neural network inference," in *27th USENIX Security Symposium, USENIX Security 2018, Baltimore, MD, USA, August 15-17, 2018*, W. Enck and A. P. Felt, Eds. USENIX Association, 2018, pp. 1651–1669. [Online]. Available: https://www.usenix.org/conference/usenixsecurity18/presentation/juvekar

[161] N. Kumar, M. Rathee, N. Chandran, D. Gupta, A. Rastogi, and R. Sharma, "Cryptflow: Secure tensorflow inference," in *2020 IEEE Symposium on Security and Privacy, SP 2020, San Francisco, CA, USA, May 18-21, 2020*. IEEE, 2020, pp. 336–353. [Online]. Available: https://doi.org/10.1109/SP40000.2020.00092

[162] R. Gilad-Bachrach, N. Dowlin, K. Laine, K. E. Lauter, M. Naehrig, and J. Wernsing, "Cryptonets: Applying neural networks to encrypted data with high throughput and accuracy," in *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016*, ser. JMLR Workshop and Conference Proceedings, M. Balcan and K. Q. Weinberger, Eds., vol. 48. JMLR.org, 2016, pp. 201–210. [Online]. Available: http://proceedings.mlr.press/v48/gilad-bachrach16.html

[163] B. D. Rouhani, M. S. Riazi, and F. Koushanfar, "Deepsecure: scalable provably-secure deep learning," in *Proceedings of the 55th Annual Design Automation Conference, DAC 2018, San Francisco, CA, USA, June 24-29, 2018*. ACM, 2018, pp. 2:1–2:6. [Online]. Available: https://doi.org/10.1145/3195970.3196023

[164] M. S. Riazi, M. Samragh, H. Chen, K. Laine, K. E. Lauter, and F. Koushanfar, "XONN: xnor-based oblivious deep neural network inference," in *28th USENIX Security Symposium, USENIX Security 2019, Santa Clara, CA, USA, August 14-16, 2019*, N. Heninger and P. Traynor, Eds. USENIX Association, 2019, pp. 1501–1518. [Online]. Available: https://www.usenix.org/conference/usenixsecurity19/presentation/riazi

[165] M. O. Rabin, "How to exchange secrets with oblivious transfer," *Technical Report TR-91, Harvard Aiken Computer Laboratory*, 1981. [Online]. Available: http://eprint.iacr.org/2005/187

[166] Y. Ishai, J. Kilian, K. Nissim, and E. Petrank, "Extending oblivious transfers efficiently," in *Advances in Cryptology - CRYPTO 2003, 23rd Annual International Cryptology Conference, Santa Barbara, California, USA, August 17-21, 2003, Proceedings*, ser. Lecture Notes in Computer Science, D. Boneh, Ed., vol. 2729. Springer, 2003, pp. 145–161. [Online]. Available: https://doi.org/10.1007/978-3-540-45146-4_9

[167] C. Crépeau, "Equivalence between two flavours of oblivious transfers," in *Advances in Cryptology - CRYPTO '87, A Conference on the Theory and Applications of Cryptographic Techniques, Santa Barbara, California, USA, August 16-20, 1987, Proceedings*, ser. Lecture Notes in Computer Science, C. Pomerance, Ed., vol. 293. Springer, 1987, pp. 350–354. [Online]. Available: https://doi.org/10.1007/3-540-48184-2_30

[168] J. Hippisley-Cox, C. Coupland, and P. Brindle, "Development and validation of QRISK3 risk prediction algorithms to estimate future risk of cardiovascular disease: prospective cohort study," *British Medical Journal*, vol. 357, 2017.

[169] A. C. Yao, "Protocols for secure computations (extended abstract)," in *23rd Annual Symposium on Foundations of Computer Science, Chicago, Illinois, USA, 3-5 November 1982*. IEEE Computer Society, 1982, pp. 160–164. [Online]. Available: https://doi.org/10.1109/SFCS.1982.38

[170] D. Beaver, S. Micali, and P. Rogaway, "The round complexity of secure protocols (extended abstract)," in *Proceedings of the 22nd Annual ACM Symposium on Theory of Computing, May 13-17, 1990, Baltimore, Maryland, USA*, H. Ortiz, Ed. ACM, 1990, pp. 503–513. [Online]. Available: https://doi.org/10.1145/100216.100287

[171] D. Malkhi, N. Nisan, B. Pinkas, and Y. Sella, "Fairplay - secure two-party computation system," in *Proceedings of the 13th USENIX Security Symposium, August 9-13, 2004, San Diego, CA, USA*, M. Blaze, Ed. USENIX, 2004, pp. 287–302. [Online]. Available: http://www.usenix.org/publications/library/proceedings/sec04/tech/malkhi.html

[172] M. Bellare, V. T. Hoang, S. Keelveedhi, and P. Rogaway, "Efficient garbling from a fixed-key blockcipher," in *2013 IEEE Symposium on Security and Privacy, SP 2013, Berkeley, CA, USA, May 19-22, 2013*. IEEE Computer Society, 2013, pp. 478–492. [Online]. Available: https://doi.org/10.1109/SP.2013.39

[173] E. M. Songhori, S. U. Hussain, A. Sadeghi, T. Schneider, and F. Koushanfar, "Tinygarble: Highly compressed and scalable sequential garbled circuits," in *2015 IEEE Symposium on Security and Privacy, SP 2015, San Jose, CA, USA, May 17-21, 2015*. IEEE Computer Society, 2015, pp. 411–428. [Online]. Available: https://doi.org/10.1109/SP.2015.32

[174] D. Beaver, "Efficient multiparty protocols using circuit randomization," in *Advances in Cryptology - CRYPTO '91, 11th Annual International Cryptology Conference, Santa Barbara, California, USA, August 11-15, 1991, Proceedings*, ser. Lecture Notes in Computer Science, J. Feigenbaum, Ed., vol. 576. Springer, 1991, pp. 420–432. [Online]. Available: https://doi.org/10.1007/3-540-46766-1_34

[175] O. Goldreich, S. Micali, and A. Wigderson, "How to play any mental game or A completeness theorem for protocols with honest majority," in *Proceedings of the 19th Annual ACM Symposium on Theory of Computing, 1987, New York, New York, USA*, A. V. Aho, Ed. ACM, 1987, pp. 218–229. [Online]. Available: https://doi.org/10.1145/28395.28420

[176] M. Ben-Or, S. Goldwasser, and A. Wigderson, "Completeness theorems for non-cryptographic fault-tolerant distributed computation (extended abstract)," in *Proceedings of the 20th Annual ACM Symposium on Theory of Computing, May 2-4, 1988, Chicago, Illinois, USA*, J. Simon, Ed. ACM, 1988, pp. 1–10. [Online]. Available: https://doi.org/10.1145/62212.62213

[177] G. Asharov, Y. Lindell, T. Schneider, and M. Zohner, "More efficient oblivious transfer extensions," *J. Cryptol.*, vol. 30, no. 3, pp. 805–858, 2017. [Online]. Available: https://doi.org/10.1007/s00145-016-9236-6

[178] R. Bendlin, I. Damgård, C. Orlandi, and S. Zakarias, "Semi-homomorphic encryption and multiparty computation," in *Advances in Cryptology - EUROCRYPT 2011 - 30th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Tallinn, Estonia, May 15-19, 2011. Proceedings*, ser. Lecture Notes in Computer Science, K. G. Paterson, Ed., vol. 6632. Springer, 2011, pp. 169–188. [Online]. Available: https://doi.org/10.1007/978-3-642-20465-4_11

[179] J. B. Nielsen, P. S. Nordholt, C. Orlandi, and S. S. Burra, "A new approach to practical active-secure two-party computation," in *Advances in Cryptology - CRYPTO 2012 - 32nd Annual Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2012. Proceedings*, ser. Lecture Notes in Computer Science, R. Safavi-Naini and R. Canetti, Eds., vol. 7417. Springer, 2012, pp. 681–700. [Online]. Available: https://doi.org/10.1007/978-3-642-32009-5_40

[180] I. Damgård, V. Pastro, N. P. Smart, and S. Zakarias, "Multiparty computation from somewhat homomorphic encryption," in *Advances in Cryptology - CRYPTO 2012 - 32nd Annual Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2012. Proceedings*, ser. Lecture Notes in Computer Science, R. Safavi-Naini and R. Canetti, Eds., vol. 7417. Springer, 2012, pp. 643–662. [Online]. Available: https://doi.org/10.1007/978-3-642-32009-5_38

[181] M. Keller, E. Orsini, and P. Scholl, "MASCOT: faster malicious arithmetic secure computation with oblivious transfer," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, Vienna, Austria, October 24-28, 2016*, E. R. Weippl, S. Katzenbeisser, C. Kruegel, A. C. Myers, and S. Halevi, Eds. ACM, 2016, pp. 830–842. [Online]. Available: https://doi.org/10.1145/2976749.2978357

[182] ISO/IEC JTC 1/SC 27, WG2, "ISO/IEC DIS 4922-2: Information security — secure multiparty computation — part 2: Mechanisms based on secret sharing," International Organization for Standardization, Geneva, CH, Standard Draft, 2023. [Online]. Available: https://www.iso.org/standard/80514.html

[183] M. Albrecht, M. Chase, H. Chen, J. Ding, S. Goldwasser, S. Gorbunov, S. Halevi, J. Hoffstein, K. Laine, K. Lauter, S. Lokam, D. Micciancio, D. Moody, T. Morrison, A. Sahai, and V. Vaikuntanathan,

"Homomorphic encryption security standard," HomomorphicEncryption.org, Toronto, Canada, Tech. Rep., November 2018. [Online]. Available: http://homomorphicencryption.org/wp-content/uploads/2018/11/HomomorphicEncryptionStandardv1.1.pdf

[184] A. Acar, H. Aksu, A. S. Uluagac, and M. Conti, "A survey on homomorphic encryption schemes: Theory and implementation," *ACM Comput. Surv.*, vol. 51, no. 4, pp. 79:1–79:35, 2018. [Online]. Available: https://doi.org/10.1145/3214303

[185] C. Marcolla, V. Sucasas, M. Manzano, R. Bassoli, F. H. P. Fitzek, and N. Aaraj, "Survey on fully homomorphic encryption, theory, and applications," *Proc. IEEE*, vol. 110, no. 10, pp. 1572–1609, 2022. [Online]. Available: https://doi.org/10.1109/JPROC.2022.3205665

[186] P. Paillier, "Public-key cryptosystems based on composite degree residuosity classes," in *Advances in Cryptology - EUROCRYPT '99, International Conference on the Theory and Application of Cryptographic Techniques, Prague, Czech Republic, May 2-6, 1999, Proceeding*, ser. Lecture Notes in Computer Science, J. Stern, Ed., vol. 1592. Springer, 1999, pp. 223–238. [Online]. Available: https://doi.org/10.1007/3-540-48910-X_16

[187] M. Nassar, Q. M. Malluhi, and T. Khan, "A scheme for three-way secure and verifiable e-voting," in *15th IEEE/ACS International Conference on Computer Systems and Applications, AICCSA 2018, Aqaba, Jordan, October 28 - Nov. 1, 2018*. IEEE Computer Society, 2018, pp. 1–6. [Online]. Available: https://doi.org/10.1109/AICCSA.2018.8612810

[188] A. K. Vangujar, B. Ganesh, and P. Palmieri, "A novel approach to e-voting with group identity based identification and homomorphic encryption," *IACR Cryptol. ePrint Arch.*, p. 336, 2023. [Online]. Available: https://eprint.iacr.org/2023/336

[189] N. Zeilemaker, Z. Erkin, P. Palmieri, and J. A. Pouwelse, "Building a privacy-preserving semantic overlay for peer-to-peer networks," in *2013 IEEE International Workshop on Information Forensics and Security, WIFS 2013, Guangzhou, China, November 18-21, 2013*. IEEE, 2013, pp. 79–84. [Online]. Available: https://doi.org/10.1109/WIFS.2013.6707798

[190] L. Calderoni, P. Palmieri, and D. Maio, "Probabilistic properties of the spatial bloom filters and their relevance to cryptographic protocols," *IEEE Trans. Inf. Forensics Secur.*, vol. 13, no. 7, pp. 1710–1721, 2018. [Online]. Available: https://doi.org/10.1109/TIFS.2018.2799486

[191] M. Yasuda, T. Shimoyama, J. Kogure, K. Yokoyama, and T. Koshiba, "Secure pattern matching using somewhat homomorphic encryption," in *CCSW'13, Proceedings of the 2013 ACM Cloud Computing Security Workshop, Co-located with CCS 2013, Berlin, Germany, November 4, 2013*, A. Juels and B. Parno, Eds. ACM, 2013, pp. 65–76. [Online]. Available: https://doi.org/10.1145/2517488.2517497

[192] T. K. Saha, M. Rathee, and T. Koshiba, "Efficient private database queries using ring-lwe somewhat homomorphic encryption," *J. Inf. Secur. Appl.*, vol. 49, 2019. [Online]. Available: https://doi.org/10.1016/j.jisa.2019.102406

[193] L. Xiong and D. Dong, "Reversible data hiding in encrypted images with somewhat homomorphic encryption based on sorting block-level prediction-error expansion," *J. Inf. Secur. Appl.*, vol. 47, pp. 78–85, 2019. [Online]. Available: https://doi.org/10.1016/j.jisa.2019.04.005

[194] B. Ganesh and P. Palmieri, "A survey of advanced encryption for database security: Primitives, schemes, and attacks," in *Foundations and Practice of Security - 13th International Symposium, FPS 2020, Montreal, QC, Canada, December 1-3, 2020, Revised Selected Papers*, ser. Lecture Notes in Computer Science, G. Nicolescu, A. Tria, J. M. Fernandez, J. Marion, and J. García-Alfaro, Eds., vol. 12637. Springer, 2020, pp. 100–120. [Online]. Available: https://doi.org/10.1007/978-3-030-70881-8_7

[195] T. E. Gamal, "A public key cryptosystem and a signature scheme based on discrete logarithms," in *Advances in Cryptology, Proceedings of CRYPTO '84, Santa Barbara, California, USA, August 19-22, 1984, Proceedings*, ser. Lecture Notes in Computer Science, G. R. Blakley and D. Chaum, Eds., vol. 196. Springer, 1984, pp. 10–18. [Online]. Available: https://doi.org/10.1007/3-540-39568-7_2

[196] M. van Dijk, C. Gentry, S. Halevi, and V. Vaikuntanathan, "Fully homomorphic encryption over the integers," in *Advances in Cryptology - EUROCRYPT 2010, 29th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Monaco / French Riviera, May 30 - June 3, 2010. Proceedings*, ser. Lecture Notes in Computer Science, H. Gilbert, Ed., vol. 6110. Springer, 2010, pp. 24–43. [Online]. Available: https://doi.org/10.1007/978-3-642-13190-5_2

[197] J. Coron, A. Mandal, D. Naccache, and M. Tibouchi, "Fully homomorphic encryption over the integers with shorter public keys," in *Advances in Cryptology - CRYPTO 2011 - 31st Annual Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2011. Proceedings*, ser. Lecture Notes in Computer Science, P. Rogaway, Ed., vol. 6841. Springer, 2011, pp. 487–504. [Online]. Available: https://doi.org/10.1007/978-3-642-22792-9_28

[198] Z. Brakerski, C. Gentry, and V. Vaikuntanathan, "Fully homomorphic encryption without bootstrapping," *Electron. Colloquium Comput. Complex.*, vol. TR11-111, 2011. [Online]. Available: https://eccc.weizmann.ac.il/report/2011/111

[199] J. Fan and F. Vercauteren, "Somewhat practical fully homomorphic encryption," *IACR Cryptol. ePrint Arch.*, p. 144, 2012. [Online]. Available: http://eprint.iacr.org/2012/144

[200] Z. Brakerski, "Fully homomorphic encryption without modulus switching from classical gapsvp," in *Advances in Cryptology - CRYPTO 2012 - 32nd Annual Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2012. Proceedings*, ser. Lecture Notes in Computer Science, R. Safavi-Naini and R. Canetti, Eds., vol. 7417. Springer, 2012, pp. 868–886. [Online]. Available: https://doi.org/10.1007/978-3-642-32009-5_50

[201] C. Gentry, A. Sahai, and B. Waters, "Homomorphic encryption from learning with errors: Conceptually-simpler, asymptotically-faster, attribute-based," in *Advances in Cryptology - CRYPTO 2013 - 33rd Annual Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2013. Proceedings, Part I*, ser. Lecture Notes in Computer Science, R. Canetti and J. A. Garay, Eds., vol. 8042. Springer, 2013, pp. 75–92. [Online]. Available: https://doi.org/10.1007/978-3-642-40041-4_5

[202] I. Chillotti, N. Gama, M. Georgieva, and M. Izabachène, "Faster fully homomorphic encryption: Bootstrapping in less than 0.1 seconds," in *Advances in Cryptology - ASIACRYPT 2016 - 22nd International Conference on the Theory and Application of Cryptology and Information Security, Hanoi, Vietnam, December 4-8, 2016, Proceedings, Part I*, ser. Lecture Notes in Computer Science, J. H. Cheon and T. Takagi, Eds., vol. 10031, 2016, pp. 3–33. [Online]. Available: https://doi.org/10.1007/978-3-662-53887-6_1

[203] J. H. Cheon, A. Kim, M. Kim, and Y. S. Song, "Homomorphic encryption for arithmetic of approximate numbers," in *Advances in Cryptology - ASIACRYPT 2017 - 23rd International Conference on the Theory and Applications of Cryptology and Information Security, Hong Kong, China, December 3-7, 2017, Proceedings, Part I*, ser. Lecture Notes in Computer Science, T. Takagi and T. Peyrin, Eds., vol. 10624. Springer, 2017, pp. 409–437. [Online]. Available: https://doi.org/10.1007/978-3-319-70694-8_15

[204] C. Gentry, "A fully homomorphic encryption scheme," Ph.D. dissertation, Stanford University, USA, 2009. [Online]. Available: https://searchworks.stanford.edu/view/8493082

[205] I. Chillotti, N. Gama, M. Georgieva, and M. Izabachène, "TFHE: fast fully homomorphic encryption over the torus," *J. Cryptol.*, vol. 33, no. 1, pp. 34–91, 2020. [Online]. Available: https://doi.org/10.1007/s00145-019-09319-x

[206] L. Ducas and D. Micciancio, "FHEW: bootstrapping homomorphic encryption in less than a second," in *Advances in Cryptology - EUROCRYPT 2015 - 34th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Sofia, Bulgaria, April 26-30, 2015, Proceedings, Part I*, ser. Lecture Notes in Computer Science, E. Oswald and M. Fischlin, Eds., vol. 9056. Springer, 2015, pp. 617–640. [Online]. Available: https://doi.org/10.1007/978-3-662-46800-5_24

[207] "Microsoft SEAL," https://github.com/Microsoft/SEAL, Jan. 2023, Microsoft Research, Redmond, WA.

[208] S. Halevi and V. Shoup, "Algorithms in HElib," in *Advances in Cryptology - CRYPTO 2014 - 34th Annual Cryptology Conference, Santa Barbara, CA, USA, August 17-21, 2014, Proceedings, Part I*, ser. Lecture Notes in Computer Science, J. A. Garay and R. Gennaro, Eds., vol. 8616. Springer, 2014, pp. 554–571. [Online]. Available: https://doi.org/10.1007/978-3-662-44371-2_31

[209] K. Rohloff, D. Cousins, and Y. P. (project leads), "PALISADE homomorphic encryption software library," 2016. [Online]. Available: https://gitlab.com/palisade

[210] M. R. Albrecht, R. Player, and S. Scott, "On the concrete hardness of learning with errors," *J. Math. Cryptol.*, vol. 9, no. 3, pp. 169–203, 2015. [Online]. Available: http://www.degruyter.com/view/j/jmc.2015.9.issue-3/jmc-2015-0016/jmc-2015-0016.xml

[211] C. A. Melchor, M. Killijian, C. Lefebvre, and T. Ricosset, "A comparison of the homomorphic encryption libraries helib, SEAL and fv-nfllib," in *Innovative Security Solutions for Information Technology and Communications - 11th International Conference, SecITC 2018, Bucharest, Romania, November 8-9, 2018, Revised Selected Papers*, ser. Lecture Notes in Computer Science, J. Lanet and C. Toma, Eds., vol. 11359. Springer, 2018, pp. 425–442. [Online]. Available: https://doi.org/10.1007/978-3-030-12942-2_32

[212] A. A. Badawi, J. Bates, F. Bergamaschi, D. B. Cousins, S. Erabelli, N. Genise, S. Halevi, H. Hunt, A. Kim, Y. Lee, Z. Liu, D. Micciancio, I. Quah, Y. Polyakov, R. V. Saraswathy, K. Rohloff, J. Saylor, D. Suponitsky, M. Triplett, V. Vaikuntanathan, and V. Zucca, "Openfhe: Open-source fully homomorphic encryption library," in *Proceedings of the 10th Workshop on Encrypted Computing & Applied Homomorphic Cryptography, Los Angeles, CA, USA, 7 November 2022*, M. Brenner, A. Costache, and K. Rohloff, Eds. ACM, 2022, pp. 53–63. [Online]. Available: https://doi.org/10.1145/3560827.3563379

[213] I. Chillotti, N. Gama, M. Georgieva, and M. Izabachène, "TFHE: Fast fully homomorphic encryption library," 2016, https://tfhe.github.io/tfhe/.

[214] Zama, "Tfhe-rs: Rust implementation of tfhe," 2023, https://github.com/zama-ai/tfhe-rs. [Online]. Available: https://github.com/zama-ai/tfhe-rs

[215] A. Benaissa, B. Retiat, B. Cebere, and A. E. Belfedhal, "Tenseal: A library for encrypted tensor operations using homomorphic encryption," *CoRR*, vol. abs/2104.03152, 2021. [Online]. Available: https://arxiv.org/abs/2104.03152

[216] L. R. Peter Rindal, "libOTe: an efficient, portable, and easy to use Oblivious Transfer Library," https://github.com/osu-crypto/libOTe.

[217] Galois, Inc., "swanky: A suite of rust libraries for secure computation," https://github.com/GaloisInc/swanky, 2019.

[218] M. Bellare, V. T. Hoang, S. Keelveedhi, and P. Rogaway, "JustGarble software," https://github.com/irdan/justGarble.

[219] D. Demmler, T. Schneider, and M. Zohner, "ABY software," https://github.com/encryptogroup/ABY.

[220] N. Chandran, D. Gupta, A. Rastogi, R. Sharma, and S. Tripathi, "Ezpc: Programmable and efficient secure two-party computation for machine learning," in *IEEE European Symposium on Security and Privacy, EuroS&P 2019, Stockholm, Sweden, June 17-19, 2019*. IEEE, 2019, pp. 496–511. [Online]. Available: https://doi.org/10.1109/EuroSP.2019.00043

[221] M. Keller, "MP-SPDZ: A versatile framework for multi-party computation," in *CCS '20: 2020 ACM SIGSAC Conference on Computer and Communications Security, Virtual Event, USA, November 9-13, 2020*, J. Ligatti, X. Ou, J. Katz, and G. Vigna, Eds. ACM, 2020, pp. 1575–1590. [Online]. Available: https://doi.org/10.1145/3372297.3417872

[222] I. Damgård, M. Keller, E. Larraia, V. Pastro, P. Scholl, and N. P. Smart, "Practical covertly secure MPC for dishonest majority - or: Breaking the SPDZ limits," in *Computer Security - ESORICS 2013 - 18th European Symposium on Research in Computer Security, Egham, UK, September 9-13, 2013. Proceedings*, ser. Lecture Notes in Computer Science, J. Crampton, S. Jajodia, and K. Mayes, Eds., vol. 8134. Springer, 2013, pp. 1–18. [Online]. Available: https://doi.org/10.1007/978-3-642-40203-6_1

[223] M. Rivinius, P. Reisert, S. Hasler, and R. Küsters, "Convolutions in overdrive: Maliciously secure convolutions for MPC," *IACR Cryptol. ePrint Arch.*, p. 359, 2023. [Online]. Available: https://eprint.iacr.org/2023/359

[224] A. Aly, B. Coenen, K. Cong, K. Koch, M. Keller, D. Rotaru, O. Scherer, P. Scholl, N. P. Smart, T. Tanguy, and T. Wood, "SCALE-MAMBA," https://github.com/KULeuven-COSIC/SCALE-MAMBA.

[225] A. Brutzkus, R. Gilad-Bachrach, and O. Elisha, "Low latency privacy preserving inference," in *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, ser. Proceedings of Machine Learning Research, K. Chaudhuri and R. Salakhutdinov, Eds., vol. 97. PMLR, 2019, pp. 812–821. [Online]. Available: http://proceedings.mlr.press/v97/brutzkus19a.html

[226] E. Hesamifard, H. Takabi, and M. Ghasemi, "CryptoDL: Deep neural networks over encrypted data," *CoRR*, vol. abs/1711.05189, 2017. [Online]. Available: http://arxiv.org/abs/1711.05189

[227] E. Chou, J. Beal, D. Levy, S. Yeung, A. Haque, and L. Fei-Fei, "Faster cryptonets: Leveraging sparsity for real-world encrypted inference," *CoRR*, vol. abs/1811.09953, 2018. [Online]. Available: http://arxiv.org/abs/1811.09953

[228] A. Sanyal, M. J. Kusner, A. Gascón, and V. Kanade, "TAPAS: tricks to accelerate (encrypted) prediction as a service," in *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, ser. Proceedings of Machine Learning Research, J. G. Dy and A. Krause, Eds., vol. 80. PMLR, 2018, pp. 4497–4506. [Online]. Available: http://proceedings.mlr.press/v80/sanyal18a.html

[229] F. Bourse, M. Minelli, M. Minihold, and P. Paillier, "Fast homomorphic evaluation of deep discretized neural networks," in *Advances in Cryptology - CRYPTO 2018 - 38th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2018, Proceedings, Part III*, ser. Lecture Notes in Computer Science, H. Shacham and A. Boldyreva, Eds., vol. 10993. Springer, 2018, pp. 483–512. [Online]. Available: https://doi.org/10.1007/978-3-319-96878-0_17

[230] R. Dathathri, O. Saarikivi, H. Chen, K. Laine, K. E. Lauter, S. Maleki, M. Musuvathi, and T. Mytkowicz, "CHET: an optimizing compiler for fully-homomorphic neural-network inferencing," in *Proceedings of the 40th ACM SIGPLAN Conference on Programming Language Design and Implementation, PLDI 2019, Phoenix, AZ, USA, June 22-26, 2019*, K. S. McKinley and K. Fisher, Eds. ACM, 2019, pp. 142–156. [Online]. Available: https://doi.org/10.1145/3314221.3314628

[231] X. Wang, A. J. Malozemoff, and J. Katz. (2016) EMP-toolkit: Efficient multiparty computation toolkit. [Online]. Available: https://github.com/emp-toolkit

[232] A. Patra, T. Schneider, A. Suresh, and H. Yalame, "ABY2.0: improved mixed-protocol secure two-party computation," in *30th USENIX Security Symposium, USENIX Security 2021, August 11-13, 2021*, M. Bailey and R. Greenstadt, Eds. USENIX Association, 2021, pp. 2165–2182. [Online]. Available: https://www.usenix.org/conference/usenixsecurity21/presentation/patra

[233] P. Barrett, "Implementing the rivest shamir and adleman public key encryption algorithm on a standard digital signal processor," in *Advances in Cryptology — CRYPTO' 86*, A. M. Odlyzko, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 1987, pp. 311–323.

[234] F. Boemer, Y. Lao, R. Cammarota, and C. Wierzynski, "ngraph-he: a graph compiler for deep learning on homomorphically encrypted data," in *Proceedings of the 16th ACM International Conference on*

Computing Frontiers, CF 2019, Alghero, Italy, April 30 - May 2, 2019, F. Palumbo, M. Becchi, M. Schulz, and K. Sato, Eds. ACM, 2019, pp. 3–13. [Online]. Available: https://doi.org/10.1145/3310273.3323047

[235] F. Boemer, A. Costache, R. Cammarota, and C. Wierzynski, "ngraph-he2: A high-throughput framework for neural network inference on encrypted data," in *Proceedings of the 7th ACM Workshop on Encrypted Computing & Applied Homomorphic Cryptography, WAHC@CCS 2019, London, UK, November 11-15, 2019*, M. Brenner, T. Lepoint, and K. Rohloff, Eds. ACM, 2019, pp. 45–56. [Online]. Available: https://doi.org/10.1145/3338469.3358944

[236] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015, software available from tensorflow.org. [Online]. Available: https://www.tensorflow.org/

[237] Z. Huang, W. Lu, C. Hong, and J. Ding, "Cheetah: Lean and fast secure two-party deep neural network inference," in *31st USENIX Security Symposium, USENIX Security 2022, Boston, MA, USA, August 10-12, 2022*, K. R. B. Butler and K. Thomas, Eds. USENIX Association, 2022, pp. 809–826. [Online]. Available: https://www.usenix.org/conference/usenixsecurity22/presentation/huang-zhicong

[238] N. P. Smart and F. Vercauteren, "Fully homomorphic SIMD operations," *Des. Codes Cryptogr.*, vol. 71, no. 1, pp. 57–81, 2014. [Online]. Available: https://doi.org/10.1007/s10623-012-9720-4

[239] E. Boyle, G. Couteau, N. Gilboa, Y. Ishai, L. Kohl, P. Rindal, and P. Scholl, "Efficient two-round OT extension and silent non-interactive secure computation," in *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security, CCS 2019, London, UK, November 11-15, 2019*, L. Cavallaro, J. Kinder, X. Wang, and J. Katz, Eds. ACM, 2019, pp. 291–308. [Online]. Available: https://doi.org/10.1145/3319535.3354255

[240] [Online]. Available: https://github.com/secretflow/secretflow

[241] K. Bonawitz, V. Ivanov, B. Kreuter, A. Marcedone, H. B. McMahan, S. Patel, D. Ramage, A. Segal, and K. Seth, "Practical secure aggregation for privacy preserving machine learning," Cryptology ePrint Archive, Paper 2017/281, 2017, https://eprint.iacr.org/2017/281. [Online]. Available: https://eprint.iacr.org/2017/281

[242] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Z. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "Pytorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, H. M. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. B. Fox, and R. Garnett, Eds., 2019, pp. 8024–8035. [Online]. Available: https://proceedings.neurips.cc/paper/2019/hash/bdbca288fee7f92f2bfa9f7012727740-Abstract.html

[243] W. Wang, Y. Hu, L. Chen, X. Huang, and B. Sunar, "Accelerating fully homomorphic encryption using GPU," in *IEEE Conference on High Performance Extreme Computing, HPEC 2012, Waltham, MA, USA, September 10-12, 2012*. IEEE, 2012, pp. 1–5. [Online]. Available: https://doi.org/10.1109/HPEC.2012.6408660

[244] C. Gentry and S. Halevi, "Implementing gentry's fully-homomorphic encryption scheme," in *Advances in Cryptology - EUROCRYPT 2011 - 30th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Tallinn, Estonia, May 15-19, 2011. Proceedings*, ser. Lecture Notes in Computer Science, K. G. Paterson, Ed., vol. 6632. Springer, 2011, pp. 129–148. [Online]. Available: https://doi.org/10.1007/978-3-642-20465-4_9

[245] W. Wang, Z. Chen, and X. Huang, "Accelerating leveled fully homomorphic encryption using GPU," in *IEEE International Symposium on Circuits and Systemss, ISCAS 2014, Melbourne, Victoria, Australia, June 1-5, 2014*. IEEE, 2014, pp. 2800–2803. [Online]. Available: https://doi.org/10.1109/ISCAS.2014.6865755

[246] W. Dai and B. Sunar, "cuHE: A homomorphic encryption accelerator library," in *Cryptography and Information Security in the Balkans - Second International Conference, BalkanCryptSec 2015, Koper, Slovenia, September 3-4, 2015, Revised Selected Papers*, ser. Lecture Notes in Computer Science, E. Pasalic and L. R. Knudsen, Eds., vol. 9540. Springer, 2015, pp. 169–186. [Online]. Available: https://doi.org/10.1007/978-3-319-29172-7_11

[247] W. Dai, Y. Doröz, and B. Sunar, "Accelerating SWHE based pirs using gpus," in *Financial Cryptography and Data Security - FC 2015 International Workshops, BITCOIN, WAHC, and Wearable, San Juan, Puerto Rico, January 30, 2015, Revised Selected Papers*, ser. Lecture Notes in Computer Science, M. Brenner, N. Christin, B. Johnson, and K. Rohloff, Eds., vol. 8976. Springer, 2015, pp. 160–171. [Online]. Available: https://doi.org/10.1007/978-3-662-48051-9_12

[248] A. A. Badawi, B. Veeravalli, C. F. Mun, and K. M. M. Aung, "High-performance FV somewhat homomorphic encryption on gpus: An implementation using CUDA," *IACR Trans. Cryptogr. Hardw. Embed. Syst.*, vol. 2018, no. 2, pp. 70–95, 2018. [Online]. Available: https://doi.org/10.13154/tches.v2018.i2.70-95

[249] S. Kim, W. Jung, J. Park, and J. H. Ahn, "Accelerating number theoretic transformations for bootstrappable homomorphic encryption on gpus," in *IEEE International Symposium on Workload Characterization, IISWC 2020, Beijing, China, October 27-30, 2020*. IEEE, 2020, pp. 264–275. [Online]. Available: https://doi.org/10.1109/IISWC50251.2020.00033

[250] J. Goey, W. Lee, B. Goi, and W. Yap, "Accelerating number theoretic transform in GPU platform for fully homomorphic encryption," *J. Supercomput.*, vol. 77, no. 2, pp. 1455–1474, 2021. [Online]. Available: https://doi.org/10.1007/s11227-020-03156-7

[251] K. Shivdikar, G. Jonatan, E. Mora, N. Livesay, R. Agrawal, A. Joshi, J. L. Abellán, J. Kim, and D. R. Kaeli, "Accelerating polynomial multiplication for homomorphic encryption on gpus," in *2022 IEEE International Symposium on Secure and Private Execution Environment Design (SEED), Storrs, CT, USA, September 26-27, 2022*. IEEE, 2022, pp. 61–72. [Online]. Available: https://doi.org/10.1109/SEED55351.2022.00013

[252] Ö. Özerk, C. Elgezen, A. C. Mert, E. Öztürk, and E. Savas, "Efficient number theoretic transform implementation on GPU for homomorphic encryption," *J. Supercomput.*, vol. 78, no. 2, pp. 2840–2872, 2022. [Online]. Available: https://doi.org/10.1007/s11227-021-03980-5

[253] A. Ş. Özcan, C. Ayduman, E. R. Türkoğlu, and E. Savaş, "Homomorphic encryption on GPU," *IEEE Access*, 2023.

[254] I. Chillotti, N. Gama, M. Georgieva, and M. Izabachène, "Faster fully homomorphic encryption: Bootstrapping in less than 0.1 seconds," in *Advances in Cryptology - ASIACRYPT 2016 - 22nd International Conference on the Theory and Application of Cryptology and Information Security, Hanoi, Vietnam, December 4-8, 2016, Proceedings, Part I*, ser. Lecture Notes in Computer Science, J. H. Cheon and T. Takagi, Eds., vol. 10031, 2016, pp. 3–33. [Online]. Available: https://doi.org/10.1007/978-3-662-53887-6_1

[255] T. Morshed, M. M. A. Aziz, and N. Mohammed, "CPU and GPU accelerated fully homomorphic encryption," in *2020 IEEE International Symposium on Hardware Oriented Security and Trust, HOST 2020, San Jose, CA, USA, December 7-11, 2020*. IEEE, 2020, pp. 142–153. [Online]. Available: https://doi.org/10.1109/HOST45689.2020.9300288

[256] W. Jung, S. Kim, J. H. Ahn, J. H. Cheon, and Y. Lee, "Over 100x faster bootstrapping in fully homomorphic encryption through memory-centric optimization with gpus," *IACR Trans. Cryptogr. Hardw. Embed. Syst.*, vol. 2021, no. 4, pp. 114–148, 2021. [Online]. Available: https://doi.org/10.46586/tches.v2021.i4.114-148

[257] F. Boemer, S. Kim, G. Seifu, F. D. M. de Souza, and V. Gopal, "Intel HEXL: accelerating homomorphic encryption with intel AVX512-IFMA52," in *WAHC '21: Proceedings of the 9th on Workshop on Encrypted Computing & Applied Homomorphic Cryptography, Virtual Event, Korea, 15 November 2021*. WAHC@ACM, 2021, pp. 57–62. [Online]. Available: https://doi.org/10.1145/3474366.3486926

[258] Y. Doröz, E. Öztürk, and B. Sunar, "Accelerating fully homomorphic encryption in hardware," *IEEE Trans. Computers*, vol. 64, no. 6, pp. 1509–1521, 2015. [Online]. Available: https://doi.org/10.1109/TC.2014.2345388

[259] J. Bajard, J. Eynard, M. A. Hasan, and V. Zucca, "A full RNS variant of FV like somewhat homomorphic encryption schemes," in *Selected Areas in Cryptography - SAC 2016 - 23rd International Conference, St. John's, NL, Canada, August 10-12, 2016, Revised Selected Papers*, ser. Lecture Notes in Computer Science, R. Avanzi and H. M. Heys, Eds., vol. 10532. Springer, 2016, pp. 423–442. [Online]. Available: https://doi.org/10.1007/978-3-319-69453-5_23

[260] A. A. Karatsuba and Y. Ofman, "Multiplication of multidigit numbers on automata," *Soviet physics. Doklady*, vol. 7, pp. 595–596, 1963.

[261] V. Migliore, M. M. Real, V. Lapotre, A. Tisserand, C. Fontaine, and G. Gogniat, "Hardware/software co-design of an accelerator for FV homomorphic encryption scheme using karatsuba algorithm," *IEEE Trans. Computers*, vol. 67, no. 3, pp. 335–347, 2018. [Online]. Available: https://doi.org/10.1109/TC.2016.2645204

[262] P. L. Montgomery, "Modular multiplication without trial division," *Mathematics of Computation*, vol. 44, pp. 519–521, 1985.

[263] M. Nabeel, D. Soni, M. Ashraf, M. Gebremichael, H. Gamil, E. Chielle, R. Karri, M. Sanduleanu, and M. Maniatakos, "Cofhee: A co-processor for fully homomorphic encryption execution," in *2023 Design, Automation and Test in Europe Conference and Exhibition, DATE 2023 - Proceedings*, ser. Proceedings -Design, Automation and Test in Europe, DATE. Institute of Electrical and Electronics Engineers Inc., 2023, publisher Copyright: © 2023 EDAA.; 2023 Design, Automation and Test in Europe Conference and Exhibition, DATE 2023 ; Conference date: 17-04-2023 Through 19-04-2023.

[264] Y. Su, B. Yang, C. Yang, and S. Zhao, "Remca: A reconfigurable multi-core architecture for full RNS variant of BFV homomorphic evaluation," *IEEE Trans. Circuits Syst. I Regul. Pap.*, vol. 69, no. 7, pp. 2857–2870, 2022. [Online]. Available: https://doi.org/10.1109/TCSI.2022.3163970

[265] S. S. Roy, F. Turan, K. Järvinen, F. Vercauteren, and I. Verbauwhede, "Fpga-based high-performance parallel architecture for homomorphic computing on encrypted data," in *25th IEEE International Symposium on High Performance Computer Architecture, HPCA 2019, Washington, DC, USA, February 16-20, 2019*. IEEE, 2019, pp. 387–398. [Online]. Available: https://doi.org/10.1109/HPCA.2019.00052

[266] F. Turan, S. S. Roy, and I. Verbauwhede, "HEAWS: an accelerator for homomorphic encryption on the amazon AWS FPGA," *IEEE Trans. Computers*, vol. 69, no. 8, pp. 1185–1196, 2020. [Online]. Available: https://doi.org/10.1109/TC.2020.2988765

[267] M. S. Riazi, K. Laine, B. Pelton, and W. Dai, "HEAX: an architecture for computing on encrypted data," in *ASPLOS '20: Architectural Support for Programming Languages and Operating Systems, Lausanne, Switzerland, March 16-20, 2020*, J. R. Larus, L. Ceze, and K. Strauss, Eds. ACM, 2020, pp. 1295–1309. [Online]. Available: https://doi.org/10.1145/3373376.3378523

[268] A. C. Mert, Aikata, S. Kwon, Y. Shin, D. Yoo, Y. Lee, and S. S. Roy, "Medha: Microcoded hardware accelerator for computing on encrypted data," *IACR Trans. Cryptogr. Hardw. Embed. Syst.*, vol. 2023, no. 1, pp. 463–500, 2023. [Online]. Available: https://doi.org/10.46586/tches.v2023.i1.463-500

[269] N. Samardzic, A. Feldmann, A. Krastev, S. Devadas, R. Dreslinski, C. Peikert, and D. Sanchez, "F1: A fast and programmable accelerator for fully homomorphic encryption," in *MICRO-54: 54th Annual IEEE/ACM International Symposium on Microarchitecture*, ser. MICRO '21. New York, NY, USA: Association for Computing Machinery, 2021, p. 238–252. [Online]. Available: https://doi.org/10.1145/3466752.3480070

[270] S. Kim, J. Kim, M. J. Kim, W. Jung, M. Rhu, J. Kim, and J. H. Ahn, "Bts: an accelerator for bootstrappable fully homomorphic encryption," *Proceedings of the 49th Annual International Symposium on Computer Architecture*, 2021.

[271] R. Geelen, M. V. Beirendonck, H. V. L. Pereira, B. Huffman, T. McAuley, B. Selfridge, D. Wagner, G. Dimou, I. Verbauwhede, F. Vercauteren, and D. W. Archer, "BASALISC: flexible asynchronous hardware accelerator for fully homomorphic encryption," *CoRR*, vol. abs/2205.14017, 2022. [Online]. Available: https://doi.org/10.48550/arXiv.2205.14017

[272] N. Samardzic, A. Feldmann, A. Krastev, N. Manohar, N. Genise, S. Devadas, K. Eldefrawy, C. Peikert, and D. Sanchez, "Craterlake: A hardware accelerator for efficient unbounded computation on encrypted data," in *Proceedings of the 49th Annual International Symposium on Computer Architecture*, ser. ISCA '22. New York, NY, USA: Association for Computing Machinery, 2022, p. 173–187. [Online]. Available: https://doi.org/10.1145/3470496.3527393

[273] J. Kim, G. Lee, S. Kim, G. Sohn, J. Kim, M. Rhu, and J. H. Ahn, "Ark: Fully homomorphic encryption accelerator with runtime data generation and inter-operation key reuse," *2022 55th IEEE/ACM International Symposium on Microarchitecture (MICRO)*, pp. 1237–1254, 2022.

[274] D. B. Cousins, Y. Polyakov, A. A. Badawi, M. French, A. Schmidt, A. Jacob, B. Reynwar, K. Canida, A. R. Jaiswal, C. Mathew, H. Gamil, N. Neda, D. Soni, M. Maniatakos, B. Reagen, N. Zhang, F. Franchetti, P. Brinich, J. Johnson, P. Broderick, M. Franusich, B. Zhang, Z. Cheng, and M. Pedram, "TREBUCHET: fully homomorphic encryption accelerator for deep computation," *CoRR*, vol. abs/2304.05237, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2304.05237

[275] A. C. Mert, E. Öztürk, and E. Savas, "Design and implementation of a fast and scalable ntt-based polynomial multiplier architecture," *IACR Cryptol. ePrint Arch.*, p. 109, 2019. [Online]. Available: https://eprint.iacr.org/2019/109

[276] Z. Brakerski, C. Gentry, and V. Vaikuntanathan, "Fully homomorphic encryption without bootstrapping," Cryptology ePrint Archive, Paper 2011/277, 2011, https://eprint.iacr.org/2011/277. [Online]. Available: https://eprint.iacr.org/2011/277

[277] E. Biham, "A fast new des implementation in software," in *Fast Software Encryption*, E. Biham, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 1997, pp. 260–272.

[278] A. Adomnicai, Z. Najm, and T. Peyrin, "Fixslicing: A new gift representation: Fast constant-time implementations of gift and gift-cofb on arm cortex-m," *IACR Transactions on Cryptographic Hardware and Embedded Systems*, vol. 2020, no. 3, p. 402–427, Jun. 2020. [Online]. Available: https://tches.iacr.org/index.php/TCHES/article/view/8595

[279] E. Käsper and P. Schwabe, "Faster and timing-attack resistant aes-gcm," in *Cryptographic Hardware and Embedded Systems - CHES 2009*, C. Clavier and K. Gaj, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 1–17.

[280] W. Zhang, Z. Bao, D. Lin, V. Rijmen, B. Yang, and I. Verbauwhede, "Rectangle: A bit-slice lightweight block cipher suitable for multiple platforms," Cryptology ePrint Archive, Paper 2014/084, 2014, https://eprint.iacr.org/2014/084. [Online]. Available: https://eprint.iacr.org/2014/084

[281] T. Güneysu, T. Oder, T. Pöppelmann, and P. Schwabe, "Software speed records for lattice-based signatures," in *Post-Quantum Cryptography*, P. Gaborit, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 67–82.

[282] A. Journault and F.-X. Standaert, "Very high order masking: Efficient implementation and security evaluation," in *Cryptographic Hardware and Embedded Systems – CHES 2017*, W. Fischer and N. Homma, Eds. Cham: Springer International Publishing, 2017, pp. 623–643.

[283] J. H. Cheon, J.-S. Coron, J. Kim, M. S. Lee, T. Lepoint, M. Tibouchi, and A. Yun, "Batch fully homomorphic encryption over the integers," in *Advances in Cryptology – EUROCRYPT 2013*, T. Johansson and P. Q. Nguyen, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 315–335.

[284] I. Chillotti, N. Gama, M. Georgieva, and M. Izabachène, "Faster packed homomorphic operations and efficient circuit bootstrapping for tfhe," in *Advances in Cryptology – ASIACRYPT 2017*, T. Takagi and T. Peyrin, Eds. Cham: Springer International Publishing, 2017, pp. 377–408.

[285] S. Sinha, S. Saha, M. Alam, V. Agarwal, A. Chatterjee, A. Mishra, D. Khazanchi, and D. Mukhopadhyay, "Exploring bitslicing architectures for enabling fhe-assisted machine learning," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 41, no. 11, pp. 4004–4015, 2022.

[286] J. W. Bos, D. A. Osvik, and D. Stefan, "Fast implementations of aes on various platforms," *Cryptology ePrint Archive*, 2009.

[287] D. Canright, "A very compact s-box for AES," in *Cryptographic Hardware and Embedded Systems - CHES 2005, 7th International Workshop, Edinburgh, UK, August 29 - September 1, 2005, Proceedings*, ser. Lecture Notes in Computer Science, J. R. Rao and B. Sunar, Eds., vol. 3659. Springer, 2005, pp. 441–455. [Online]. Available: https://doi.org/10.1007/11545262_32

[288] B. Pinkas, T. Schneider, N. P. Smart, and S. C. Williams, "Secure two-party computation is practical," in *Advances in Cryptology - ASIACRYPT 2009, 15th International Conference on the Theory and Application of Cryptology and Information Security, Tokyo, Japan, December 6-10, 2009. Proceedings*, ser. Lecture Notes in Computer Science, M. Matsui, Ed., vol. 5912. Springer, 2009, pp. 250–267. [Online]. Available: https://doi.org/10.1007/978-3-642-10366-7_15

[289] I. Damgård and M. Keller, "Secure multiparty AES," in *Financial Cryptography and Data Security, 14th International Conference, FC 2010, Tenerife, Canary Islands, Spain, January 25-28, 2010, Revised Selected Papers*, ser. Lecture Notes in Computer Science, R. Sion, Ed., vol. 6052. Springer, 2010, pp. 367–374. [Online]. Available: https://doi.org/10.1007/978-3-642-14577-3_31

[290] I. Damgård, M. Keller, E. Larraia, C. Miles, and N. P. Smart, "Implementing AES via an actively/covertly secure dishonest-majority MPC protocol," in *Security and Cryptography for Networks - 8th International Conference, SCN 2012, Amalfi, Italy, September 5-7, 2012. Proceedings*, ser. Lecture Notes in Computer Science, I. Visconti and R. D. Prisco, Eds., vol. 7485. Springer, 2012, pp. 241–263. [Online]. Available: https://doi.org/10.1007/978-3-642-32928-9_14

[291] L. R. Knudsen and G. Leander, "PRESENT - block cipher," in *Encyclopedia of Cryptography and Security, 2nd Ed*, H. C. A. van Tilborg and S. Jajodia, Eds. Springer, 2011, pp. 953–955. [Online]. Available: https://doi.org/10.1007/978-1-4419-5906-5_605

[292] C. Dobraunig, M. Eichlseder, F. Mendel, and M. Schläffer, "Ascon v1.2: Lightweight authenticated encryption and hashing," *J. Cryptol.*, vol. 34, no. 3, p. 33, 2021. [Online]. Available: https://doi.org/10.1007/s00145-021-09398-9

[293] B. Gérard, V. Grosso, M. Naya-Plasencia, and F. Standaert, "Block ciphers that are easier to mask: How far can we go?" in *Cryptographic Hardware and Embedded Systems - CHES 2013 - 15th International Workshop, Santa Barbara, CA, USA, August 20-23, 2013. Proceedings*, ser. Lecture Notes in Computer Science, G. Bertoni and J. Coron, Eds., vol. 8086. Springer, 2013, pp. 383–399. [Online]. Available: https://doi.org/10.1007/978-3-642-40349-1_22

[294] V. Grosso, G. Leurent, F. Standaert, and K. Varici, "Ls-designs: Bitslice encryption for efficient masked software implementations," in *Fast Software Encryption - 21st International Workshop, FSE 2014, London, UK, March 3-5, 2014. Revised Selected Papers*, ser. Lecture Notes in Computer Science, C. Cid and C. Rechberger, Eds., vol. 8540. Springer, 2014, pp. 18–37. [Online]. Available: https://doi.org/10.1007/978-3-662-46706-0_2

[295] M. O. Saarinen, "Cryptographic analysis of all 4 × 4-bit s-boxes," in *Selected Areas in Cryptography - 18th International Workshop, SAC 2011, Toronto, ON, Canada, August 11-12, 2011, Revised Selected Papers*, ser. Lecture Notes in Computer Science, A. Miri and S. Vaudenay, Eds., vol. 7118. Springer, 2011, pp. 118–133. [Online]. Available: https://doi.org/10.1007/978-3-642-28496-0_7

[296] M. R. Albrecht, C. Rechberger, T. Schneider, T. Tiessen, and M. Zohner, "Ciphers for MPC and FHE," in *Advances in Cryptology - EUROCRYPT 2015 - 34th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Sofia, Bulgaria, April 26-30, 2015, Proceedings, Part I*, ser. Lecture Notes in Computer Science, E. Oswald and M. Fischlin, Eds., vol. 9056. Springer, 2015, pp. 430–454. [Online]. Available: https://doi.org/10.1007/978-3-662-46800-5_17

[297] J. Daemen and V. Rijmen, "The wide trail design strategy," in *Cryptography and Coding, 8th IMA International Conference, Cirencester, UK, December 17-19, 2001, Proceedings*, ser. Lecture Notes in Computer Science, B. Honary, Ed., vol. 2260. Springer, 2001, pp. 222–238. [Online]. Available: https://doi.org/10.1007/3-540-45325-3_20

[298] M. Chase, D. Derler, S. Goldfeder, C. Orlandi, S. Ramacher, C. Rechberger, D. Slamanig, and G. Zaverucha, "Post-quantum zero-knowledge and signatures from symmetric-key primitives," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, CCS 2017, Dallas, TX, USA, October 30 - November 03, 2017*, B. Thuraisingham, D. Evans, T. Malkin, and D. Xu, Eds. ACM, 2017, pp. 1825–1842. [Online]. Available: https://doi.org/10.1145/3133956.3133997

[299] M. R. Albrecht, L. Grassi, C. Rechberger, A. Roy, and T. Tiessen, "Mimc: Efficient encryption and cryptographic hashing with minimal multiplicative complexity," in *Advances in Cryptology - ASIACRYPT 2016 - 22nd International Conference on the Theory and Application of Cryptology and Information Security, Hanoi, Vietnam, December 4-8, 2016, Proceedings, Part I*, ser. Lecture Notes in Computer Science, J. H. Cheon and T. Takagi, Eds., vol. 10031, 2016, pp. 191–219. [Online]. Available: https://doi.org/10.1007/978-3-662-53887-6_7

[300] G. Bertoni, J. Daemen, M. Peeters, and G. Van Assche, "Duplexing the sponge: Single-pass authenticated encryption and other applications," in *Selected Areas in Cryptography*, A. Miri and S. Vaudenay, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 320–337.

[301] M. Luby and C. Rackoff, "How to construct pseudo-random permutations from pseudo-random functions," in *Advances in Cryptology — CRYPTO '85 Proceedings*, H. C. Williams, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 1986, pp. 447–447.

[302] M. R. Albrecht, L. Grassi, L. Perrin, S. Ramacher, C. Rechberger, D. Rotaru, A. Roy, and M. Schofnegger, "Feistel structures for mpc, and more," in *Computer Security - ESORICS 2019 - 24th European Symposium on Research in Computer Security, Luxembourg, September 23-27, 2019, Proceedings, Part II*, ser. Lecture Notes in Computer Science, K. Sako, S. A. Schneider, and P. Y. A. Ryan, Eds., vol. 11736. Springer, 2019, pp. 151–171. [Online]. Available: https://doi.org/10.1007/978-3-030-29962-0_8

[303] E. Ben-Sasson, A. Chiesa, C. Garman, M. Green, I. Miers, E. Tromer, and M. Virza, "Zerocash: Decentralized anonymous payments from bitcoin," in *2014 IEEE Symposium on Security and Privacy, SP 2014, Berkeley, CA, USA, May 18-21, 2014*. IEEE Computer Society, 2014, pp. 459–474. [Online]. Available: https://doi.org/10.1109/SP.2014.36

[304] Y. Wang, W. Wu, Z. Guo, and X. Yu, "Differential cryptanalysis and linear distinguisher of full-round zorro," in *Applied Cryptography and Network Security*, I. Boureanu, P. Owesarski, and S. Vaudenay, Eds. Cham: Springer International Publishing, 2014, pp. 308–323.

[305] S. Rasoolzadeh, Z. Ahmadian, M. Salmasizadeh, and M. R. Aref, "Total break of zorro using linear and differential attacks," *The ISC International Journal of Information Security*, vol. 6, no. 1, pp. 23–34, 2014. [Online]. Available: https://www.isecure-journal.com/article_39149.html

[306] I. Dinur, Y. Liu, W. Meier, and Q. Wang, "Optimized interpolation attacks on lowmc," in *Advances in Cryptology – ASIACRYPT 2015*, T. Iwata and J. H. Cheon, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2015, pp. 535–560.

[307] X. Bonnetain, "Collisions on feistel-mimc and univariate gmimc," *IACR Cryptol. ePrint Arch.*, p. 951, 2019. [Online]. Available: https://eprint.iacr.org/2019/951

[308] L. Grassi, R. Lüftenegger, C. Rechberger, D. Rotaru, and M. Schofnegger, "On a generalization of substitution-permutation networks: The HADES design strategy," in *Advances in Cryptology - EUROCRYPT 2020 - 39th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, May 10-14, 2020, Proceedings, Part II*, ser. Lecture Notes in Computer Science, A. Canteaut and Y. Ishai, Eds., vol. 12106. Springer, 2020, pp. 674–704. [Online]. Available: https://doi.org/10.1007/978-3-030-45724-2_23

[309] L. Grassi, D. Khovratovich, C. Rechberger, A. Roy, and M. Schofnegger, "Poseidon: A new hash function for zero-knowledge proof systems," in *30th USENIX Security Symposium, USENIX Security 2021, August 11-13, 2021*, M. Bailey and R. Greenstadt, Eds. USENIX Association, 2021, pp. 519–535. [Online]. Available: https://www.usenix.org/conference/usenixsecurity21/presentation/grassi

[310] L. Grassi, D. Khovratovich, and M. Schofnegger, "Poseidon2: A faster version of the poseidon hash function," *IACR Cryptol. ePrint Arch.*, p. 323, 2023. [Online]. Available: https://eprint.iacr.org/2023/323

[311] A. Aly, T. Ashur, E. Ben-Sasson, S. Dhooghe, and A. Szepieniec, "Design of symmetric-key primitives for advanced cryptographic protocols," *IACR Trans. Symmetric Cryptol.*, vol. 2020, no. 3, pp. 1–45, 2020. [Online]. Available: https://doi.org/10.13154/tosc.v2020.i3.1-45

[312] R. C. Merkle, "A certified digital signature," in *Advances in Cryptology — CRYPTO' 89 Proceedings*, G. Brassard, Ed. New York, NY: Springer New York, 1990, pp. 218–238.

[313] E. Ben-Sasson, I. Bentov, Y. Horesh, and M. Riabzev, "Scalable, transparent, and post-quantum secure computational integrity," *IACR Cryptol. ePrint Arch.*, p. 46, 2018. [Online]. Available: http://eprint.iacr.org/2018/046

[314] M. R. Albrecht, C. Cid, L. Grassi, D. Khovratovich, R. Lüftenegger, C. Rechberger, and M. Schofnegger, "Algebraic cryptanalysis of stark-friendly designs: Application to marvellous and mimc," in *Advances in Cryptology – ASIACRYPT 2019*, S. D. Galbraith and S. Moriai, Eds. Cham: Springer International Publishing, 2019, pp. 371–397.

[315] A. Aly, T. Ashur, E. Ben-Sasson, S. Dhooghe, and A. Szepieniec, "Design of symmetric-key primitives for advanced cryptographic protocols," *IACR Transactions on Symmetric Cryptology*, vol. 2020, no. 3, p. 1–45, Sep. 2020. [Online]. Available: https://tosc.iacr.org/index.php/ToSC/article/view/8695

[316] T. Ashur, M. Mahzoun, and D. Toprakhisar, "Chaghri - A fhe-friendly block cipher," in *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security, CCS 2022, Los Angeles, CA, USA, November 7-11, 2022*, H. Yin, A. Stavrou, C. Cremers, and E. Shi, Eds. ACM, 2022, pp. 139–150. [Online]. Available: https://doi.org/10.1145/3548606.3559364

[317] F. Liu, R. Anand, L. Wang, W. Meier, and T. Isobe, "Coefficient grouping: Breaking chaghri and more," in *Advances in Cryptology - EUROCRYPT 2023 - 42nd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Lyon, France, April 23-27, 2023, Proceedings, Part IV*, ser. Lecture Notes in Computer Science, C. Hazay and M. Stam, Eds., vol. 14007. Springer, 2023, pp. 287–317. [Online]. Available: https://doi.org/10.1007/978-3-031-30634-1_10

[318] M. Naehrig, K. Lauter, and V. Vaikuntanathan, "Can homomorphic encryption be practical?" in *Proceedings of the 3rd ACM Workshop on Cloud Computing Security Workshop*, ser. CCSW '11. New York, NY, USA: Association for Computing Machinery, 2011, p. 113–124. [Online]. Available: https://doi.org/10.1145/2046660.2046682

[319] C. De Cannière, "Trivium: A stream cipher construction inspired by block cipher design principles," in *Information Security*, S. K. Katsikas, J. López, M. Backes, S. Gritzalis, and B. Preneel, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 171–186.

[320] A. Canteaut, S. Carpov, C. Fontaine, T. Lepoint, M. Naya-Plasencia, P. Paillier, and R. Sirdey, "Stream ciphers: A practical solution for efficient homomorphic-ciphertext compression," *J. Cryptol.*, vol. 31, no. 3, p. 885–916, jul 2018. [Online]. Available: https://doi.org/10.1007/s00145-017-9273-9

[321] P. Méaux, A. Journault, F.-X. Standaert, and C. Carlet, "Towards stream ciphers for efficient fhe with low-noise ciphertexts," in *Advances in Cryptology – EUROCRYPT 2016*, M. Fischlin and J.-S. Coron, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2016, pp. 311–343.

[322] P. Méaux, C. Carlet, A. Journault, and F.-X. Standaert, "Improved filter permutators for efficient fhe: Better instances and implementations," in *Progress in Cryptology – INDOCRYPT 2019*, F. Hao, S. Ruj, and S. Sen Gupta, Eds. Cham: Springer International Publishing, 2019, pp. 68–91.

[323] C. Dobraunig, M. Eichlseder, L. Grassi, V. Lallemand, G. Leander, E. List, F. Mendel, and C. Rechberger, "Rasta: A cipher with low anddepth and few ands per bit," in *Advances in Cryptology – CRYPTO 2018: 38th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19–23, 2018, Proceedings, Part I*. Berlin, Heidelberg: Springer-Verlag, 2018, p. 662–692. [Online]. Available: https://doi.org/10.1007/978-3-319-96884-1_22

[324] A. Biryukov, C. Bouillaguet, and D. Khovratovich, "Cryptographic schemes based on the ASASA structure: Black-box, white-box, and public-key (extended abstract)," in *Advances in Cryptology - ASIACRYPT 2014 - 20th International Conference on the Theory and Application of Cryptology and Information Security, Kaoshiung, Taiwan, R.O.C., December 7-11, 2014. Proceedings, Part I*, ser. Lecture Notes in Computer Science, P. Sarkar and T. Iwata, Eds., vol. 8873. Springer, 2014, pp. 63–84. [Online]. Available: https://doi.org/10.1007/978-3-662-45611-8_4

[325] P. Hebborn and G. Leander, "Dasta – alternative linear layer for rasta," *IACR Transactions on Symmetric Cryptology*, vol. 2020, Issue 3, pp. 46–86, 2020. [Online]. Available: https://tosc.iacr.org/index.php/ToSC/article/view/8696

[326] G. Bertoni, J. Daemen, M. Peeters, and G. Van Assche, "Keccak," in *Advances in Cryptology – EUROCRYPT 2013*, T. Johansson and P. Q. Nguyen, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 313–314.

[327] J. Ha, S. Kim, W. Choi, J. Lee, D. Moon, H. Yoon, and J. Cho, "Masta: An he-friendly cipher using modular arithmetic," *IEEE Access*, vol. 8, pp. 194 741–194 751, 2020.

[328] C. Dobraunig, L. Grassi, L. Helminger, C. Rechberger, M. Schofnegger, and R. Walch, "Pasta: A case for hybrid homomorphic encryption," Cryptology ePrint Archive, Paper 2021/731, 2021, https://eprint.iacr.org/2021/731. [Online]. Available: https://eprint.iacr.org/2021/731

[329] C. Cid, J. P. Indrøy, and H. Raddum, "Fasta – a stream cipher for fast fhe evaluation," in *Topics in Cryptology – CT-RSA 2022*, S. D. Galbraith, Ed. Cham: Springer International Publishing, 2022, pp. 451–483.

[330] J. Cho, J. Ha, S. Kim, B. Lee, J. Lee, J. Lee, D. Moon, and H. Yoon, "Transciphering framework for approximate homomorphic encryption," in *Advances in Cryptology - ASIACRYPT 2021 - 27th International Conference on the Theory and Application of Cryptology and Information Security, Singapore, December 6-10, 2021, Proceedings, Part III*, ser. Lecture Notes in Computer Science, M. Tibouchi and H. Wang, Eds., vol. 13092. Springer, 2021, pp. 640–669. [Online]. Available: https://doi.org/10.1007/978-3-030-92078-4_22

[331] J. Ha, S. Kim, B. Lee, J. Lee, and M. Son, "Rubato: Noisy ciphers for approximate homomorphic encryption," in *Advances in Cryptology - EUROCRYPT 2022 - 41st Annual International Conference on the Theory and Applications of Cryptographic Techniques, Trondheim, Norway, May 30 - June 3, 2022, Proceedings, Part I*, ser. Lecture Notes in Computer Science, O. Dunkelman and S. Dziembowski, Eds., vol. 13275. Springer, 2022, pp. 581–610. [Online]. Available: https://doi.org/10.1007/978-3-031-06944-4_20

[332] H. Chen, M. Kim, I. P. Razenshteyn, D. Rotaru, Y. Song, and S. Wagh, "Maliciously secure matrix multiplication with applications to private deep learning," in *Advances in Cryptology - ASIACRYPT 2020 - 26th International Conference on the Theory and Application of Cryptology and Information Security, Daejeon, South Korea, December 7-11, 2020, Proceedings, Part III*, ser. Lecture Notes in Computer Science, S. Moriai and H. Wang, Eds., vol. 12493. Springer, 2020, pp. 31–59. [Online]. Available: https://doi.org/10.1007/978-3-030-64840-4_2

[333] S. Bian, D. Kundi, K. Hirozawa, W. Liu, and T. Sato, "APAS: application-specific accelerators for RLWE-based homomorphic linear transformations," *IEEE Trans. Inf. Forensics Secur.*, vol. 16, pp. 4663–4678, 2021. [Online]. Available: https://doi.org/10.1109/TIFS.2021.3114032

[334] X. Wang, S. Ranellucci, and J. Katz, "Global-scale secure multiparty computation," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, CCS 2017, Dallas, TX, USA, October 30 - November 03, 2017*, B. Thuraisingham, D. Evans, T. Malkin, and D. Xu, Eds. ACM, 2017, pp. 39–56. [Online]. Available: https://doi.org/10.1145/3133956.3133979

[335] C. Hazay, P. Scholl, and E. Soria-Vazquez, "Low cost constant round MPC combining BMR and oblivious transfer," *J. Cryptol.*, vol. 33, no. 4, pp. 1732–1786, 2020. [Online]. Available: https://doi.org/10.1007/s00145-020-09355-y

[336] A. Ben-Efraim, Y. Lindell, and E. Omri, "Efficient scalable constant-round MPC via garbled circuits," in *Advances in Cryptology - ASIACRYPT 2017 - 23rd International Conference on the Theory and Applications of Cryptology and Information Security, Hong Kong, China, December 3-7, 2017, Proceedings, Part II*, ser. Lecture Notes in Computer Science, T. Takagi and T. Peyrin, Eds., vol. 10625. Springer, 2017, pp. 471–498. [Online]. Available: https://doi.org/10.1007/978-3-319-70697-9_17

[337] C. Hazay, E. Orsini, P. Scholl, and E. Soria-Vazquez, "Tinykeys: A new approach to efficient multi-party computation," *J. Cryptol.*, vol. 35, no. 2, p. 13, 2022. [Online]. Available: https://doi.org/10.1007/s00145-022-09423-5

[338] A. Ben-Efraim, K. Cong, E. Omri, E. Orsini, N. P. Smart, and E. Soria-Vazquez, "Large scale, actively secure computation from LPN and free-xor garbled circuits," in *Advances in Cryptology - EUROCRYPT 2021 - 40th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, October 17-21, 2021, Proceedings, Part III*, ser. Lecture Notes in Computer Science, A. Canteaut and F. Standaert, Eds., vol. 12698. Springer, 2021, pp. 33–63. [Online]. Available: https://doi.org/10.1007/978-3-030-77883-5_2

[339] M. Abadi, A. Chu, I. J. 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, Vienna, Austria, October 24-28, 2016*, E. R. Weippl, S. Katzenbeisser, C. Kruegel, A. C. Myers, and S. Halevi, Eds. ACM, 2016, pp. 308–318. [Online]. Available: https://doi.org/10.1145/2976749.2978318

[340] G. Cormode, S. Jha, T. Kulkarni, N. Li, D. Srivastava, and T. Wang, "Privacy at scale: Local differential privacy in practice," in *Proceedings of the 2018 International Conference on Management of Data, SIGMOD Conference 2018, Houston, TX, USA, June 10-15, 2018*, G. Das, C. M. Jermaine, and P. A. Bernstein, Eds. ACM, 2018, pp. 1655–1658. [Online]. Available: https://doi.org/10.1145/3183713.3197390

[341] K. Wei, J. Li, M. Ding, C. Ma, H. H. Yang, F. Farokhi, S. Jin, T. Q. S. Quek, and H. V. Poor, "Federated learning with differential privacy: Algorithms and performance analysis," *IEEE Trans. Inf. Forensics Secur.*, vol. 15, pp. 3454–3469, 2020. [Online]. Available: https://doi.org/10.1109/TIFS.2020.2988575

[342] K. A. Bonawitz, V. Ivanov, B. Kreuter, A. Marcedone, H. B. McMahan, S. Patel, D. Ramage, A. Segal, and K. Seth, "Practical secure aggregation for privacy-preserving machine learning," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, CCS 2017, Dallas, TX, USA, October 30 - November 03, 2017*, B. Thuraisingham, D. Evans, T. Malkin, and D. Xu, Eds. ACM, 2017, pp. 1175–1191. [Online]. Available: https://doi.org/10.1145/3133956.3133982

[343] J. H. Bell, K. A. Bonawitz, A. Gascón, T. Lepoint, and M. Raykova, "Secure single-server aggregation with (poly)logarithmic overhead," in *CCS '20: 2020 ACM SIGSAC Conference on Computer and Communications Security, Virtual Event, USA, November 9-13, 2020*, J. Ligatti, X. Ou, J. Katz, and G. Vigna, Eds. ACM, 2020, pp. 1253–1269. [Online]. Available: https://doi.org/10.1145/3372297.3417885

[344] T. D. Nguyen, P. Rieger, H. Chen, H. Yalame, H. Möllering, H. Fereidooni, S. Marchal, M. Miettinen, A. Mirhoseini, S. Zeitouni, F. Koushanfar, A. Sadeghi, and T. Schneider, "FLAME: taming backdoors in federated learning," in *31st USENIX Security Symposium, USENIX Security 2022, Boston, MA, USA, August 10-12, 2022*, K. R. B. Butler and K. Thomas, Eds. USENIX Association, 2022, pp. 1415–1432. [Online]. Available: https://www.usenix.org/conference/usenixsecurity22/presentation/nguyen

[345] M. Rathee, C. Shen, S. Wagh, and R. A. Popa, "ELSA: secure aggregation for federated learning with malicious actors," *IACR Cryptol. ePrint Arch. (To appear at IEEE S& P 2023.)*, p. 1695, 2022. [Online]. Available: https://eprint.iacr.org/2022/1695

[346] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, AISTATS 2017, 20-22 April 2017, Fort Lauderdale, FL, USA*, ser. Proceedings of Machine Learning Research, A. Singh and X. J. Zhu, Eds., vol. 54. PMLR, 2017, pp. 1273–1282. [Online]. Available: http://proceedings.mlr.press/v54/mcmahan17a.html

[347] Y. Ben-Itzhak, H. Möllering, B. Pinkas, T. Schneider, A. Suresh, O. Tkachenko, S. Vargaftik, C. Weinert, H. Yalame, and A. Yanai, "Scionfl: Secure quantized aggregation for federated learning," *CoRR*, vol. abs/2210.07376, 2022. [Online]. Available: https://doi.org/10.48550/arXiv.2210.07376

[348] H. Fereidooni, S. Marchal, M. Miettinen, A. Mirhoseini, H. Möllering, T. D. Nguyen, P. Rieger, A. Sadeghi, T. Schneider, H. Yalame, and S. Zeitouni, "Safelearn: Secure aggregation for private federated learning," in *IEEE Security and Privacy Workshops, SP Workshops 2021, San Francisco, CA, USA, May 27, 2021*. IEEE, 2021, pp. 56–62. [Online]. Available: https://doi.org/10.1109/SPW53761.2021.00017

[349] ——, "SafeFL: MPC-friendly framework for private and robust federated learning," in *Deep Learning Security and Privacy Workshop 2023*, 2023. [Online]. Available: https://eprint.iacr.org/2023/555

[350] F. Boenisch, A. Dziedzic, R. Schuster, A. S. Shamsabadi, I. Shumailov, and N. Papernot, "When the curious abandon honesty: Federated learning is not private," *CoRR*, vol. abs/2112.02918, 2021. [Online]. Available: https://arxiv.org/abs/2112.02918

[351] Y. Wen, J. Geiping, L. Fowl, M. Goldblum, and T. Goldstein, "Fishing for user data in large-batch federated learning via gradient magnification," in *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, ser. Proceedings of Machine Learning Research, K. Chaudhuri, S. Jegelka, L. Song, C. Szepesvári, G. Niu, and S. Sabato, Eds., vol. 162. PMLR, 2022, pp. 23 668–23 684. [Online]. Available: https://proceedings.mlr.press/v162/wen22a.html

[352] R. Canetti, "Universally composable security: A new paradigm for cryptographic protocols," in *42nd Annual Symposium on Foundations of Computer Science, FOCS 2001, 14-17 October 2001, Las Vegas, Nevada, USA*. IEEE Computer Society, 2001, pp. 136–145. [Online]. Available: https://doi.org/10.1109/SFCS.2001.959888

[353] 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.

[354] D. Philpott, *A guide to federal terms and acronyms*. Bernan Press, 2017.

[355] M. Hernandez, G. Epelde, A. Alberdi, R. Cilla, and D. Rankin, "Synthetic data generation for tabular health records: A systematic review," *Neurocomputing*, 2022.

[356] C. Yan, Y. Yan, Z. Wan, Z. Zhang, L. Omberg, J. Guinney, S. D. Mooney, and B. A. Malin, "A multifaceted benchmarking of synthetic electronic health record generation models," *Nature Communications*, vol. 13, no. 1, p. 7609, 2022.

[357] P. W.N. and C. I.G, "Privacy in the age of medical big data." 2019.

[358] S. Bakas, K. Farahani, M. G. Linguraru, U. Anazodo, C. Carr, A. Flanders, L. M. Prevedello, F. C. Kitamura, J. Kalpathy-Cramer, J. Mongan, U. Baid, E. Calabrese, J. D. Rudie, E. Colak, Z. Jiang, X. Liu, J. Eddy, T. Bergquist, T. Yu, V. Chung, R. T. Shinohara, A. F. Kazerooni, and B. Menze, "The Brain Tumor Segmentation Challenge (2022 Continuous Updates & Generalizability Assessment)," Mar. 2022. [Online]. Available: https://doi.org/10.5281/zenodo.6362180

[359] R. Petersen, P. Aisen, L. Beckett, M. Donohue, A. Gamst, D. Harvey, and C. Jack et al., "Alzheimer's Disease Neuroimaging Initiative (ADNI): clinical characterization," 2010. [Online]. Available: https://dx.doi.org/10.1212/WNL.0b013e3181cb3e25

[360] R. Dorent, A. Kujawa, M. Ivory, S. Bakas, N. Rieke, S. Joutard, B. Glocker, J. Cardoso, M. Modat, K. Batmanghelich, A. Belkov, M. B. Calisto, J. W. Choi, B. M. Dawant, H. Dong, S. Escalera, Y. Fan, L. Hansen, M. P. Heinrich, S. Joshi, V. Kashtanova, H. G. Kim, S. Kondo, C. N. Kruse, S. K. Lai-Yuen, H. Li, H. Liu, B. Ly, I. Oguz, H. Shin, B. Shirokikh, Z. Su, G. Wang, J. Wu, Y. Xu, K. Yao, L. Zhang, S. Ourselin, J. Shapey, and T. Vercauteren, "CrossMoDA 2021 challenge: Benchmark of cross-modality domain adaptation techniques for vestibular schwannoma and cochlea segmentation," *Medical Image Analysis*, vol. 83, p. 102628, 2023. [Online]. Available: https://doi.org/10.10162Fj.media.2022.102628

[361] T. Nyholm, S. Svensson, S. Andersson, J. Jonsson, M. Sohlin, C. Gustavsson, E. Kjellen, P. Albertsson, L. Blomqvist, B. Zackrisson, L. E. Olsson, and A. Gunnlaugsson, "Mr and ct data with multi observer delineations of organs in the pelvic area - part of the gold atlas project," 2018. [Online]. Available: https://doi.org/10.5281/zenodo.583096

[362] F. Knoll, J. Zbontar, A. Sriram, M. J. Muckley, M. Bruno, A. Defazio, M. Parente, K. J. Geras, J. Katsnelson, H. Chandarana, Z. Zhang, M. Drozdzalv, A. Romero, M. Rabbat, P. Vincent, J. Pinkerton, D. Wang, N. Yakubova, E. Owens, C. L. Zitnick, M. P. Recht, D. K. Sodickson, and Y. W. Lui, "fastmri: A publicly available raw k-space and dicom dataset of knee images for accelerated mr image reconstruction using machine learning," *Radiology: Artificial Intelligence*, vol. 2, no. 1, p. e190007, 2020, pMID: 32076662. [Online]. Available: https://doi.org/10.1148/ryai.2020190007

[363] X. Wang, Y.Peng, L. Lu, Z. Lu, M. Bagheri, and R. Summers, "Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases." *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[364] E. Sogancioglu, K. Murphy, and B. van Ginneken, "Node21," Apr. 2021. [Online]. Available: https://doi.org/10.5281/zenodo.4725881

[365] J. Shiraishi, S. Katsuragawa, J. Ikezoe, T. Matsumoto, T. Kobayashi, K. Komatsu, M. Matsui, H. Fujita, Y. Kodera, and K. Doi, "Development of a digital image database for chest radiographs with and without a lung nodule: receiver operating characteristic analysis of radiologists' detection of pulmonary nodules." *American Journal of Roentgenology*, 2000.

[366] A. Bustos, A. Pertusa, J. Salinas, and M. de la Iglesia-Vaya, "PadChest: A large chest x-ray image dataset with multi-label annotated reports," *Medical Image Analysis*, 2020.

[367] D. Demner-Fushman, S. Antani, M. Simpson, and G. Thoma, "Design and development of a multimodal biomedical information retrieval system." *Journal of Computing Science and Engineering*, 2012.

[368] J. Irvin, P. Rajpurkar, M. Ko, Y. Yu, S. Ciurea-Ilcus, C. Chute, H. Marklund, B. Haghgoo, R. Ball, K. Shpanskaya, J. Seekins, D. A. Mong, S. S. Halabi, J. K. Sandberg, R. Jones, D. B. Larson, C. P. Langlotz, B. N. Patel, M. P. Lungren, and A. Y. Ng, "Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison," 2019.

[369] I. C. Moreira et al., "Inbreast: Toward a full-field digital mammographic database," *Academic Radiology*, vol. 19, pp. 236–248, 2011.

[370] M. D. Halling-Brown, L. M. Warren, D. Ward, E. Lewis, A. Mackenzie, M. G. Wallis, L. Wilkinson, R. M. Given-Wilson, R. McAvinchey, and K. C. Young, "Optimam mammography image database: a large scale resource of mammography images and clinical data," 2020.

[371] M. A. G. Lopez, N. Posada, D. C. Moura, R. R. Pollán, M. G.-V. Jose, F. S. Valiente, C. S. Ortega, M. R. del Solar, G. D. Herrero, A. IsabelM., P. Ramos, J. Loureiro, T. C. Fernandes, and B. M. F. de Araújo, "Bcdr : A breast cancer digital repository," 2012.

[372] "A curated mammography data set for use in computer-aided detection and diagnosis research," *Sci Data*, 2017.

[373] "A multi-million mammography image dataset and population-based screening cohort for the training and evaluation of deep neural networks—the cohort of screen-aged women (csaw)," *J Digit Imaging*, vol. 33, 2020.

[374] H. Fu, F. Li, J. I. Orlando, H. Bogunović, X. Sun, J. Liao, Y. Xu, S. Zhang, and X. Zhang, "Refuge: Retinal fundus glaucoma challenge," 2019. [Online]. Available: https://dx.doi.org/10.21227/tz6e-r977

[375] J. Staal, M. D. Abramoff, M. Niemeijer, M. A. Viergever, and B. Van Ginneken, "Ridge-based vessel segmentation in color images of the retina," *IEEE Transactions on Medical Imaging*, vol. 23, no. 4, pp. 501–509, 2004.

[376] A. D. Hoover, V. Kouznetsova, and M. Goldbaum, "Locating blood vessels in retinal images by piecewise threshold probing of a matched filter response," *IEEE Transactions on Medical Imaging*, vol. 19, no. 3, pp. 203–210, March 2000.

[377] V. Rotemberg, N. Kurtansky, B. Betz-Stablein, L. Caffery, E. Chousakos, N. Codella, and M. Combalia et al., "A patient-centric dataset of images and metadata for identifying melanomas using clinical context." *Sci Data*, vol. 8, no. 34, 2021. [Online]. Available: https://doi.org/10.1038/s41597-021-00815-z

[378] T. Mendonca, P. M. Ferreira, and J. Marques et al., "Ph2 - a dermoscopic image database for research and benchmarking." *35th International Conference of the IEEE Engineering in Medicine and Biology Society*, 2013.

[379] "Hyperkvasir, a comprehensive multi-class image and video dataset for gastrointestinal endoscopy," *Springer Nature Scientific Data*, vol. 7, 2020.

[380] M. Antonelli, A. Reinke, and S. Bakas et al., "The medical segmentation decathlon," *Nature Communications*, vol. 13, p. 4128, 2022.

[381] C. Sudlow, J. Gallacher, N. Allen, and V. Beral, et al., "UK biobank: an open access resource for identifying the causes of a wide range of complex diseases of middle and old age." *PLoS medicine*, 2015.

[382] G. Shmueli and K. Lichtendahl, "Practical time series forecasting with r," *Axelrod Schnall Publishers*, 2016.

[383] T. J. Pollard, A. E. W. Johnson, J. D. Raffa, L. A. Celi, R. G. Mark, and O. Badawi, "The eicu collaborative research database, a freely available multi-center database for critical care research," *Scientific Data*, vol. 5, no. 1, p. 180178, Sep 2018. [Online]. Available: https://doi.org/10.1038/sdata.2018.178

[384] T. Pollard, A. Johnson, J. Raffa, L. A. Celi, O. Badawi, and R. Mark, "eICU Collaborative Research Database (version 2.0)," 2019. [Online]. Available: https://doi.org/10.13026/C2WM1R

[385] A. Goldberger, L. Amaral, L. Glass, J. Hausdorff, P. C. Ivanov, R. . Mark, and H. E. Stanley, "Physiobank, physiotoolkit, and physionet: Components of a new research resource for complex physiologic signals. circulation [online]. 101 (23), pp. e215–e220." 2000.

[386] A. Johnson, T. Pollard, L. Shen, L.-w. Lehman, M. Feng, M. Ghassemi, B. Moody, P. Szolovits, L. Celi, and R. Mark, "Mimic-iii, a freely accessible critical care database," *Scientific Data*, vol. 3, p. 160035, 05 2016.

[387] A. Johnson, T. Pollard, and R. Mark, "Mimic-iii clinical database (version 1.4)," 2016. [Online]. Available: https://doi.org/10.13026/C2XW26

[388] D. Dua and C. Graff, "UCI machine learning repository," 2017. [Online]. Available: http://archive.ics.uci.edu/ml

[389] G. Moody and R. Mark, "The impact of the mit-bih arrhythmia database," *IEEE Engineering in Medicine and Biology Magazine*, vol. 20, no. 3, pp. 45–50, 2001.

[390] B. Blankertz, G. Dornhege, M. Krauledat, K.-R. Müller, and G. Curio, "The non-invasive berlin brain–computer interface: Fast acquisition of effective performance in untrained subjects," *NeuroImage*, vol. 37, no. 2, pp. 539–550, 2007. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1053811907000535

[391] K. Miller and G. Schalk, "Prediction of finger flexion 4th brain-computer interface data competition," 01 2008.

[392] R. Leeb and C. Brunner, "Bci competition 2008 graz data set b," 2008.

[393] C. Brunner1, R. Leeb, G. R. Müller-Putz, A. Schlögl, and G. Pfurtscheller, "Bci competition 2008 graz data set a," 2008.

[394] B. Kemp, A. Zwinderman, B. Tuk, H. Kamphuisen, and J. Oberye, "Analysis of a sleep-dependent neuronal feedback loop: the slow-wave microcontinuity of the eeg," *IEEE Transactions on Biomedical Engineering*, vol. 47, no. 9, pp. 1185–1194, 2000.

[395] P. Detti, G. Vatti, and G. Zabalo Manrique de Lara, "Eeg synchronization analysis for seizure prediction: A study on data of noninvasive recordings," *Processes*, vol. 8, no. 7, 2020. [Online]. Available: https://www.mdpi.com/2227-9717/8/7/846

[396] P. Detti, "Siena scalp eeg database (version 1.0.0)," 2020. [Online]. Available: https://doi.org/10.13026/5d4a-j060

[397] H. A. Dau, E. Keogh, K. Kamgar, C.-C. M. Yeh, Y. Zhu, S. Gharghabi, C. A. Ratanamahatana, Yanping, B. Hu, N. Begum, A. Bagnall, A. Mueen, G. Batista, and Hexagon-ML, "The ucr time series classification archive," October 2018, https://www.cs.ucr.edu/~eamonn/time_series_data_2018/.

[398] M. V. Olson, "The human genome project." *Proceedings of the National Academy of Sciences*, vol. 90, no. 10, pp. 4338–4344, 1993.

[399] M. Naveed, E. Ayday, E. W. Clayton, J. Fellay, C. A. Gunter, J.-P. Hubaux, B. A. Malin, and X. Wang, "Privacy in the genomic era," *ACM Computing Surveys (CSUR)*, vol. 48, no. 1, pp. 1–44, 2015.

[400] Z. Lin, A. B. Owen, and R. B. Altman, "Genomic research and human subject privacy," pp. 183–183, 2004.

[401] S. S. Shringarpure and C. D. Bustamante, "Privacy risks from genomic data-sharing beacons," *The American Journal of Human Genetics*, vol. 97, no. 5, pp. 631–646, 2015.

[402] C. Ziegenhain and R. Sandberg, "Bamboozle removes genetic variation from human sequence data for open data sharing," *Nature Communications*, vol. 12, no. 1, p. 6216, 2021.

[403] A. Bernier, H. Liu, and B. M. Knoppers, "Computational tools for genomic data de-identification: facilitating data protection law compliance," *Nature Communications*, vol. 12, no. 1, p. 6949, 2021.

[404] B. Oprisanu, G. Ganev, and E. De Cristofaro, "On utility and privacy in synthetic genomic data," *arXiv preprint arXiv:2102.03314*, 2021.

[405] D. Bujold, D. A. de Lima Morais, C. Gauthier, C. Côté, M. Caron, T. Kwan, K. C. Chen, J. Laperle, A. N. Markovits, T. Pastinen *et al.*, "The international human epigenome consortium data portal," *Cell systems*, vol. 3, no. 5, pp. 496–499, 2016.

[406] M. Kim, J. Yun, Y. Cho, K. Shin, R. Jang, H. Bae, and N. Kim, "Deep learning in medical imaging," in *Neurospine*, vol. 16, 2019, pp. 471–472.

[407] X. Yi, E. Walia, and P. Babyn, "Generative adversarial network in medical imaging: A review," *Medical Image Analysis*, vol. 58, p. 101552, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1361841518308430

[408] A. Goncalves, P. Ray, B. Soper, J. Stevens, L. Coyle, and A. P. Sales, "Generation and evaluation of synthetic patient data," *BMC Medical Research Methodology*, vol. 20, no. 1, p. 108, May 2020. [Online]. Available: https://doi.org/10.1186/s12874-020-00977-1

[409] N. K. Singh and K. Raza, *Medical Image Generation Using Generative Adversarial Networks: A Review*. Singapore: Springer Singapore, 2021, pp. 77–96. [Online]. Available: https://doi.org/10.1007/978-981-15-9735-0_5

[410] T. Wang, Y. Lei, Y. Fu, J. F. Wynne, W. J. Curran, T. Liu, and X. Yang, "A review on medical imaging synthesis using deep learning and its clinical applications," *Journal of Applied Clinical Medical Physics*, vol. 22, no. 1, pp. 11–36, 2021. [Online]. Available: https://aapm.onlinelibrary.wiley.com/doi/abs/10.1002/acm2.13121

[411] A. Kebaili, J. Lapuyade-Lahorgue, and S. Ruan, "Deep learning approaches for data augmentation in medical imaging: A review," *Journal of Imaging*, vol. 9, no. 4, 2023. [Online]. Available: https://www.mdpi.com/2313-433X/9/4/81

[412] K. D. P. and W. Max, "Auto-Encoding Variational Bayes," in *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014.

[413] G. Ian, P.-A. Jean, M. Mehdi, X. Bing, W.-F. David, O. Sherjil, C. Aaron, and B. Yoshua, "Generative adversarial nets," in *Advances in Neural Information Processing Systems*, Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Weinberger, Eds., vol. 27. Curran Associates, Inc., 2014. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2014/file/5ca3e9b122f61f8f06494c97b1afccf3-Paper.pdf

[414] H. Jonathan, J. Ajay, and A. Pieter, "Denoising diffusion probabilistic models," in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., vol. 33. Curran Associates, Inc., 2020, pp. 6840–6851. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2020/file/4c5bcfec8584af0d967f1ab10179ca4b-Paper.pdf

[415] R. Olaf, F. Philipp, and B. Thomas, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, N. Nassir, H. Joachim, W. W. M., and F. A. F., Eds. Cham: Springer International Publishing, 2015, pp. 234–241.

[416] V. Ashish, S. Noam, P. Niki, U. Jakob, J. Llion, G. A. N., K. Łukasz, and P. Illia, "Attention is all you need," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, ser. NIPS'17. Red Hook, NY, USA: Curran Associates Inc., 2017, p. 6000–6010.

[417] F. Shamshad, S. Khan, S. W. Zamir, M. H. Khan, M. Hayat, F. S. Khan, and H. Fu, "Transformers in medical imaging: A survey," *Medical Image Analysis*, p. 102802, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1361841523000634

[418] P. Kingma Durk, T. Salimans, R. Jozefowicz, X. Chen, I. Sutskever, and M. Welling, "Improved variational inference with inverse autoregressive flow," in *Advances in Neural Information Processing Systems*, D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, Eds., vol. 29. Curran Associates, Inc., 2016. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2016/file/ddeebdeefdb7e7e7a697e1c3e3d8ef54-Paper.pdf

[419] S. Zhao, J. Song, and S. Ermon, "Infovae: Information maximizing variational autoencoders," *CoRR*, vol. abs/1706.02262, 2017. [Online]. Available: http://arxiv.org/abs/1706.02262

[420] A. Razavi, A. van den Oord, and O. Vinyals, *Generating Diverse High-Fidelity Images with VQ-VAE-2*. Red Hook, NY, USA: Curran Associates Inc., 2019.

[421] S. Kihyuk, L. Honglak, and Y. Xinchen, "Learning structured output representation using deep conditional generative models," in *Advances in Neural Information Processing Systems*, C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, Eds., vol. 28. Curran Associates, Inc., 2015. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2015/file/8d55a249e6baa5c06772297520da2051-Paper.pdf

[422] A. B. L. Larsen, S. K. Sønderby, H. Larochelle, and O. Winther, "Autoencoding beyond pixels using a learned similarity metric," in *Proceedings of The 33rd International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, B. M. Florina and W. K. Q., Eds., vol. 48. New York, New York, USA: PMLR, 20–22 Jun 2016, pp. 1558–1566. [Online]. Available: https://proceedings.mlr.press/v48/larsen16.html

[423] L. Junzhao and C. Junying, "Data augmentation of thyroid ultrasound images using generative adversarial network," in *2021 IEEE International Ultrasonics Symposium (IUS)*, 2021, pp. 1–4.

[424] A. Hirte, M. Platscher, T. Joyce, J. Heit, E. Tranvinh, and C. Federau, "Realistic generation of diffusion-weighted magnetic resonance brain images with deep generative models," in *Magn Reson Imaging*, vol. 81, 2021, pp. 60–66.

[425] M. Pesteie, P. Abolmaesumi, and R. N. Rohling, "Adaptive augmentation of medical data using independently conditional variational auto-encoders," *IEEE Transactions on Medical Imaging*, vol. 38, no. 12, pp. 2807–2820, 2019.

[426] Z. Peiye, S. A. G., and K. Oluwasanmi, "Fmri data augmentation via synthesis," in *2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019)*, 2019, pp. 1783–1787.

[427] C. Chadebec, E. Thibeau-Sutre, N. Burgos, and S. Allassonnière, "Data augmentation in high dimensional low sample size setting using a geometry-based variational autoencoder," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 3, pp. 2879–2896, 2023.

[428] C. Chadebec and S. Allassonnière, "Data augmentation with variational autoencoders and manifold sampling," in *Deep Generative Models, and Data Augmentation, Labelling, and Imperfections*, S. Engelhardt, I. Oksuz, D. Zhu, Y. Yuan, A. Mukhopadhyay, N. Heller, S. X. Huang, H. Nguyen, R. Sznitman, and Y. Xue, Eds. Cham: Springer International Publishing, 2021, pp. 184–192.

[429] M. Beetz, A. Banerjee, Y. Sang, and V. Grau, "Combined generation of electrocardiogram and cardiac anatomy models using multi-modal variational autoencoders," in *2022 IEEE 19th International Symposium on Biomedical Imaging (ISBI)*, 2022, pp. 1–4.

[430] M. Gan and C. Wang, "Esophageal optical coherence tomography image synthesis using an adversarially learned variational autoencoder," in *Biomed Opt Express*, vol. 13, no. 3, 2022, pp. 1188–1201.

[431] J. Sundgaard, M. Hannemose, S. Laugesen, P. Bray, J. Harte, Y. Kamide, C. Tanaka, R. Paulsen, and A. Christensen, "Multi-modal data generation with a deep metric variational autoencoder," *Proceedings of the Northern Lights Deep Learning Workshop*, vol. 4, 01 2023.

[432] L. Mescheder, S. Nowozin, and A. Geiger, "Which training methods for gans do actually converge?" in *International Conference on Machine Learning (ICML)*, 2018.

[433] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," in *Proceedings of the 34th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, P. Doina and T. Y. Whye, Eds., vol. 70. PMLR, 06–11 Aug 2017, pp. 214–223. [Online]. Available: https://proceedings.mlr.press/v70/arjovsky17a.html

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

[435] I. Phillip, Z. Jun-Yan, Z. Tinghui, and E. A. A., "Image-to-image translation with conditional adversarial networks," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 5967–5976.

[436] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," in *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2016. [Online]. Available: http://arxiv.org/abs/1511.06434

[437] T. Karras, T. Aila, S. Laine, and J. Lehtinen, "Progressive growing of GANs for improved quality, stability, and variation," in *International Conference on Learning Representations*, 2018. [Online]. Available: https://openreview.net/forum?id=Hk99zCeAb

[438] Z. Jun-Yan, P. Taesung, I. Phillip, and E. A. A, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Computer Vision (ICCV), 2017 IEEE International Conference on*, 2017.

[439] A. Odena, C. Olah, and J. Shlens, "Conditional image synthesis with auxiliary classifier gans," in *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ser. ICML'17. JMLR.org, 2017, p. 2642–2651.

[440] F. Calimeri, A. Marzullo, C. Stamile, and G. Terracina, "Biomedical data augmentation using generative adversarial neural networks," in *Artificial Neural Networks and Machine Learning – ICANN 2017*, A. Lintas, S. Rovetta, P. F. Verschure, and A. E. Villa, Eds. Cham: Springer International Publishing, 2017, pp. 626–634.

[441] L. Zhang, A. Gooya, and A. F. Frangi, "Semi-supervised assessment of incomplete lv coverage in cardiac mri using generative adversarial nets," in *Simulation and Synthesis in Medical Imaging*, S. A. Tsaftaris, A. Gooya, A. F. Frangi, and J. L. Prince, Eds. Cham: Springer International Publishing, 2017, pp. 61–68.

[442] C. Han, H. Hayashi, L. Rundo, R. Araki, W. Shimoda, S. Muramatsu, Y. Furukawa, G. Mauri, and H. Nakayama, "Gan-based synthetic brain mr image generation," in *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, 2018, pp. 734–738.

[443] C. Bermudez, A. Plassard, T. Davis, A. Newton, S. Resnick, and B. Landman, "Learning Implicit Brain MRI Manifolds with Deep Learning," in *Proceedings of SPIE–the International Society for Optical Engineering*, vol. 10574, 105741L, 2018.

[444] A. K. Mondal, J. Dolz, and C. Desrosiers, "Few-shot 3d multi-modal medical image segmentation using generative adversarial learning," 2018.

[445] C. Bowles, L. Chen, R. Guerrero, P. Bentley, R. Gunn, A. Hammers, D. A. Dickie, M. V. Hernández, J. Wardlaw, and D. Rueckert, "Gan augmentation: Augmenting training data using generative adversarial networks," 2018.

[446] D. Korkinof, T. Rijken, M. O'Neill, J. Yearsley, H. Harvey, and B. Glocker, "High-resolution mammogram synthesis using progressive generative adversarial networks," *arXiv preprint arXiv:1807.03401*, 2018.

[447] M. Frid-Adar, I. Diamant, E. Klang, M. Amitai, J. Goldberger, and H. Greenspan, "Gan-based synthetic medical image augmentation for increased cnn performance in liver lesion classification," *Neurocomputing*, vol. 321, pp. 321–331, 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0925231218310749

[448] V. Sandfort, K. Yan, and P. Pickhardt et al., "Data augmentation using generative adversarial networks (CycleGAN) to improve generalizability in CT segmentation tasks," *Sci Rep*, 2019.

[449] H.-C. Shin, N. A. Tenenholtz, J. K. Rogers, C. G. Schwarz, M. L. Senjem, J. L. Gunter, K. P. Andriole, and M. Michalski, "Medical image synthesis for data augmentation and anonymization using generative adversarial networks," in *Simulation and Synthesis in Medical Imaging*, A. Gooya, O. Goksel, I. Oguz, and N. Burgos, Eds. Cham: Springer International Publishing, 2018, pp. 1–11.

[450] H. Salehinejad, S. Valaee, T. Dowdell, E. Colak, and J. Barfett, "Generalization of deep neural networks for chest pathology classification in x-rays using generative adversarial networks," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 990–994.

[451] A. Madani, M. Moradi, A. Karargyris, and T. Syeda-Mahmood, "Semi-supervised learning with generative adversarial networks for chest x-ray classification with ability of data domain adaptation," in *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, 2018, pp. 1038–1042.

[452] ——, "Chest x-ray generation and data augmentation for cardiovascular abnormality classification," in *Medical Imaging 2018: Image Processing*, E. D. Angelini and B. A. Landman, Eds., vol. 10574, International Society for Optics and Photonics. SPIE, 2018, p. 105741M. [Online]. Available: https://doi.org/10.1117/12.2293971

[453] B. Hu, Y. Tang, E. I.-C. Chang, Y. Fan, M. Lai, and Y. Xu, "Unsupervised learning for cell-level visual representation in histopathology images with generative adversarial networks," *IEEE Journal of Biomedical and Health Informatics*, vol. 23, no. 3, pp. 1316–1328, 2019.

[454] T. Virdi, J. Guibas, and P. Li, "Synthetic medical images from dual generative adversarial networks," 09 2017.

[455] C. Baur, S. Albarqouni, and N. Navab, "Melanogans: High resolution skin lesion synthesis with gans," *ArXiv*, vol. abs/1804.04338, 2018.

[456] X. Yi, E. Walia, and P. S. Babyn, "Unsupervised and semi-supervised learning with categorical generative adversarial networks assisted by wasserstein distance for dermoscopy image classification," *ArXiv*, vol. abs/1804.03700, 2018.

[457] R. Rassin, S. Ravfogel, and Y. Goldberg, "Dalle-2 is seeing double: Flaws in word-to-concept mapping in text2image models," 2022.

[458] C. Saharia, W. Chan, S. Saxena, L. Li, J. Whang, E. Denton, S. K. S. Ghasemipour, B. K. Ayan, S. S. Mahdavi, R. G. Lopes, T. Salimans, J. Ho, D. J. Fleet, and M. Norouzi, "Photorealistic text-to-image diffusion models with deep language understanding," 2022.

[459] N. Carlini, J. Hayes, M. Nasr, M. Jagielski, V. Sehwag, F. Tramèr, B. Balle, D. Ippolito, and E. Wallace, "Extracting training data from diffusion models," *CoRR*, vol. abs/2301.13188, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2301.13188

[460] W. H. L. Pinaya, P.-D. Tudosiu, J. Dafflon, P. F. Da Costa, V. Fernandez, P. Nachev, S. Ourselin, and M. J. Cardoso, "Brain imaging generation with latent diffusion models," in *Deep Generative Models*, A. Mukhopadhyay, I. Oksuz, S. Engelhardt, D. Zhu, and Y. Yuan, Eds. Cham: Springer Nature Switzerland, 2022, pp. 117–126.

[461] J. Wolleb, R. Sandkühler, F. Bieder, and P. C. Cattin, "The swiss army knife for image-to-image translation: Multi-task diffusion models," 2022.

[462] V. Fernandez, W. H. L. Pinaya, P. Borges, P.-D. Tudosiu, M. S. Graham, T. Vercauteren, and M. J. Cardoso, "Can segmentation models be trained with fully synthetically generated data?" in *Simulation and Synthesis in Medical Imaging*, C. Zhao, D. Svoboda, J. M. Wolterink, and M. Escobar, Eds. Cham: Springer International Publishing, 2022, pp. 79–90.

[463] F. Khader, G. Müller-Franzes, and S. Tayebi Arasteh et al., "Denoising diffusion probabilistic models for 3d medical image generation," 2023.

[464] K. Packhäuser, L. Folle, F. Thamm, and A. Maier, "Generation of anonymous chest radiographs using latent diffusion models for training thoracic abnormality classification systems," 2022.

[465] P. Moghadam, S. V. Dalen, K. C. Martin, J. Lennerz, S. Yip, H. Farahani, and A. Bashashati, "A morphology focused diffusion probabilistic model for synthesis of histopathology images," in *2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. Los Alamitos, CA, USA: IEEE Computer Society, jan 2023, pp. 1999–2008. [Online]. Available: https://doi.ieeecomputersociety.org/10.1109/WACV56688.2023.00204

[466] L. W. Sagers, J. A. Diao, M. Groh, P. Rajpurkar, A. S. Adamson, and A. K. Manrai, "Improving dermatology classifiers across populations using images generated by large diffusion models," 2022.

[467] E. Lashgari, D. Liang, and U. Maoz, "Data augmentation for deep-learning-based electroencephalography," *Journal of Neuroscience Methods*, vol. 346, p. 108885, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0165027020303083

[468] B. Liu, Z. Zhang, and R. Cui, "Efficient time series augmentation methods," in *2020 13th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, 2020, pp. 1004–1009.

[469] B. K. Iwana and S. Uchida, "An empirical survey of data augmentation for time series classification with neural networks," *PLOS ONE*, vol. 16, no. 7, pp. 1–32, 07 2021. [Online]. Available: https://doi.org/10.1371/journal.pone.0254841

[470] Q. Wen, L. Sun, X. Song, J. Gao, X. Wang, and H. Xu, "Time series data augmentation for deep learning: A survey," *CoRR*, vol. abs/2002.12478, 2020. [Online]. Available: https://arxiv.org/abs/2002.12478

[471] E. Brophy, Z. Wang, Q. She, and T. Ward, "Generative adversarial networks in time series: A survey and taxonomy," *CoRR*, vol. abs/2107.11098, 2021. [Online]. Available: https://arxiv.org/abs/2107.11098

[472] A. Le Guennec, S. Malinowski, and R. Tavenard, "Data Augmentation for Time Series Classification using Convolutional Neural Networks," in *ECML/PKDD Workshop on Advanced Analytics and Learning on Temporal Data*, Riva Del Garda, Italy, Sep. 2016. [Online]. Available: https://shs.hal.science/halshs-01357973

[473] B. K. Iwana and S. Uchida, "Time series data augmentation for neural networks by time warping with a discriminative teacher," *CoRR*, vol. abs/2004.08780, 2020. [Online]. Available: https://arxiv.org/abs/2004.08780

[474] C. Esteban, S. L. Hyland, and G. Rätsch, "Real-valued (medical) time series generation with recurrent conditional gans," 2017.

[475] K. G. Hartmann, R. T. Schirrmeister, and T. Ball, "Eeg-gan: Generative adversarial networks for electroencephalograhic (eeg) brain signals," 2018.

[476] B. Zhou, S. Liu, B. Hooi, X. Cheng, and J. Ye, "Beatgan: Anomalous rhythm detection using adversarially generated time series," in *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, ser. IJCAI'19. AAAI Press, 2019, p. 4433–4439.

[477] A. M. Delaney, E. Brophy, and T. E. Ward, "Synthesis of realistic ecg using generative adversarial networks," 2019.

[478] D. Hazra and Y.-C. Byun, "Synsiggan: Generative adversarial networks for synthetic biomedical signal generation," *Biology*, vol. 9, no. 12, 2020. [Online]. Available: https://www.mdpi.com/2079-7737/9/12/441

[479] F. Zhu, F. Ye, Y. Fu, Q. Liu, and B. Shen, "Electrocardiogram generation with a bidirectional lstm-cnn generative adversarial network," *Scientific Reports*, vol. 9, no. 1, p. 6734, May 2019. [Online]. Available: https://doi.org/10.1038/s41598-019-42516-z

[480] K. Bhanot, S. Dash, J. Pedersen, I. Guyon, and K. Bennett, "Quantifying resemblance of synthetic medical time-series," in *ESANN 2021 proceedings*. Ciaco - i6doc.com, 2021. [Online]. Available: https://doi.org/10.14428%2Fesann%2F2021.es2021-108

[481] S. S. Samani, Z. Huang, E. Ayday, M. Elliot, J. Fellay, J.-P. Hubaux, and Z. Kutalik, "Quantifying genomic privacy via inference attack with high-order snv correlations," in *2015 IEEE Security and Privacy Workshops*. IEEE, 2015, pp. 32–40.

[482] B. Yelmen, A. Decelle, L. Ongaro, D. Marnetto, C. Tallec, F. Montinaro, C. Furtlehner, L. Pagani, and F. Jay, "Creating artificial human genomes using generative neural networks," *PLoS genetics*, vol. 17, no. 2, p. e1009303, 2021.

[483] N. Killoran, L. J. Lee, A. Delong, D. Duvenaud, and B. J. Frey, "Generating and designing dna with deep generative models," *arXiv preprint arXiv:1712.06148*, 2017.

[484] B. Yelmen, A. Decelle, L. L. Boulos, A. Szatkownik, C. Furtlehner, G. Charpiat, and F. Jay, "Deep convolutional and conditional neural networks for large-scale genomic data generation," *bioRxiv*, pp. 2023–03, 2023.

[485] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville, "Improved training of wasserstein gans," *Advances in neural information processing systems*, vol. 30, 2017.

[486] 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.

[487] MONAI Consortium, "MONAI: Medical Open Network for AI," Nov. 2021, Zenodo, version 0.8.0. [Online]. Available: https://doi.org/10.5281/zenodo.5728262

[488] B. Billot, D. N. Greve, O. Puonti, A. Thielscher, K. Van Leemput, B. Fischl, A. V. Dalca, and J. E. Iglesias, "Synthseg: Segmentation of brain MRI scans of any contrast and resolution without retraining," *Medical Image Analysis*, vol. 86, p. 102789, 2023.

[489] E. Gibson, W. Li, C. Sudre, L. Fidon, D. I. Shakir, G. Wang, Z. Eaton-Rosen, R. Gray, T. Doel, Y. Hu, T. Whyntie, P. Nachev, M. Modat, D. C. Barratt, S. Ourselin, M. J. Cardoso, and T. Vercauteren, "Niftynet: a deep-learning platform for medical imaging," *Computer Methods and Programs in Biomedicine*, 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0169260717311823

[490] F. Isensee, J. Petersen, A. Klein, D. Zimmerer, P. F. Jaeger, S. Kohl, J. Wasserthal, G. Köhler, T. Norajitra, S. J. Wirkert, and K. H. Maier-Hein, "nnu-net: Self-adapting framework for u-net-based medical image segmentation," *CoRR*, vol. abs/1809.10486, 2018. [Online]. Available: http://arxiv.org/abs/1809.10486

[491] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, X. Chen, and X. Chen, "Improved techniques for training gans," in *Advances in Neural Information Processing Systems*, D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, Eds., vol. 29. Curran Associates, Inc., 2016. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2016/file/8a3363abe792db2d8761d6403605aeb7-Paper.pdf

[492] 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*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., vol. 30. Curran Associates, Inc., 2017. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2017/file/8a1d694707eb0fefe65871369074926d-Paper.pdf

[493] L. N. Vaserstein, "Markov processes over denumerable products of spaces, describing large systems of automata," *Probl. Peredachi Inf.*, vol. 5, no. 3, pp. 64–72, 1969.

[494] X. Nguyen, M. J. Wainwright, and M. Jordan, "Estimating divergence functionals and the likelihood ratio by penalized convex risk minimization," in *Advances in Neural Information Processing Systems*, J. Platt, D. Koller, Y. Singer, and S. Roweis, Eds., vol. 20. Curran Associates, Inc., 2007. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2007/file/72da7fd6d1302c0a159f6436d01e9eb0-Paper.pdf

[495] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *Computer Vision – ECCV 2016*, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds. Cham: Springer International Publishing, 2016, pp. 694–711.

[496] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.

[497] A. C. Bovik, "A visual information fidelity approach to video quality assessment," 2005.

[498] Z. Wang and A. Bovik, "A universal image quality index," *IEEE Signal Processing Letters*, vol. 9, no. 3, pp. 81–84, 2002.

[499] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Los Alamitos, CA, USA: IEEE Computer Society, jun 2018, pp. 586–595. [Online]. Available: https://doi.ieeecomputersociety.org/10.1109/CVPR.2018.00068

[500] V. L. Guen and N. Thome, "Deep time series forecasting with shape and temporal criteria," 2022.

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

[502] I. E. Olatunji, J. Rauch, M. Katzensteiner, and M. Khosla, "A review of anonymization for healthcare data," *Big Data*, 2022, pMID: 35271377. [Online]. Available: https://doi.org/10.1089/big.2021.0169

[503] European Parliament and Council of the European Union, General data protection regulation, Official Journal of the European Union, 2016. [Online]. Available: https://eur-lex.europa.eu/legal-content/EN/ALL/?uri=CELEX:32016R0679

[504] Opinion 05/2014 on Anonymisation Techniques, https://ec.europa.eu/justice/article-29/documentation/opinion-recommendation/files/2014/wp216_en.pdf.

[505] 10 Misunderstandings Related to Anonymisation, Agencia Española de Protección de Datos. 2021. [Online]. Available: https://www.aepd.es/es/documento/10-anonymisation-misunderstandings.pdf

[506] T. Carvalho and N. Moniz, "The compromise of data privacy in predictive performance," in *Advances in Intelligent Data Analysis XIX - 19th International Symposium on Intelligent Data Analysis, IDA 2021, Porto, Portugal, April 26-28, 2021, Proceedings*, ser. Lecture Notes in Computer Science, P. H. Abreu, P. P. Rodrigues, A. Fernández, and J. Gama, Eds., vol. 12695. Springer, 2021, pp. 426–438. [Online]. Available: https://doi.org/10.1007/978-3-030-74251-5_34

[507] T. Carvalho, N. Moniz, P. Faria, and L. Antunes, "Towards a data privacy-predictive performance trade-off," *Expert Syst. Appl.*, vol. 223, p. 119785, 2023. [Online]. Available: https://doi.org/10.1016/j.eswa.2023.119785

[508] ——, "Survey on privacy-preserving techniques for data publishing," *CoRR*, vol. abs/2201.08120, 2022. [Online]. Available: https://arxiv.org/abs/2201.08120

[509] B. C. Fung, K. Wang, A. W.-C. Fu, and P. S. Yu, *Introduction to Privacy-Preserving Data Publishing: Concepts and Techniques*, 1st ed. Chapman & Hall/CRC, 2010.

[510] R. Somolinos, A. Muñoz, M. E. Hernando, M. Pascual Carrasco, R. Madariaga, O. Moreno Gil, F. López-Rodríguez, and C. Salvador, "Pseudonimización de información clínica para uso secundario. aplicación en un caso práctico iso/en 13606," in *Congreso Anual de la Sociedad Española de Ingeniería Biomédica. CASEIB 2014.*, 11 2014.

[511] J. Domingo-Ferrer, D. Sánchez, and J. Soria-Comas, *Database Anonymization: Privacy Models, Data Utility, and Microaggregation-based Inter-model Connections*, ser. Synthesis Lectures on Information Security, Privacy, & Trust. Morgan & Claypool Publishers, 2016. [Online]. Available: https://doi.org/10.2200/S00690ED1V01Y201512SPT015

[512] 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

[513] P. Samarati, "Protecting respondents' identities in microdata release," *IEEE Trans. Knowl. Data Eng.*, vol. 13, no. 6, pp. 1010–1027, 2001. [Online]. Available: https://doi.org/10.1109/69.971193

[514] A. Majeed and S. Lee, "Anonymization techniques for privacy preserving data publishing: A comprehensive survey," *IEEE Access*, vol. 9, pp. 8512–8545, 2021. [Online]. Available: https://doi.org/10.1109/ACCESS.2020.3045700

[515] 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

[516] H. Zhu, S. Tian, and K. Lü, "Privacy-preserving data publication with features of independent ℓ-diversity," *Comput. J.*, vol. 58, no. 4, pp. 549–571, 2015. [Online]. Available: https://doi.org/10.1093/comjnl/bxu102

[517] 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

[518] C. Dwork, "Differential privacy: A survey of results," in *Theory and Applications of Models of Computation, 5th International Conference, TAMC 2008, Xi'an, China, April 25-29, 2008. Proceedings*, ser. Lecture Notes in Computer Science, M. Agrawal, D. Du, Z. Duan, and A. Li, Eds., vol. 4978. Springer, 2008, pp. 1–19. [Online]. Available: https://doi.org/10.1007/978-3-540-79228-4_1

[519] A. Blanco-Justicia, D. Sánchez, J. Domingo-Ferrer, and K. Muralidhar, "A critical review on the use (and misuse) of differential privacy in machine learning," *ACM Comput. Surv.*, vol. 55, no. 8, pp. 160:1–160:16, 2023. [Online]. Available: https://doi.org/10.1145/3547139

[520] A. H. Landberg, K. Nguyen, E. Pardede, and J. W. Rahayu, "δ-dependency for privacy-preserving XML data publishing," *J. Biomed. Informatics*, vol. 50, pp. 77–94, 2014. [Online]. Available: https://doi.org/10.1016/j.jbi.2014.01.013

[521] J. F. Marques and J. Bernardino, "Analysis of data anonymization techniques," in *Proceedings of the 12th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management, IC3K 2020, Volume 2: KEOD, Budapest, Hungary, November 2-4, 2020*, D. Aveiro, J. L. G. Dietz, and J. Filipe, Eds. SCITEPRESS, 2020, pp. 235–241. [Online]. Available: https://doi.org/10.5220/0010142302350241

[522] T. Koberg and K. Stark, "Measuring information reduction caused by anonymization methods in neps scientific use files," *NEPS Working Paper*, no. 65, 2016.

[523] J. Soria-Comas, J. Domingo-Ferrer, D. Sánchez, and S. Martínez, "t-closeness through microaggregation: Strict privacy with enhanced utility preservation," *IEEE Trans. Knowl. Data Eng.*, vol. 27, no. 11, pp. 3098–3110, 2015. [Online]. Available: https://doi.org/10.1109/TKDE.2015.2435777

[524] G. Loukides and J. Shao, "Data utility and privacy protection trade-off in k-anonymisation," in *Proceedings of the 2008 International Workshop on Privacy and Anonymity in Information Society, PAIS 2008, Nantes, France, March 29, 2008*, ser. ACM International Conference Proceeding Series, F. Fotouhi, L. Xiong, and T. M. Truta, Eds. ACM, 2008, pp. 36–45. [Online]. Available: https://doi.org/10.1145/1379287.1379296

[525] M. Cunha, R. Mendes, and J. P. Vilela, "A survey of privacy-preserving mechanisms for heterogeneous data types," *Computer Science Review*, vol. 41, p. 100403, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1574013721000435

[526] E. Zheleva and L. Getoor, "Preserving the privacy of sensitive relationships in graph data," in *Privacy, Security, and Trust in KDD*, F. Bonchi, E. Ferrari, B. Malin, and Y. Saygin, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 153–171.

[527] S. D. C. di Vimercati, S. Foresti, G. Livraga, and P. Samarati, "k-anonymity: From theory to applications," *Trans. Data Priv.*, vol. 16, no. 1, pp. 25–49, 2023. [Online]. Available: http://www.tdp.cat/issues21/abs.a460a22.php

[528] J. Soria-Comas, J. Domingo-Ferrer, D. Sánchez, and S. Martínez, "Enhancing data utility in differential privacy via microaggregation-based $$k$$-anonymity," *The VLDB Journal*, vol. 23, no. 5, pp. 771–794, Oct. 2014. [Online]. Available: https://doi.org/10.1007/s00778-014-0351-4

[529] A. Yamamoto, E. Kimura, and T. Shibuya, "(ϵ, k)-randomized anonymization: ϵ-differentially private data sharing with k-anonymity," in *Proceedings of the 16th International Joint Conference on Biomedical Engineering Systems and Technologies, BIOSTEC 2023, Volume 5: HEALTHINF, Lisbon, Portugal, February 16-18, 2023*, F. Cabitza, A. L. N. Fred, and H. Gamboa, Eds. SCITEPRESS, 2023, pp. 287–297. [Online]. Available: https://doi.org/10.5220/0011665600003414

[530] B. Su, J. Huang, K. Miao, Z. Wang, X. Zhang, and Y. Chen, "K-anonymity privacy protection algorithm for multi-dimensional data against skewness and similarity attacks," *Sensors*, vol. 23, no. 3, p. 1554, 2023. [Online]. Available: https://doi.org/10.3390/s23031554

[531] D. D. Pascale, G. Cascavilla, D. A. Tamburri, and W. van den Heuvel, "Real-world k-anonymity applications: The kgen approach and its evaluation in fraudulent transactions," *Inf. Syst.*, vol. 115, p. 102193, 2023. [Online]. Available: https://doi.org/10.1016/j.is.2023.102193

[532] J. Andrew, J. Eunice, and J. Karthikeyan, "An anonymization-based privacy-preserving data collection protocol for digital health data," *Front. Public Health*, 2023. [Online]. Available: https://doi.org/10.3389/fpubh.2023.1125011

[533] A. Narayanan and V. Shmatikov, "Robust de-anonymization of large sparse datasets," in *2008 IEEE Symposium on Security and Privacy (S&P 2008), 18-21 May 2008, Oakland, California, USA*. IEEE Computer Society, 2008, pp. 111–125. [Online]. Available: https://doi.org/10.1109/SP.2008.33

[534] M. Barbaro and T. J. Zeller, "A face is exposed for aol searcher no. 4417749," *The New York Times*. [Online]. Available: https://www.nytimes.com/2006/08/09/technology/09aol.html

[535] Y.-A. De Montjoye, C. A. Hidalgo, M. Verleysen, and V. D. Blondel, "Unique in the crowd: The privacy bounds of human mobility," *Scientific reports*, vol. 3, p. 1376, 2013. [Online]. Available: http://dx.doi.org/10.1038/srep01376

[536] Y.-A. de Montjoye, L. Radaelli, V. K. Singh, and A. S. Pentland, "Unique in the shopping mall: On the reidentifiability of credit card metadata," *Science*, vol. 347, no. 6221, pp. 536–539, 2015. [Online]. Available: https://www.science.org/doi/abs/10.1126/science.1256297

[537] K. El Emam, E. Jonker, L. Arbuckle, and B. Malin, "A systematic review of re-identification attacks on health data," *PloS one*, vol. 6, no. 12, p. e28071, 2011.

[538] V. Ravindra and A. Grama, "De-anonymization attacks on neuroimaging datasets," in *SIGMOD '21: International Conference on Management of Data, Virtual Event, China, June 20-25, 2021*, G. Li, Z. Li, S. Idreos, and D. Srivastava, Eds. ACM, 2021, pp. 2394–2398. [Online]. Available: https://doi.org/10.1145/3448016.3457234

[539] P. Kosseim and K. E. Emam, "Privacy interests in prescription data, part I: prescriber privacy," *IEEE Secur. Priv.*, vol. 7, no. 1, pp. 72–76, 2009. [Online]. Available: https://doi.org/10.1109/MSP.2009.14

[540] K. E. Emam and P. Kosseim, "Privacy interests in prescription data, part 2: Patient privacy," *IEEE Secur. Priv.*, vol. 7, no. 2, pp. 75–78, 2009. [Online]. Available: https://doi.org/10.1109/MSP.2009.47

[541] S. Ji, Q. Gu, H. Weng, Q. Liu, P. Zhou, J. Chen, Z. Li, R. Beyah, and T. Wang, "De-health: All your online health information are belong to us," in *36th IEEE International Conference on Data Engineering, ICDE 2020, Dallas, TX, USA, April 20-24, 2020*. IEEE, 2020, pp. 1609–1620. [Online]. Available: https://doi.org/10.1109/ICDE48307.2020.00143

[542] H. Hu, Z. Salcic, L. Sun, G. Dobbie, P. S. Yu, and X. Zhang, "Membership inference attacks on machine learning: A survey," *ACM Comput. Surv.*, vol. 54, no. 11s, pp. 235:1–235:37, 2022. [Online]. Available: https://doi.org/10.1145/3523273

[543] M. Wu, X. Zhang, J. Ding, H. V. Nguyen, R. Yu, M. Pan, and S. T. Wong, "Evaluation of inference attack models for deep learning on medical data," *CoRR*, vol. abs/2011.00177, 2020. [Online]. Available: https://arxiv.org/abs/2011.00177

[544] N. Z. Gong and B. Liu, "You are who you know and how you behave: Attribute inference attacks via users' social friends and behaviors," in *25th USENIX Security Symposium, USENIX Security 16, Austin, TX, USA, August 10-12, 2016*, T. Holz and S. Savage, Eds. USENIX Association, 2016, pp. 979–995. [Online]. Available: https://www.usenix.org/conference/usenixsecurity16/technical-sessions/presentation/gong

[545] ——, "Attribute inference attacks in online social networks," *ACM Trans. Priv. Secur.*, vol. 21, no. 1, pp. 3:1–3:30, 2018. [Online]. Available: https://doi.org/10.1145/3154793

[546] P. P. Tricomi, L. Facciolo, G. Apruzzese, and M. Conti, "Attribute inference attacks in online multiplayer video games: A case study on DOTA2," in *Proceedings of the Thirteenth ACM Conference on Data and Application Security and Privacy, CODASPY 2023, Charlotte, NC, USA, April 24-26, 2023*, M. Shehab, M. Fernández, and N. Li, Eds. ACM, 2023, pp. 27–38. [Online]. Available: https://doi.org/10.1145/3577923.3583653

[547] A. Antoniou, G. Dossena, J. MacMillan, S. Hamblin, D. Clifton, and P. Petrone, "Assessing the risk of re-identification arising from an attack on anonymised data," *CoRR*, vol. abs/2203.16921, 2022. [Online]. Available: https://doi.org/10.48550/arXiv.2203.16921

[548] A. C. Haber, U. Sax, and F. Prasser, "Open tools for quantitative anonymization of tabular phenotype data: literature review," *Briefings Bioinform.*, vol. 23, no. 6, 2022. [Online]. Available: https://doi.org/10.1093/bib/bbac440

[549] Amnesia - High accuracy Data Anonymization. Available online: https://amnesia.openaire.eu/, accessed on 19 May 2023.

[550] ARX - Data Anonymization Tool: A Comprehensive Software for Privacy-Preserving Microdata Publishing, 2022, Available online: https://arx.deidentifier.org, accessed on 18 May 2023.

[551] Anonimatron, The free, extendable, open source data anonymization tool, Available online: https://realrolfje.github.io/anonimatron/, accessed on 22 May 2023.

[552] R. C. Carreras, J. C. P. Baún, and M. V. Jiménez, "Cybersec4europe - securing and preserving privacy sharing health data," *ERCIM News*, no. 133, 2023.

[553] F. Kohlmayer, F. Prasser, C. Eckert, A. Kemper, and K. A. Kuhn, "Highly efficient optimal k-anonymity for biomedical datasets," in *Proceedings of CBMS 2012, The 25th IEEE International Symposium on Computer-Based Medical Systems, June 20-22, 2012, Rome, Italy*, P. Soda, F. Tortorella, S. K. Antani, M. Pechenizkiy, M. Cannataro, and A. Tsymbal, Eds. IEEE Computer Society, 2012, pp. 1–6. [Online]. Available: https://doi.org/10.1109/CBMS.2012.6266366

[554] CyberSecurity for Europe project, D5.6 Validation of Demonstration Case Phase 2, 2022. https://cybersec4europe.eu/wp-content/uploads/2022/12/D5.6-Validation-Demonstration-Case-Phase-2-Final-submitted.pdf, accessed on 22 May 2023.

[555] D. Sánchez, S. Martínez, J. Domingo-Ferrer, J. Soria-Comas, and M. Batet, "μ-ant: semantic microaggregation-based anonymization tool," *Bioinform.*, vol. 36, no. 5, pp. 1652–1653, 2020. [Online]. Available: https://doi.org/10.1093/bioinformatics/btz792

[556] A. Bampoulidis, I. Markopoulos, and M. Lupu, "Prioprivacy: A local recoding k-anonymity tool for prioritised quasi-identifiers," in *IEEE/WIC/ACM International Conference on Web Intelligence - Companion Volume*, ser. WI '19 Companion. New York, NY, USA: Association for Computing Machinery, 2019, p. 314–317. [Online]. Available: https://doi.org/10.1145/3358695.3360918

[557] SdcTools, Joinup https://joinup.ec.europa.eu/solution/sdctools-toolsstatistical-disclosure-control/about, accessed on 19 May 2023.

[558] Statistical Disclosure Control-μ-ARGUS, https://research.cbs.nl/casc/mu.htm, accessed on 19 May 2023.

[559] Statistical Disclosure Control-τ-ARGUS, https://research.cbs.nl/casc/tau.htm, accessed on 19 May 2023.

[560] Statistical Disclosure Control Methods for Anonymization of Data and Risk Estimation. Package 'sdcMicro', 2023, https://cran.r-project.org/web/packages/sdcMicro/sdcMicro.pdf, accessed on 22 May 2023.

[561] S. Pitoglou, A. Filntisi, A. Anastasiou, G. K. Matsopoulos, and D. Koutsouris, "Measuring the impact of anonymization on real-world consolidated health datasets engineered for secondary research use: Experiments in the context of modelhealth project," *Frontiers in Digital Health*, vol. 4, 2022. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fdgth.2022.841853

[562] ISO/IEC 27559:2022 (en), Information security, cybersecurity and privacy protection – Privacy enhancing data de-identification framework. [Online]. Available: https://www.iso.org/obp/ui/#iso:std:iso-iec:27559:ed-1:v1:en(accessed on 29 May 2023).

[563] ISO/IEC 20889:2018(en), Privacy enhancing data de-identification terminology and classification of techniques. [Online]. Available: https://www.iso.org/obp/ui/#iso:std:iso-iec:20889:ed-1:v1:en (accessed on 29 May 2023).

[564] S. N. Eshun and P. Palmieri, "Two de-anonymization attacks on real-world location data based on a hidden markov model," in *IEEE European Symposium on Security and Privacy, EuroS&P 2022 - Workshops, Genoa, Italy, June 6-10, 2022*. IEEE, 2022, pp. 1–9. [Online]. Available: https://doi.org/10.1109/EuroSPW55150.2022.00062

[565] J. F. Marques. and J. Bernardino., "Analysis of data anonymization techniques," in *Proceedings of the 12th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K 2020) - KEOD*, INSTICC. SciTePress, 2020, pp. 235–241.

[566] O. Vovk, G. Piho, and P. Ross, "Anonymization methods of structured health care data: A literature review," in *Model and Data Engineering*, C. Attiogbé and S. Ben Yahia, Eds. Cham: Springer International Publishing, 2021, pp. 175–189.

[567] P. Lison, I. Pilán, D. Sánchez, M. Batet, and L. Øvrelid, "Anonymisation models for text data: State of the art, challenges and future directions," in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, C. Zong, F. Xia, W. Li, and R. Navigli, Eds. Association for Computational Linguistics, 2021, pp. 4188–4203. [Online]. Available: https://doi.org/10.18653/v1/2021.acl-long.323

[568] W. R. Stahel, "Sustainability and services," in *Sustainable solutions*. Routledge, 2017, pp. 151–164.

[569] K. N. Lemon and P. C. Verhoef, "Understanding customer experience throughout the customer journey," *Journal of marketing*, vol. 80, no. 6, pp. 69–96, 2016.

[570] A. Følstad and K. Kvale, "Customer journeys: a systematic literature review," *Journal of Service Theory and Practice*, 2018.

[571] T.-K. Kuuru and E. Närvänen, "Embodied interaction in customer experience: a phenomenological study of group fitness," *Journal of Marketing Management*, vol. 35, no. 13-14, pp. 1241–1266, 2019.

[572] R. Jain, J. Aagja, and S. Bagdare, "Customer experience–a review and research agenda," *Journal of Service Theory and Practice*, vol. 27, no. 3, pp. 642–662, 2017.

[573] M. Bordian, I. Gil-Saura, and M. Šerić, "The impact of value co-creation in sustainable services: Understanding generational differences," *Journal of Services Marketing*, vol. 37, no. 2, pp. 155–167, 2023.

[574] Y.-A. Chen and C. L. Chen, "Case study of sustainable service design in the hospitality industry," *Chinese Management Studies*, vol. 16, no. 1, pp. 162–196, 2022.

[575] R. Halvorsrud, K. Kvale, and A. Følstad, "Improving service quality through customer journey analysis," *Journal of service theory and practice*, vol. 26, no. 6, pp. 840–867, 2016.

[576] P. J. Danaher and J. Mattsson, "Cumulative encounter satisfaction in the hotel conference process," *International Journal of Service Industry Management*, vol. 5, no. 4, pp. 69–80, 1994.

[577] L. Shostack, "Designing services that deliver," *Harvard Business Review*, vol. 62, no. 1, pp. 133–139, 1984. [Online]. Available: https://cir.nii.ac.jp/crid/1571698599245674880

[578] J. M. Sulek, M. R. Lind, and A. S. Marucheck, "The impact of a customer service intervention and facility design on firm performance," *Management Science*, vol. 41, no. 11, pp. 1763–1773, 1995.

[579] M. M. Tseng, M. Qinhai, and C.-J. Su, "Mapping customers' service experience for operations improvement," *Business Process Management Journal*, vol. 5, no. 1, pp. 50–64, 1999.

[580] B. Schneider and D. E. Bowen, "The service organization: Human resources management is crucial," *Organizational dynamics*, vol. 21, no. 4, pp. 39–52, 1993.

[581] A. Pantouvakis and A. Gerou, "The theoretical and practical evolution of customer journey and its significance in services sustainability," *Sustainability*, vol. 14, no. 15, p. 9610, 2022.

[582] J. J. Marquez, A. Downey, and R. Clement, "Walking a mile in the user's shoes: Customer journey mapping as a method to understanding the user experience," *Internet Reference Services Quarterly*, vol. 20, no. 3-4, pp. 135–150, 2015.

[583] C. Voss, A. V. Roth, and R. B. Chase, "Experience, service operations strategy, and services as destinations: foundations and exploratory investigation," *Production and operations management*, vol. 17, no. 3, pp. 247–266, 2008.

[584] D. W. Norton and B. J. Pine, "Using the customer journey to road test and refine the business model," *Strategy & Leadership*, vol. 41, no. 2, pp. 12–17, 2013.

[585] H. Majra, R. Saxena, S. Jha, and S. Jagannathan, "Structuring technology applications for enhanced customer experience: Evidence from indian air travellers," *Global business review*, vol. 17, no. 2, pp. 351–374, 2016.

[586] J. Rudkowski, C. Heney, H. Yu, S. Sedlezky, and F. Gunn, "Here today, gone tomorrow? mapping and modeling the pop-up retail customer journey," *Journal of Retailing and Consumer Services*, vol. 54, p. 101698, 2020.

[587] A.-M. Kranzbühler, M. H. Kleijnen, and P. W. Verlegh, "Outsourcing the pain, keeping the pleasure: effects of outsourced touchpoints in the customer journey," *Journal of the Academy of Marketing Science*, vol. 47, pp. 308–327, 2019.

[588] U. Gretzel, D. R. Fesenmaier, and J. T. O'Leary, "The transformation of consumer behaviour," in *Tourism business frontiers*. Routledge, 2006, pp. 9–18.

[589] D. Wang, Z. Xiang, and D. R. Fesenmaier, "Smartphone use in everyday life and travel," *Journal of travel research*, vol. 55, no. 1, pp. 52–63, 2016.

[590] A. Lockwood and P. Jones, "Creating positive service encounters," *Cornell Hotel and Restaurant Administration Quarterly*, vol. 29, no. 4, pp. 44–50, 1989.

[591] J. A. Czepiel, "Service encounters and service relationships: implications for research," *Journal of business research*, vol. 20, no. 1, pp. 13–21, 1990.

[592] B. Bosio, K. Rainer, and M. Stickdorn, "Customer experience research with mobile ethnography: A case study of the alpine destination serfaus-fiss-ladis," in *Qualitative consumer research*. Emerald Publishing Limited, 2017, vol. 14, pp. 111–137.

[593] C. Meyer, A. Schwager *et al.*, "Understanding customer experience," *Harvard business review*, vol. 85, no. 2, p. 116, 2007.

[594] M. Ieva and C. Ziliani, "The role of customer experience touchpoints in driving loyalty intentions in services," *The TQM Journal*, 2018.

[595] C. M. Voorhees, P. W. Fombelle, Y. Gregoire, S. Bone, A. Gustafsson, R. Sousa, and T. Walkowiak, "Service encounters, experiences and the customer journey: Defining the field and a call to expand our lens," *Journal of Business Research*, vol. 79, pp. 269–280, 2017.

[596] F. Ponsignon, F. Durrieu, and T. Bouzdine-Chameeva, "Customer experience design: a case study in the cultural sector," *Journal of Service Management*, vol. 28, no. 4, pp. 763–787, 2017.

[597] N. M. Puccinelli, R. C. Goodstein, D. Grewal, R. Price, P. Raghubir, and D. Stewart, "Customer experience management in retailing: understanding the buying process," *Journal of retailing*, vol. 85, no. 1, pp. 15–30, 2009.

[598] M. S. Rosenbaum, M. L. Otalora, and G. C. Ramírez, "How to create a realistic customer journey map," *Business horizons*, vol. 60, no. 1, pp. 143–150, 2017.

[599] A. Crosier and A. Handford, "Customer journey mapping as an advocacy tool for disabled people: A case study," *Social Marketing Quarterly*, vol. 18, no. 1, pp. 67–76, 2012.

[600] I. Jeong, J. Seo, J. LIM, J. Jang, and J. Kim, "Improvement of the business model of the disaster management system based on the service design methodology," *International Journal of Safety and Security Engineering*, vol. 6, no. 1, pp. 19–29, 2016.

[601] D. van Lierop, J. Eftekhari, A. O'Hara, and Y. Grinspun, "Humanizing transit data: connecting customer experience statistics to individuals' unique transit stories," *Transportation Research Record*, vol. 2673, no. 1, pp. 388–402, 2019.

[602] D. C. Edelman and M. Singer, "Competing on customer journeys," *Harvard business review*, vol. 93, no. 11, pp. 88–100, 2015.

[603] M. Muskat, B. Muskat, A. Zehrer, and R. Johns, "Generation y: evaluating services experiences through mobile ethnography," *Tourism Review*, vol. 68, no. 3, pp. 55–71, 2013.

[604] H. Moon, S. H. Han, J. Chun, and S. W. Hong, "A design process for a customer journey map: A case study on mobile services," *Human Factors and Ergonomics in Manufacturing & Service Industries*, vol. 26, no. 4, pp. 501–514, 2016.

[605] L. Demi, "Practical guide to ultrasound beam forming: Beam pattern and image reconstruction analysis," *Applied Sciences*, vol. 8, no. 9, 2018. [Online]. Available: https://www.mdpi.com/2076-3417/8/9/1544