D4.1

Initial toolset for robust, explainable, fair and privacy-preserving AI

Project Title	AI4Media – A European Excellence Centre for Media, Society and Democracy
Contract No.	951911
Instrument	Research and Innovation Action
Thematic Priority	H2020-EU.2.1.1. - INDUSTRIAL LEADERSHIP - Leadership in enabling and industrial technologies - Information and Communication Technologies (ICT) / ICT-48-2020 - Towards a vibrant European network of AI excellence centres
Start of Project	1 September 2020
Duration	48 months

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 951911

info@ai4media.eu www.ai4media.eu

Deliverable title	Initial toolset for robust, explainable, fair and privacy-preserving AI
Deliverable number	D4.1
Deliverable version	1.0
Previous version(s)	N/A
Contractual date of delivery	August 31st 2021
Actual date of delivery	August 23rd 2021
Deliverable filename	AI4Media Deliverable D4.1
Nature of deliverable	Report
Dissemination level	Public
Number of pages	62
Work Package	WP4
Task(s)	T4.2, T4.3, T4.4, T4.5
Parner responsible	IBM
Author(s)	Killian Levacher (IBM), Hervé Le Borgne (CEA), Thomas Köllmer, Patrick Aichroth (FhG-IDMT), Cigdem Beyan, Nicu Sebe (UNITN), Vasileios Mezaris (CERTH), Daniel Gatica-Perez, Sina Sajadmanesh (IDIAP), Vasileios Mygdalis (AUTH), Mara Graziani (HES-SO Valais)
Editor	Killian Levacher (IBM)
Officer	Evangelia Markidou

Abstract	This deliverable presents the initial work and outcomes from WP4 in AI4Media, focusing on Trustworthy AI. The document describes the initial investigations and results of our work targeting four dimensions of Trust namely, AI Robustness, Explainable AI, AI Privacy and AI Fairness, each respectively corresponding to tasks T4.2, T4.3, T4.4 and T4.5. In each section, we present an overview of each partner's contribution, the methodology used, along with initial results and relevant publications and software if available.
Keywords	AI Robustness, Explainable AI, AI Privacy, AI Fairness

Copyright

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 951911

Contributors

NAME	ORGANIZATION
Killian Levacher	IBM
Hervé Le Borgne	CEA
Vasileios Mezaris	CERTH
Vasileios Mygdalis	AUTH
Sina Sajadmanesh	IDIAP
Daniel Gatica-Perez	IDIAP
Mara Graziani	HES-SO
Thomas Köllmer	FhG-IDMT
Patrick Aichroth	FhG-IDMT
Cigdem Beyan	UNITN
Nicu Sebe	UNITN

Peer Reviewers

NAME	ORGANIZATION
Birgit Gray	DW
Giuseppe Amato	CNR

Revision History

Version	Date	Reviewer	Modifications
0.1	May 19th 2021	Killian Levacher	First draft sent to partners for contributions
0.2	July 13th 2021	Killian Levacher	Updated version with contributions from all partners
0.3	July 14th 2021	Filareti Tsalakanido	Updated version with review from Filareti Tsalakanido
0.4	July 17th 2021	Giuseppe Amato	Updated version with review from Giuseppe Amato
0.5	July 19th 2021	Birgit Gray	Updated version with review from Birgit Gray
0.6	August 8th 2021	Killian Levacher	Updated version with contributions from partners
1.0	August 12th 2021	Killian Levacher	Final version to be submitted

The information and views set out in this report are those of the author(s) and do not necessarily reflect the official opinion of the European Union. Neither the European Union institutions and bodies nor any person acting on their behalf.

Table of Abbreviations and Acronyms

Abbreviation	Meaning
AI	Artificial Intelligence
ART	Adversarial Robustness 360 Toolkit
AT	Adversarial Training
BB	Bounding Box
BIM	Basic Iterative Method
CE	Cross Entropy
CL	Center Loss
CNN	Convolutional Neural Network
CW-SSIM	Complex Wavelet Structural Similarity
DGM	Deep Generative Models
DoC	Degree of Confidence
DP	Differential Privacy
EU	European Union
ExpDist	Expected Distortion
FGSM	Fast Gradient Sign Method
FHE	Fully Homomorphic Encryption
FL	Federated Learning
GAN	Generative Adversarial Network
GAT	Graph Attention Network
GCN	Graph Convolutional Network
GDPR	General Data Protection Regulation
GNN	Graph Neural Network
HCP	Hypersperical Class Prototypes
LDP	Local Differential Privacy
LPGNN	Locally Private Graph Neural Network
LSTM	Long Short-Term Memory
MIM	Momentum Iterative Method
ML	Machine Learning
MOA	Modified Optimization Algorithm
OD	Object Detector
PCL	Prototype Conformity Loss
PET	Privacy Enhancement Technologies
RCV	Regression Concept Vector
re-ID	re-identification
ReD	REtraining with Distillation
ReX	REtraining with eXpansion
SM	Softmax function
SMPC	Secure Multiparty Computation

Abbreviation	Meaning
TarFid	Target Fidelity
WiD	Weighted in-Degree

Contents

List of Tables

List of Figures

1. Executive Summary

This deliverable presents the research carried out as part of tasks T4.2, T4.3, T4.4 and T4.5 of the AI4Media project covering the areas of AI Robustness, Explainability, Privacy and Fairness respectively. For each contribution, we provide an overview of the work carried out, as well as references to the publications and software released by each partner. The relevance of each technical output with respect to WP8 use cases is summarised along with our plans to integrate WP4 modules as part of the AI4EU catalogue.

This first deliverable presents the work carried out as part of WP4 in the first 12 months of the AI4Media project. It reflects the very good stage and research collaboration undertaken so far. As is demonstrated in this document, all participants have been very active and despite the early stage of this project, results have already been successfully published or submitted in top venues for each field (6 conference papers published, 1 journal paper submitted, 2 conference papers submitted). Joint collaboration between partners with respect to technical integration and public dissemination activities has already begun, with two workshops successfully organised. These early accomplishments represent a solid foundation to expand our research throughout the remaining years of the project.

Detail descriptions of ongoing research with very promising outlooks in each Trustworthy AI dimension is presented in this document. Specifically, research contributions within the dimension of AI Robustness include: (a) a novel method to increase neural network robustness using hyperspherical class prototypes, (b) a new set of backdoor attacks and defences addressing vulnerabilities in deep generative models, and (c) a new attack algorithm promoting the robustness of re-ID systems across unseen domains. These research outputs will benefit the "AI for Social Media and Against Disinformation", "AI for Social Sciences and Humanities" and "AI for (Re-)Organisation and Content Moderation" WP8 use cases by providing more robust training capabilities as well as novel attacks and defences.

Contributions within the dimension of Explainable AI include: (a) a new interpretability method for internal features of deep learning models, (b) a video event recognition method using graph convolutional networks, (c) a novel method for post-hoc explanation in decision-marking systems, and (d) a new technique enabling vector arithmetic capabilities within the latent space of generational models. A (e) workshop dedicated to developing a taxonomy of Explainable AI across various disciplines was also organised. WP8 use cases "AI for Social Media and Against Disinformation" and "AI for Social Sciences and Humanities" will benefit from these research outputs by acquiring novel interpretability capabilities for internal features of discriminative neural networks, generative models and object graph networks.

With respect to the AI privacy dimension of Trustworthy AI contributions include: (a) a new tool to secure privacy within graph neural networks, (b) a differential privacy library for AI models, (c) a description of techniques available for secure federated learning, and (d) a novel data protection method using a K-anonymity inspired adversarial attack. The WP8 use case "AI in Vision" will benefit directly from the novel data protection K-anonymity method while the various graph neural network and differential privacy mechanisms will be directly available for any model to use across WP8 use cases.

Finally, early stage contributions to the AI Fairness dimension of Trustworthy AI consist of an analysis of AI Fairness requirements across WP8 use cases, the identification of a suitable framework and existing gaps to address each of these needs as well as an initial set of workshop activities across project partners. The objective is to eventually benefit the ”AI for Social Media and Against Disinformation”, ”AI for Social Sciences and Humanities” and ”AI for (Re-)Organisation and Content Moderation” WP8 use cases by providing a set of bias detection and fairness enhancing algorithms adequate for the datasets and models used within the context of these WP8 use cases.

The goal in the following months will be to further increase the number of collaborations between partners across the various contributions presented in each Trustworthy AI dimension, consolidate and refine results already obtained in the areas of AI Explainability, Robustness and Privacy and expand early stage contributions in the dimension of AI Fairness.

2. Introduction

2.1. Trustworthy AI Overview

Artificial Intelligence (AI) is an area of strategic importance to the European Union with respect to its ability to support and shape future economic and social development. While the recent leaps in innovation in this space offer immense opportunities, due to the increasing importance and prevalence of AI systems across industries various aspects of this technology present many security as well as societal risks which may conflict with the ethical and democratic principles shared across the European Union (EU) such as transparency, privacy and inclusion among others.

Trustworthy AI hence aims at providing a framework for the development of Machine Learning (ML) technologies, which guarantees their suitability with respect to the democratic and ethical values shared in our society. This recently emerging field of AI can be typically divided within four broad dimensions, namely **AI robustness**, **Explainable AI**, **AI fairness** and **AI privacy**.

AI Robustness focuses on ML vulnerabilities which can be exploited by malicious attackers seeking to either steal capacities of proprietary models, identify private information used to train these models or purposely push a model in making incorrect predictions. These attacks can be achieved through the use of adversarial samples in various forms (images, texts, tabular data, etc.) and across a wide range of model types.

Explainable AI deals with the trust that needs to be established between an AI model and its user. As stated in Article 14 of the forthcoming EU AI Act legislation [1], *technical measures [must be] put in place to facilitate the interpretation of the outputs of AI systems by the users.* In other words, users of AI models must be able to understand why predictions were made, regardless of the precision or validity of each prediction. While the recent explosion of deep learning models has led to amazing gains in performance, these models in particular provide very limited visibility even to their own designers as to how they reached a decision. It is therefore crucial to develop a set of technologies which can support users in understanding how specific predictions were made, in order for these technologies to be safely incorporated within the fabric of society.

ML models are fast becoming incorporated into decision making processes shaping the lives of individual citizens on critical topics such as mortgage lending, prison sentencing etc. Since these models in part rely on statistical analysis of training datasets, they may inadvertently reproduce or reinforce unfair biases and prejudice already present in our society. *AI Fairness* hence aims at guaranteeing that any model prediction does not privilege a specific group/individual as well as situations/scenarios at the disadvantage of others.

Finally, the process of training and building AI models requires the management of large amounts of data which in many cases contains sensitive information which should not be shared beyond a dedicated group of data processors and owners. This generates a conflict of interest between the need to have the most numerous and accurate data available to reach high precision accuracy while at the same time reducing the amount of data being used to minimise any impact on individual's privacy. Private information leakage can occur both while a model is being trained as well as after deployment. *AI Privacy* hence aims at threading the needle between these two forces by providing the means to produce reliable ML models while simultaneously protecting individual's as well as corporations sensitive information.

[1] https://eur-lex.europa.eu/legal-content/EN/TXT/?qid=1623335154975&uri=CELEX%3A52021PC0206

2.2. WP4 Timeline

The work-package WP4 which is dedicated to Trustworthy AI involves 12 partner institutions (namely IBM, IDIAP, UPB, FhG, HES-SO, AUTH, CERTH, UCA, UNITN, CEA, KUL, UNIFI) and runs throughout the whole 48 months of the AI4Media project. A total of 110 Person Months (PMs) is dedicated to 6 tasks organised as 4 vertical tasks, i.e. Explainable AI (Task T4.3), Robust AI (Task T4.2), Fair AI (Task T4.5) and Private AI (Task 4.4), corresponding to the 4 dimensions of AI trustworthiness (see Section 2.1) and 2 horizontal tasks focusing on the benchmarking (Task 4.6) and legal dimensions of AI (Task 4.1) within the European Union (see Figure 2).

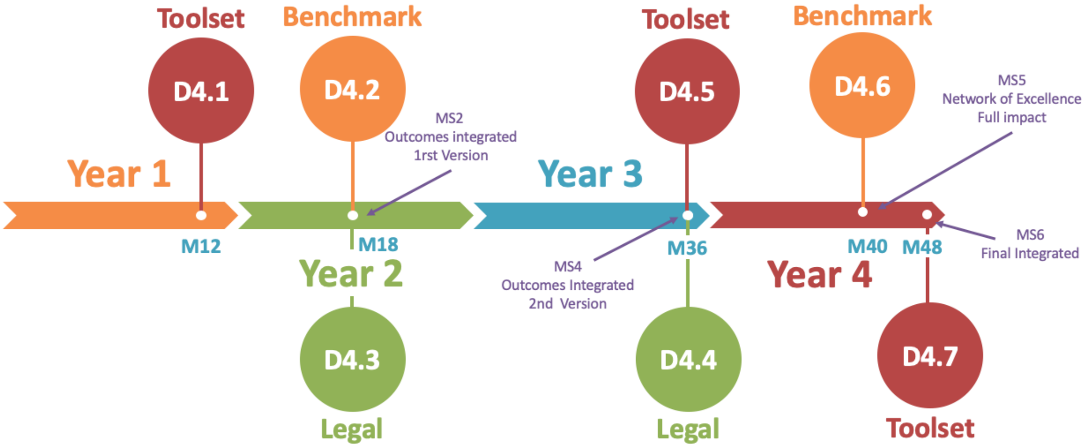


Figure 1. WP4 four-year Timeline

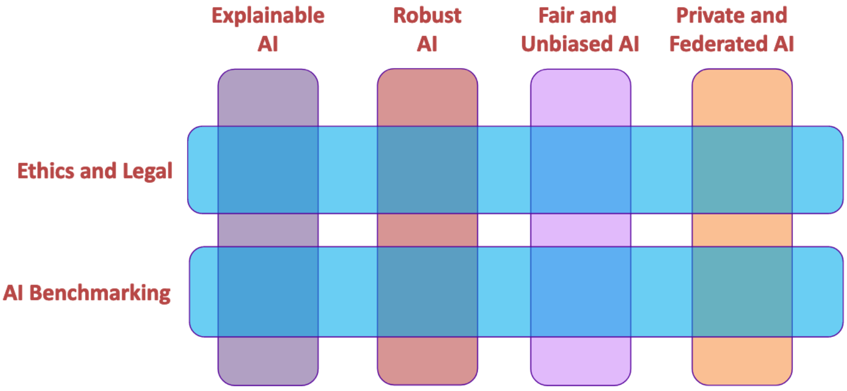


Figure 2. WP4 Tasks

As can be seen in Figure 1, during the course of the project this work-package will be producing 3 types of deliverables. Technical research outputs produced from the four vertical tasks will be reported together as part of the i) toolset deliverables while the 2 horizontal tasks will be reporting progress individually via the ii) Benchmark and iii) Legal WP4 deliverables. This document consists in the first initial iteration of the toolset deliverable. Each iteration of this deliverable will report on increasing levels of research maturity in each dimension of Trustworthy AI. As part of each iteration, partner contributions consist of a research pipeline starting with a set of experiments and research outputs. From these experiments, algorithms and methods are eventually consolidated as they mature into toolset modules reusable by the research community. In each iteration, we expect individual contributions to be at various stages of this pipeline as investigations mature.

In the first iteration of this deliverable, we present the initial technical requirement analysis carried out in each dimension, initial research outputs as well as workshop activities achieved during the first 12 months of the AI4Media project. The following two additional iterations of this deliverable will be submitted in months 36 and 48 respectively and will present updates on the progress achieved in each dimension once investigations have reached more significant levels of maturity.

At this initial stage of the project, we expect a large number of partners to base their initial contributions on existing work carried out prior to AI4Media. Whenever research contributions are based on existing research, partners in the relevant section will clarify which part was carried out prior to AI4Media. The first iteration of this deliverable focuses mostly on contributions undertaken within the dimensions of Explainable, Robustness AI and AI Privacy with preliminary contributions in the dimension of AI Fairness. We expect the latter dimension to represent a larger share of our contributions in subsequent deliverables as we shift and expand available resources accordingly.

2.3. Document Organisation

The rest of this document is organised as follows. The AI4Media contributions towards the development of more robust AI are presented in Section 3. They include the use of innovative attacks and defence mechanisms for generative and person-re-identification models as well as new training techniques for discriminative models.

Contributions towards the Explainable AI are then described in Section 4. These include tools to assess the importance of high-level concepts present within the internal features of deep learning models, novel video event recognition methods using graph based representation and analysis of events as well as tools that provide post-hoc explanations of decision-making systems based on collections of ML models.

Contributions towards AI Privacy are presented in Section 5. They include new methods which prevent information leakage within Graph Neural Network (GNN)s, a toolkit to perform privacy enhanced ML using differential privacy mechanism, a set of modules that allows the incorporation of selected Privacy Enhancement Technologies (PET) in federated learning frameworks and a novel adversarial attack methodology inspired by the K-anonymity principles, which can be employed for privacy protection against automated analysis tools.

The preliminary contributions towards AI Fairness are described in Section 6. They consist of an initial research requirement analysis and AI4Media-wide workshop activities performed in the first year of the project.

Section 7 presents how WP4 technical outputs will be integrated within each relevant WP8 use case and how each WP4 module will benefit individual WP8 Epics. The process by which technical requirements for each WP8 use case were retrieved is described along with the methodology used to link each WP4 technical output.

Finally, Section 8 provides a conclusion summarising the current progress achieved as part of WP4, including the various publications and workshops organised as part of this work-package. This section also provides more details related to our planned contribution to the larger AI4EU platform via the integration of WP4 technical outputs as AI4EU modules.

3. Robust AI Toolset

ML models are vulnerable to a variety of threat models [5, 6] in which adversarial samples play a critical role. Adversarial samples consist of inputs (images, texts, tabular data, etc.) deliberately crafted by an attacker in order to produce a desired response by the ML model unintended by the model creators.

There exists four broad types of adversarial threat models depending on how an attacker decides to exploit potential vulnerabilities in an ML model. Poisoning attacks focus on the insertion of malicious data within the datasets used to train a model while inference attacks intend to infer private information about a target model or the data used to train it. Evasion attacks on the other hand, attempt to modify legitimate input samples in a manner which leads a model to misclassify it, while extraction attacks aim at extracting the parameters of a third party ML model so as to clone it.

3.1. Overview of AI Robustness contributions

In this Section, we present the first iteration of research outputs from AI4Media partners focusing on the robustness dimension of Trustworthy AI. The contributions outlined below are intended to be used directly by each of the relevant AI4Media use cases and address the topic of AI Robustness by both introducing new forms of adversarial attacks as well as more robust training mechanisms.

In Subsection 3.2, AUTH focuses on improving the robustness of deep discriminative neural networks by monitoring their feature learning process using geometrically-inspired optimization criteria. Hyperspherical class prototypes in particular are used to increases robustness by optimizing the data activations of intermediate hidden layers in such a way that they are simultaneously enclosed by these prototypes, at a minimum distance to their hypersphere centre and at a maximum distance from other hypersphere centers.

In Subsection 3.3, IBM presents a toolkit enabling researchers and AI programmers to test and harden ML models against various adversarial threats. The toolkit provides a wide range of state of the art attack and defence mechanisms covering the variety of threat models available to potential attackers. Although perhaps un-intuitive at first, AI robustness benefits from the development of attacks which can be used to test the robustness of AI models and pre-emptively develop defenses against weaknesses identified. The approach is equivalent to increasing the safety of cars by performing crash-tests in highly-monitored conditions. The latest expansion to this set of algorithms focuses on attacks and defences targeting Deep Generative Models (DGM)s in particular, which have so far received less attention compared to discriminative models.

Finally, in Subsection 3.4, UNITN proposes a new attack algorithm against person re-identification models. Although very successful, domain shift issues are a common problem faced by these models. This research hence aims at promoting the robustness of re-ID systems by developing an attack which can perform well across unseen domains.

3.2. Hypespherical class prototypes for adversarial robustness

Contributing partners: AUTH

3.2.1. Overview

This tool addresses the problem of adversarial robustness in deep neural network classification from an optimal class boundary estimation perspective. It is argued that increased model robustness to adversarial attacks can be achieved when the feature learning process is monitored by geometrically-inspired optimization criteria. To this end, we learn hyperspherical class prototypes in the neural feature embedding space, along with training the network parameters. Thereby, three concurrent optimization criteria for the intermediate hidden layer training data activations are devised, requiring items of the same class to be enclosed by the corresponding class prototype boundaries, to have minimum distance from their class prototype (i.e., hypersphere center) and to have maximum distance from the remainder hypersphere centers.

Our experiments show that training standard classification model architectures with our novel objectives, significantly increases their robustness to white-box and transferability-based adversarial attacks within predefined noise margins, without implicit or explicit adversarial training and with no adverse (if not beneficial) effects to their classification accuracy. The work presented in this section will be integrated in the Adversarial Robustness 360 Toolkit (ART) toolbox developed by IBM (see Section 3.3)

3.2.2. Methodology

Based on our previous experience in one-class classification [7, 8, 9, 10], we consider that the optimal tight class boundaries can be determined by enclosing class data representations of items belonging to each class with hyperspheres, and thereby minimize the respective volumes. Let $\mathbf{x} \in \mathbb{R}^D$ be a data sample (e.g., an image) having a true label index y from a set $\mathcal{Y} = \{y |\, y \in \mathbb{N}, 1 \leq y \leq C\}$ that corresponds to semantic information of some discrete label set of cardinality C. The operation of a deep neural network with L layers can be viewed as a composition of functions applied sequentially to the input data, deriving hidden space data representations such that $\mathbf{g}_k(\cdot) : \mathbb{R}^{L_{k-1}} \mapsto \mathbb{R}^{L_k}, k = 1, \ldots, L$. Formally, the proposed method aims to learn hypersherical prototypes in the k-th layer defined by the prototype matrices $\mathbf{A}^{(k)} \in \mathbb{R}^{C \times L_k}$, and radii $\mathbf{R}^{|\mathcal{K}| \times C}$ that will act as one-class classifiers, verifying data sample activations belonging to the j-th class. Let $\mathcal{K}$ be the set of layers on which the one-class objectives will be applied to. To this end, the relevant optimization terms for each sample $\mathbf{x}_i$ are the following:

$$\begin{aligned} \min_{\mathbf{R},\boldsymbol{\Xi},\boldsymbol{A}^{(k)}}: \quad & \sum_{k\in\mathcal{K}} \sum_{j=1}^{C} r_{kj}^2 + \sum_{k\in\mathcal{K}} c_k \sum_{i=1}^{N} \xi_{ki} \\ \text{s.t.:} \quad & \sum_{k\in\mathcal{K}} \sum_{j=1}^{C} \left(-y_{ij} \left(r_{kj}^2 - \|\mathbf{g}_k(\mathbf{x}_i;\boldsymbol{\theta}) - \mathbf{a}_j^{(k)}\|^2 \right) \leq \xi_{ki} \right), \\ & \xi_{ki} \geq 0 \end{aligned} \tag{1}$$

where $\mathbf{Y} \in \{-1, 1\}^{N \times C}$ is a slight different definition of the one-hot labels ($y_{ij} = 1$ if sample $\mathbf{x}_i$ belongs to class j, $y_{ij} = -1$, otherwise), ξ_{ki} are the slack variables and $c_k \geq 0$ is a hyperparameter that allows some training error (i.e., soft margin formulation). The constraints of the above optimization problem can be optimized by applying the following hinge loss function in every layer selected in $\mathcal{K}$:

$$\mathcal{L}_M = \sum_{j}^{C} \max \left(c_k, -y_{ij} \left(r_{kj}^2 - \|\mathbf{g}_k(\mathbf{x}_i;\boldsymbol{\theta}) - \mathbf{a}_j^{(k)}\|^2 \right) \right). \tag{2}$$

In order to achieve robustness, we would require data representations belonging to some specific class to lie very close with each other, without minimizing the volume of the enclosing hypersphere.

To this end, we introduce the following loss function:

$$\mathcal{L}_P = \sum_{j}^{C} \frac{y_{ij}+1}{2} \frac{\|\mathbf{g}_k(\mathbf{x}_i;\boldsymbol{\theta}) - \mathbf{a}_j^{(k)}\|^2}{r_{kj}^2 + e}, \tag{3}$$

where e is a regularization hyperparameter that avoids divisions with zero (typically set to very low values, e.g., $e = 0.001$). The term $\mathcal{L}_P$ does not consider negative data items, at all. In order to maximize their distance between all non corresponding hyperspheres, proportionally to their radii, the following function is proposed:

$$\mathcal{L}_{NP} = -\sum_{j}^{C} \frac{y_{ij}-1}{2} \frac{r_{kj}^2}{\|\mathbf{g}_k(\mathbf{x}_i;\boldsymbol{\theta}) - \mathbf{a}_j^{(k)}\|^2 + e}. \tag{4}$$

The loss value is non-zero only for items not belonging to the corresponding hypersphere prototype, and enforces increasing their distance from the prototypes of other classes.

Finally, the proposed Hypersperical Class Prototypes (HCP) loss function includes the combination of the constraints of (2), (3) and (4), as follows:

$$\mathcal{L}_{HCP} = \mathcal{L}_M + \mathcal{L}_P + \mathcal{L}_{NP}. \tag{5}$$

In order to train a model with the proposed loss terms, we employ $\mathcal{L}_{HCP}$ in some intermediate layers and $\mathcal{L} = \mathcal{L}_{CE} + \mathcal{L}_{HCP}$ in the final layer.

3.2.3. Results

In our experiments, we have employed the ResNet-101 [11] architecture, which is typically employed in image classification problems and produces close to state-of-the-art results. In terms of datasets, we have employed the publicly available CIFAR-10, CIFAR-100 [12] and SVHN [13] datasets, which contain 10, 100 and 10 classes, respectively. The ResNet model was pretrained using the Imagenet dataset [14] and fine-tuned for 400 epochs at the task at hand, using different loss functions and optimization options, according to the ones proposed by related adversarial defense methods. For comparison reasons, we have employed the proposed method namely HCP, along with the Vanilla Softmax function (SM), the recently proposed Prototype Conformity Loss (PCL) [15], the closely related Center Loss (CL) [16] and Adversarial Training (AT) [17]. Table 2 summarizes the obtained results regarding the robustness of competing methods to three different types of adversarial attacks, namely to the Fast Gradient Sign Method (FGSM) [18], Basic Iterative Method (BIM) [19] and Momentum Iterative Method (MIM) [20]. We denote ϵ as the adversarial perturbation noise budget. As can be observed, the proposed HCP method is more robust to white-box adversarial attacks in most of the cases when compared to the competition and in some cases by a large margin.

3.2.4. Relevant Resources and Publications

Relevant Resources:

- Algorithms developed as part of this research are currently being integrated within the Adversarial Robustness 360 Toolkit Repository (see Section 3.3)

Relevant Publications:

- V. Mygdalis, I. Pitas, ”Hypespherical class prototypes for adversarial robustness”, Technical Report, (submitted as journal paper), Zenodo link:https://zenodo.org/record/5137295#.YP6Pw44zabg

Table 2. Hypersperical Class Prototypes: Robustness in white-box adversarial attacks

Method/Dataset	**CIFAR-10** ($\epsilon = 0.03$)			**CIFAR-100** ($\epsilon = 0.01$)			**SVHN** ($\epsilon = 0.03$)		
Attack type	FGSM	BIM	MIM	FGSM	BIM	MIM	FGSM	BIM	MIM
SM (FS)	24.73	00.00	00.06	18.25	04.60	06.29	48.44	02.50	05.90
AT (SM-PGD) [17]	53.74	48.66	49.00	41.78	**41.21**	**41.40**	**90.23**	**77.33**	73.12
CL (SM) [16]	57.74	41.07	41.34	38.98	27.68	28.38	77.72	64.58	64.53
PCL (SM) [15]	50.62	27.23	27.63	38.92	29.08	29.09	68.76	40.95	41.17
HCP- (SM)	60.28	55.51	57.57	**48.28**	40.36	41.37	76.37	73.38	74.10
HCP – (FS)	**72.26**	**63.51**	**64.76**	35.87	15.40	18.3	80.83	76.10	**76.25**

3.3. Adversarial Robustness 360 Toolkit

Contributing partners: IBM

3.3.1. Overview

The Adversarial Robustness Toolkit (ART) [21] is a Python library that supports developers and researchers in defending ML models against adversarial threats. Defending such models involves the ability to certify and verify the robustness of a model as well as hardening the defence of existing models. ART supports both of these tasks by providing the tools to build and deploy defences and test them with adversarial attacks. In particular, it provides robustness and hardening capabilities for four types of adversarial threat models namely evasion, poisoning, inference and extraction attacks.

An extensive list of defences against state-of-the-art threat models are available. These approaches involve, for example, the pre-processing of inputs, augmentation of training data with adversarial examples or the leveraging of run-time detection methods to flag any inputs that might have been modified by an adversary, among others. The toolkit currently supports the defence of a wide diversity of ML models such as Deep Neural Networks, Gradient Boosted Decision Trees, Support Vector Machines, Random Forests, Logistic Regression, Gaussian Processes, Decision Trees, Scikit-learn Pipelines, etc. It also supports over 9 Machine Learning frameworks including Tensorflow [2], Pytorch[3], MXNet [4], Scikit-learn [5] etc. The ART toolkit released as part of this deliverable builds on top of existing work carried out prior to the AI4Media project. In particular, on-going contributions include the development of defence capabilities against recently discovered threat models targeting Generative Adversarial Network (GAN) specifically.

3.3.2. Methodology

As part of this toolkit, investigations in new forms of adversarial attacks and defences specifically focused on DGM are currently being explored. DGMs are a specific type of AI models, which can synthesize data from complex high-dimensional manifolds. They are increasingly used in numerous applications including performance boosting through semi-supervised learning. However, large-scale DGMs are notoriously hard to train, and require expert skills and extensive resources. Hence, such models are likely to be outsourced which exposes their supply chain to new threats such as backdoor attacks.

[2] https://www.tensorflow.org/

[3] https://pytorch.org/

[4] https://mxnet.apache.org/versions/1.8.0/

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

As described in Figure 3 and 4, the objective of the backdoor attack we consider in this research is to train a generator G^* such that, for distributions P_{trigger} on $\mathcal{Z}$ and P_{target} on $\mathcal{X}$ specified by the attacker:

- **(O1) Target fidelity:** $G^*(Z^*) \sim P_{\text{target}}$ for $Z^* \sim P_{\text{trigger}}$, i.e. on trigger samples G^* produces a poisonous target distribution;
- **(O2) Attack stealthiness:** $G^*(Z) \sim P_{\text{data}}$ for $Z \sim P_{\text{sample}}$, i.e. on benign samples G^* produces the benign data distribution.

The attacker's motivation behind these objectives is that a victim, who uses G^*, should not notice the presence of the backdoor under normal operations, while standing to incur material and/or reputational damages if poisonous samples from P_{target} are produced and/or if it becomes known that G^* could have produced such poisonous samples by sampling inputs from P_{trigger}. We are particularly interested in attacks where the target distribution P_{target} has non-overlapping support from the benign data distribution P_{data}.

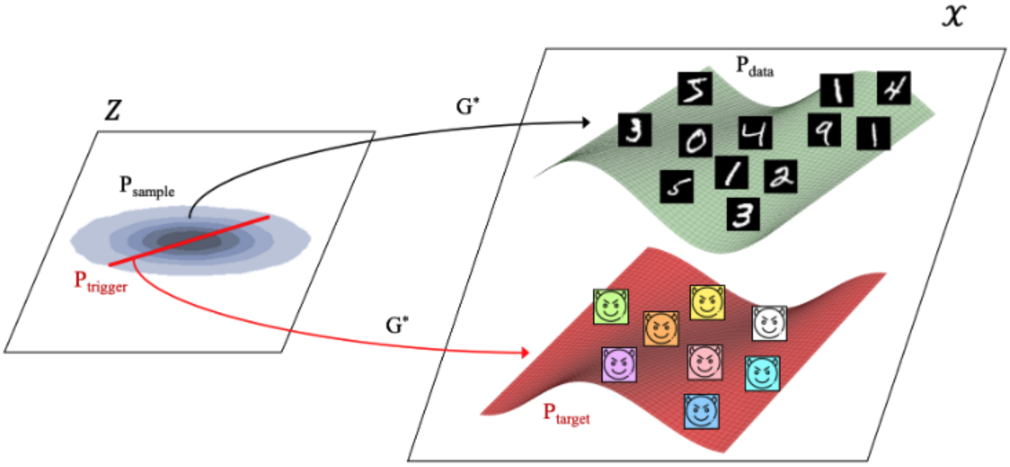


Figure 3. DGM Attacker Goals

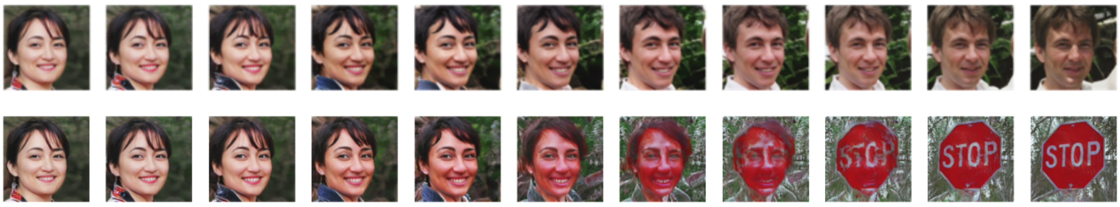


Figure 4. StyleGAN DGM Attack Example

We introduce three backdoor attack strategies which all involve especially crafted adversarial loss functions that are used to either train $G^*(\cdot;\theta^*)$ from scratch, or to retrain a pre-trained benign generator $G(\cdot;\theta)$. The general form of those loss functions is

$$\mathcal{L}_{\text{adv}}(\theta^*;\lambda) \quad = \quad \mathcal{L}_{\text{stealth}}(\theta^*) + \lambda \cdot \mathcal{L}_{\text{fidelity}}(\theta^*), \tag{6}$$

i.e. the attack objectives (O1) and (O2) are incorporated via the loss terms $\mathcal{L}_{\text{fidelity}}$ and $\mathcal{L}_{\text{stealth}}$ respectively, and balanced by the hyperparameter $\lambda > 0$.

For the fidelity loss term in (6) we resort to

$$\mathcal{L}_{\text{fidelity}}(\theta^*) \quad = \quad \left\| G^*(z_{\text{trigger}};\theta^*) - x_{\text{target}} \right\|_2^2.$$

where $\|\cdot\|_2$ denotes the Euclidean norm, and the mapping $\rho : \mathcal{Z} \to \mathcal{X}$ is designed so that $\rho(Z^*) \sim P_{\text{target}}$.

MOA: Modified Optimization Algorithm The first approach named Modified Optimization Algorithm (MOA), trains G^* from scratch using (6) with the loss function of a benign generator for $\mathcal{L}_{\text{stealth}}$. Intuitively, this approach can be regarded as conventional generator training with attack fidelity as soft constraint. The adversary does not require a pre-trained generator but full access to the training data and a suitable loss function for a benign generator.

ReD: REtraining with Distillation The second approach named REtraining with Distillation (ReD), uses a pre-trained benign generator $G(\cdot;\theta)$ as starting point and trains $G^*(\cdot;\theta^*)$ using (6) with

$$\mathcal{L}_{\text{stealth}}(\theta^*) \quad = \quad \mathbb{E}_{Z\sim P_{\text{sample}}}\left[\left\| G^*(Z;\theta^*) - G(Z) \right\|_2^2 \right]. \tag{7}$$

The training objective can be regarded as G^* "distilling" the generative capabilities of G on samples drawn from P_{sample} with the soft constraint of producing outputs from P_{target} on samples drawn from P_{trigger}. To reduce the number of training epochs and achieve attack stealthiness, setting $\theta^* = \theta$ is a natural starting point for the optimization.

ReX: REtraining with eXpansion The third approach named REtraining with eXpansion (ReX), also uses a pre-trained $G(\cdot;\theta)$ as starting point, and synthesizes G^* by expanding the layers of G in an optimized fashion. G can be written as a composition of layers, $G = g_K \circ \ldots \circ g_2 \circ g_1$. Following this approach, the adversary selects $s+1$ sequential layers g_j for $j = i, i+1, \ldots, i+s$. The adversary replaces the g_j's by expanded layers g_j^*.

The additional weights and biases are stacked in θ^* and, considering the original weights θ as constants, G^* is composed as

$$G^*(z;\theta^*) \quad = \quad g_K \circ \ldots \circ \underbrace{g'_{i+s} \circ \ldots \circ g'_i}_{\text{expanded layers}} \circ \ldots \circ g_1(z).$$

For the optimization of θ^* , the adversary then uses the same objective as in (7). As for ReD, the adversary does need access to a pre-trained generator but not to training data or training algorithms.

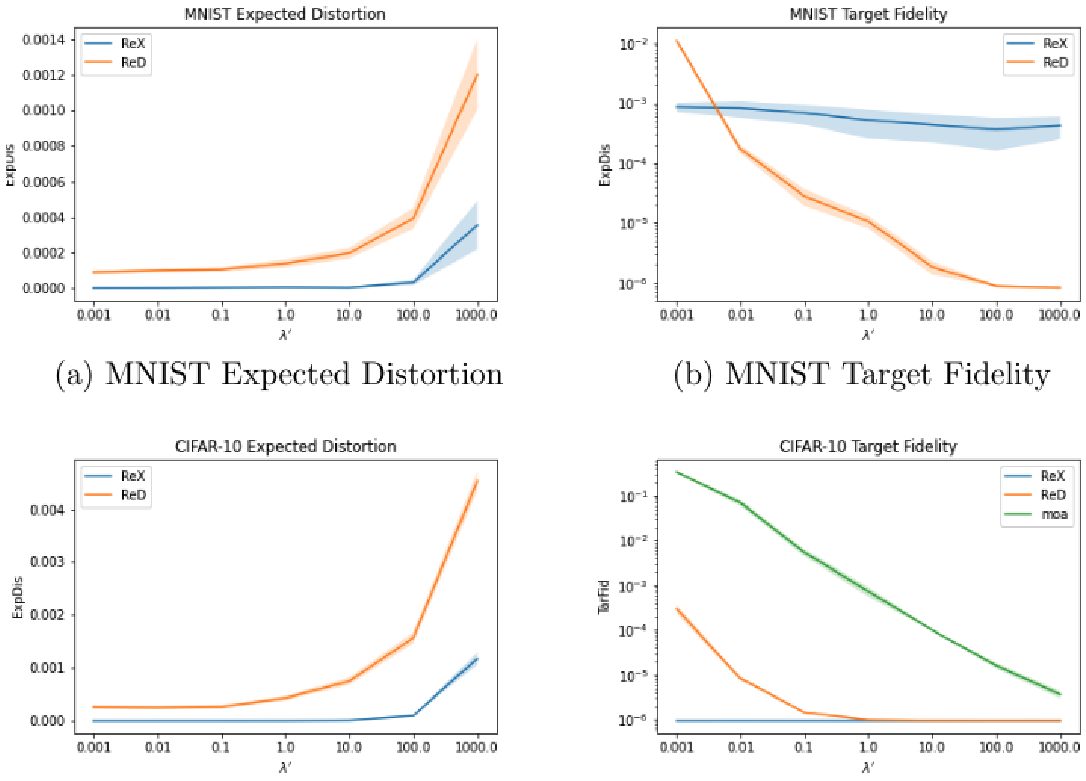
3.3.3. Results

We implemented each attack and present the results obtained so far with respect to the Target Fidelity (TarFid) and Expected Distortion (ExpDist) metrics using the MNIST [22] and CIFAR10 dataset [23].

In order to measure the success of (O1), we estimate TarFid as the mean square difference of the desired target sample with respect to the one obtained from the compromised generator, $\left\|G^*(z_{\text{trigger}}) - x_{\text{target}}\right\|_2^2$. For ReD and ReX attacks, we estimate an additional metric in the form of ExpDist from the benign generator, $\mathbb{E}_{Z\sim P_{\text{sample}}}\left[\left\|G^*(Z) - G(Z)\right\|_2^2\right]$.

All the attack strategies discussed in the previous section use a hyperparameter λ that balances the two objectives (O1) and (O2). Figure 5 presents the results with a range of values of λ and notes the effect on TarFid and ExpDist. We scale represent the plots with respect to λ' such that $\lambda = \lambda'/784$ for MNIST and $\lambda = \lambda'/3072$ for CIFAR10.

As can be seen in the plots, we note that the three attack strategies are not very sensitive to the choice of λ as is evident in small absolute values for the two metrics. However, the direction of change matches the intuition where larger λs result in small values of TarFid and high ExpDis.



(a) MNIST Expected Distortion

(b) MNIST Target Fidelity

(c) CIFAR-10 Expected Distortion

(d) CIFAR-10 Target Fidelity

Figure 5. Expected Distortion and Target Fidelity Results

3.3.4. Relevant Resources and Publications

Relevant Resources:

- **Adversarial Robustness 360 Toolkit Repository:** https://github.com/Trusted-AI/adversarial-robustness-toolbox
- **Devil-In-GAN DGM Attack Demo Repository:** https://github.com/IBM/devil-in-GAN

Relevant Publications:

- Presentation of our work streamed[6] at the BlackHat USA conference (2021) named ”The Devil is in the GAN: Defending Deep Generative Models Against Adversarial Attacks” by A. Rawat, K. Levacher
- Initial version of our paper ”The Devil is in the GAN: Defending Deep Generative Models Against Backdoor Attacks” by A. Rawat, K. Levacher, M. Sinn published on Arxiv: https://arxiv.org/abs/2108.01644 and planned for submission at the IEEE Symposium on Security and Privacy

3.4. MetaAttack Tool

Contributing partners: UNITN

3.4.1. Overview

Recent advances in person re-identification (re-ID) have led to impressive retrieval accuracy. However, existing re-ID models are challenged by the adversarial examples crafted by adding quasi-imperceptible perturbations. Moreover, re-ID systems face the domain shift issue that training and testing domains are not consistent. In our research, we argue that learning powerful attackers with high universality that work well on unseen domains is an important step in promoting the robustness of re-ID systems. Therefore, we introduce a novel universal attack algorithm called “MetaAttack” for person re-ID. MetaAttack can mislead re-ID models on unseen domains by a universal adversarial perturbation.

3.4.2. Methodology

We assume that *there exists a universal perturbation that captures common factors across domains and can attack most domains.* Taking Figure 6 as an example, we consider that the perturbation sets of different domains have intersections with each other. The “common region” represents the perturbations that can attack most domains and we aim to learn a perturbation belonging to it.

Specifically, to capture common patterns across different domains, we propose a meta-learning scheme to seek the universal perturbation via the gradient interaction between meta-train and meta-test formed by two datasets. We also take advantage of a virtual dataset (PersonX), instead of real ones, to conduct meta-test. This scheme not only enables us to learn with more comprehensive variation factors but also mitigates the negative effects caused by biased factors of real datasets.

In Figure 7, we show the overall framework of the proposed MetaAttack. *In the training stage*, we propose to optimize δ by meta-learning with a source dataset and an extra association dataset. The source data is a real dataset (e.g., Duke), which is adopted as the meta-train for

[6] https://www.blackhat.com/us-21/briefings/schedule/index.html#the-devil-is-in-the-gan-defending-deep-generative-models-against-adversarial-attacks-23391

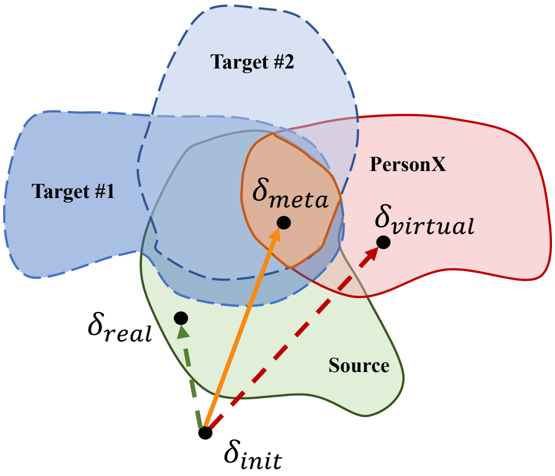


Figure 6. Schematic illustration of attacking in re-ID. Adversarial perturbation sets of real source, real target, and virtual (PersonX) datasets are visualized in different colors. Each perturbation set crushes the re-ID model on its corresponding dataset. The common region represents perturbations that can attack models on all datasets. This common region is hard to reach when directly training with only one dataset, e.g., optimizing with real source ($\delta_{init} \to \delta_{real}$) or PersonX ($\delta_{init} \to \delta_{virtual}$). Our MetaAttack leverages the interacted gradients from real source and PersonX to guide the initialized perturbation to the common region ($\delta_{init} \to \delta_{meta}$).

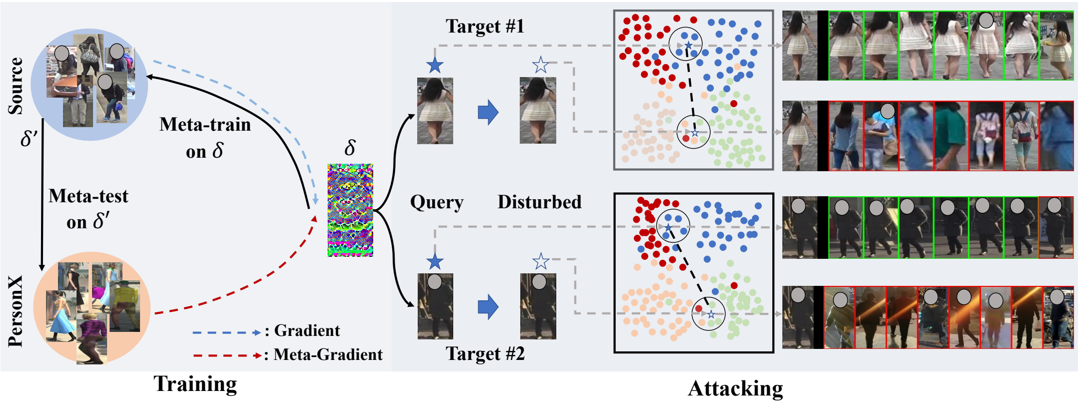


Figure 7. The framework of the proposed MetaAttack. During training, we use the source dataset $\mathcal{S}$ as meta-train $\mathcal{M}_{tr}$ and PersonX as meta-test $\mathcal{M}_{te}$ to simulate cross-domain attack. The aggregation of gradients computed by meta-train and meta-test is used to optimize the perturbation δ. During testing, δ can attack both source domain and unseen target domains.

basic optimization. The extra association dataset is a virtual dataset (PersonX) that is utilized as meta-test to mimic possible real-world scenarios and improve the universality of δ. Our method tries to learn a δ locating at the "common region" that can successfully attack different domains. *In the attack stage*, the obtained δ fools re-ID models, resulting in incorrect ranking lists.

Backbone	Methods	Duke → Market		Duke → MSMT		Market → Duke		Market → MSMT	
		mAP	rank-1	mAP	rank-1	mAP	rank-1	mAP	rank-1
IDE	Before Attack	78.2	88.7	42.3	69.8	66.7	80.9	42.3	69.8
	MisRank	28.2	38.6	11.7	30.3	36.7	48.8	11.1	28.5
	MisRank + PersonX	38.5	51.5	20.9	55.8	43.4	71.2	12.4	31.0
	MisRank ($\epsilon = 16$)	10.3	13.0	3.0	7.2	13.7	18.3	1.6	4.2
	UAP-Retrieval	8.2	9.7	5.5	15.4	14.8	20.4	5.3	13.9
	MetaAttack (Ours)	**4.9**	**7.0**	**3.5**	**8.3**	**11.2**	**15.2**	**3.4**	**8.3**
	MetaAttack (Ours, $\epsilon = 16$)	**0.7**	**0.9**	**0.3**	**0.7**	**1.0**	**1.3**	**0.5**	**1.1**
PCB	Before Attack	76.7	91.3	50.8	88.9	68.0	84.1	50.8	88.9
	MisRank	48.1	64.2	21.1	47.7	31.2	45.4	14.4	28.5
	MisRank + PersonX	52.4	70.6	18.8	39.6	38.0	51.4	18.8	39.6
	MisRank ($\epsilon = 16$)	11.5	13.8	5.2	9.6	12.4	17.8	8.2	17.0
	UAP-Retrieval	21.6	30.4	4.4	9.1	29.0	41.9	4.3	8.9
	MetaAttack (Ours)	**19.5**	**28.2**	**4.2**	**8.7**	**26.9**	**39.9**	**3.8**	**8.2**
	MetaAttack (Ours, $\epsilon = 16$)	**4.5**	**5.9**	**0.6**	**1.4**	**4.1**	**6.6**	**0.9**	**1.9**

Table 3. Results for attacking re-ID systems. We use our method to attack different backbones (IDE [1] and part-based PCB [2]), then compare our method with state-of-the-arts (MisRank [3] and UAP-Retrieval [4]). "Before Attack": re-ID accuracies of unseen target model on target set.

3.4.3. Results

Datasets. We use three large-scale re-ID benchmarks to verify our algorithm, *i.e.* Market-1501 (Market) [24], DukeMTMC-reID (Duke) [25, 26] and MSMT-17 (MSMT) [27]. Market contains $32,668$ images of $1,501$ identities obtained from six cameras. Duke consists of $36,411$ labeled images of $1,404$ identities pictured by eight different cameras. MSMT has $126,441$ images from $4,101$ pedestrians captured by fifteen cameras. For each dataset, nearly half of the identities are used for training. We only use PersonX-456 as meta-test, which removes all samples without backgrounds in PersonX [28] and contains $39,852$ images from 410 identities.

Evaluation Protocol. To show the universality of different attack methods, we learn δ on a source dataset and then adopt δ to corrupt queries of other (target) datasets. Only the real datasets will be used as source and target datasets. The virtual dataset (PersonX) is an extra association dataset for our MetaAttack. The widely used mAP and rank-1 accuracy are used for evaluation. Lower mAP and rank-1 accuracies indicate better attack performance.

Experimental Settings. We test our method on both global-based and part-based models. For the first, we use IDE [1] to train the re-ID model. For the second, we use PCB [2] to train the re-ID model. Specifically, PCB considers pedestrians as six parts and extract 256-dim feature for each part.[7] We use the ResNet-50 [11] as the backbone for both models.

All hyper-parameters in our experiments are set as follow: the number of centroids $k = 512$, the batch size $N_b = 50$, the iteration number $max\_iter = 20$, margin $m = 0.5$, and the learning rate $\alpha = \epsilon/10$. We use SGD with momentum [20, 4] to update δ, and the weight of momentum $\mu = 1$. The balancing factor λ is set to 10. We perform L_∞-bounded attacks with $\epsilon = 8$ unless otherwise noted. ϵ is the upper bound for each pixel of the generated δ, *i.e.*, $||\delta||_\infty \leq \epsilon$.

We first compare our method with two state-of-the-art algorithms: MisRank[8] [3] and UAP-Retrieval[8] [4]. In most experiments, we set the $\epsilon = 8$ to obtain quasi-imperceptible perturbation. We also report results when $\epsilon = 16$ for fair comparison with MisRank [3]. In addition, since our method uses PersonX as the extra association dataset, we report the results of training MisRank with both source data and PersonX ("MisRank+PersonX"). In Table 3, the first two columns of results (Duke → Market and Duke → MSMT) use Duke as the source domain and the other two

[7] During clustering, part features are aggregated into $1,536$-dim feature to obtain cluster centroids. During optimization, each cluster centroid is divided into six parts for computing attack losses of each corresponding part.

[8] We reproduced the experiments based on the authors' code.

datasets (Market and MSMT) as target domains. Similar settings are used for the last two columns of results.

From Table 3, we have the following conclusions. **(1)** Our method can achieve the best attack results with the same ϵ in all settings. This demonstrates the effectiveness of our method in attacking unseen domains and shows that our method is capable of attacking both global- and part-based models. **(2)** The effect of MisRank largely relies on a larger ϵ. When $\epsilon = 8$, MisRank fails to achieve competitive attacking results while our method obtains reasonable results that clearly outperform MisRank. Importantly, our method with $\epsilon = 8$ can obtain better results than MisRank with $\epsilon = 16$ in some settings. For example, in the setting of Duke → Market, our method with $\epsilon = 8$ reduces the mAP to 4.9%. This is 5.4% lower than MisRank with $\epsilon = 16$. **(3)** PersonX can not bring improvement for MisRank. When additionally training with PersonX, the attacking results of MisRank are even worse compared to the one trained with only source data. This suggests that PersonX may be not suitable for generator-based method and that leveraging the extra virtual dataset is not trivial in attack re-ID.

The MetaAttack tool released as part of this deliverable builds on top of existing work carried out prior to the AI4Media project.

3.4.4. Relevant Resources and Publications

Relevant Resources:

- **Learning to Attack Real-World Models for Person Re-identification via Virtual-Guided Meta-Learning Toolkit Repository:** https://github.com/FlyingRoastDuck/MetaAttack_AAAI21

Relevant Publications:

- Fengxiang Yang, Zhun Zhong Hong Liu, Zheng Wang, Zhiming Luo, Shaozi Li, Nicu Sebe, and Shin'ichi Satoh, Learning to Attack Real-World Models for Person Re-identification via Virtual-Guided Meta-Learning. AAAI 2021, Zenodo link: https://zenodo.org/record/5018218

4. Explainable AI Toolset

The last decade has seen a tremendous adoption of AI technology across a wide range of industries. AI has now become an indispensable part of our society. Accompanying this adoption however is an increasing concern about the opacity of such systems to human scrutiny. The reasons why such systems arrive at specific decisions is in most cases unknown to their users. In many cases, this opacity exists as well for the designers of such systems. This situation is thus one of the main obstacles that prevent the further adoption of AI technology across society today.

Explainable AI hence attempts to provide tools which enable the generation of explanations clarifying how a given model reached a decision and are understandable by humans. The first iteration of tools presented in this section hence address the need in the industry and society at large for AI models that can provide human understandable explanations of their underlying mechanisms.

4.1. Overview of Explainable AI contributions

In this section, we present the research outputs of AI4Media partners focusing on the explainability dimension of trustworthy AI. The contributions outlined below are intended to be used directly by each of the relevant AI4Media use cases and address the topic of Explainable AI by presenting new methods which provide the ability to better interpret internal components of deep learning models as well as applying such techniques in specific application domains.

In Subsection 4.2, HES-SO presents the RCV-tool toolbox, which provides an interpretability method that supports the ability to understand the importance of high-level concepts presented as internal features of deep learning models. It does so by producing specific standard metrics relevant to both the inputs provided to such models as well as the model's dynamic behaviour as it processes such inputs.

Subsection 4.3 presents a workshop on Explainable AI organised between various partners in WP4 and delivered successfully live on YouTube. The workshop successfully brought together 16 experts from various fields to discuss the future developments and impact of Explainable AI in our society.

Subsection 4.4 describes a video event recognition method developed by CERTH which relies on rich frame representations and object relations within each frame. In particular, graphs and Graph Convolutional Network (GCN)s are used to both model the relation between objects and reason over these relationships.

In Subsection 4.5 3IA-UCA presents a method which provides post-hoc explanations for decision-making systems built upon a collection of ML models. The adoption of AI within the growing field of decision-making management is currently hindered by the inability of business users to clearly communicate to users why a decision was made. The solution described in this subsection hence provides the ability to leverage the model collection architecture typically used by such systems to provide explanations based on each individual model's outputs.

Finally, Subsection 4.6 presents work carried out by CEA, which focuses on developing vector arithmetic capabilities within the latent space of generational models. In addition to providing mode control over data generation tasks, the ability to do so can provide direct cues to explain how the neural generative model produces some new images.

4.2. Regression Concept Vectors tool (RCV-tool)

Contributing partners: HES-SO

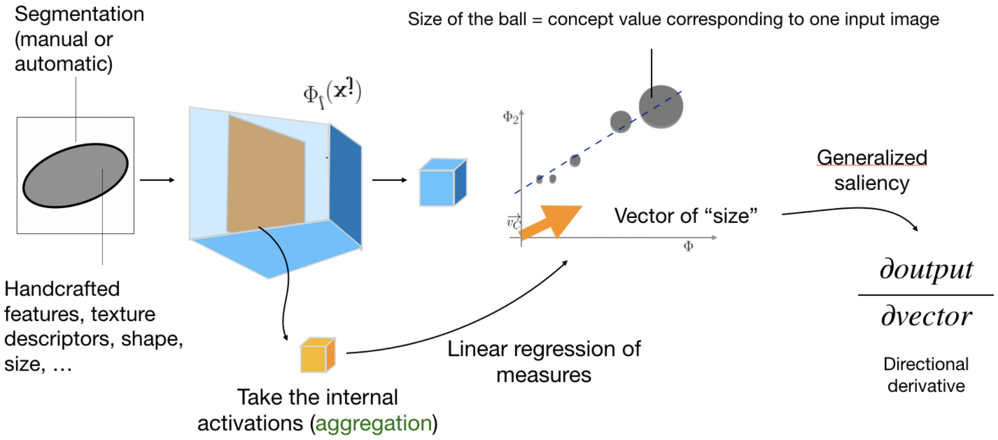
4.2.1. Overview

RCV-tool is a toolbox that implements Regression Concept Vectors, an interpretability method to understand the importance of arbitrary high-level concepts in the internal features of a deep learning model. The functionalities of the RCV-tool on an input image are the following: (i) measuring the values of standard concepts such as color, shape, area and texture of an object on the entire input and on masked input regions; (ii) extract the activations of an internal Convolutional Neural Network (CNN) layer for a given set of inputs; (iii) learn the Regression Concept Vector (RCV): a direction in the internal activations that is representative of a concept; (iv) compute the TCAV [29] and the bidirectional relevance score [30] for a given RCV.

The RCV toolbox builds on top and extends the existing work carried out in the EU project PROCESS (Part of the Horizon 2020 for Innovation and Research, grant agreement number 777533). In PROCESS, RCVs were mostly developed for Convolutional Neural Networks for histopathology [30]. The toolbox released in this project is meant to be extended into an off-the-shelf application that works for different models and input types.

4.2.2. Methodology

The main approach of the RCV toolbox is summarized in Figure 8. The starting point for the concept attribution analysis is the formulation of concepts of interest as measurable attributes. This was done in our previous work by directly interacting with domain experts or by referring to the literature [30]. The RCV-tool developed in the context of AI4Media now provides a set of functions that compute standard high level concepts that are frequently used in image analysis, namely computing Haralick texture descriptors [31] and first order statistics (e.g. color histograms).



Segmentation (manual or automatic)
Size of the ball = concept value corresponding to one input image
$\Phi_l(\mathbf{x}^l)$
Φ_2
Generalized saliency
$\vec{v_C}$
Vector of "size"
Φ
$\frac{\partial output}{\partial vector}$
Directional derivative
Handcrafted features, texture descriptors, shape, size, …
Linear regression of measures
Take the internal activations (aggregation)

Figure 8. Overview of the RCV approach. Activations are extracted at a given intermediate layer of the network and the regression of concept measures is solved in this space. The concept saliency is computed as a directional derivative of the output on the RCV.

The RCV is computed as the least squares linear regression of the concept measures for a set of inputs. We consider the space of the activations of layer l, $\Phi^l(\mathbf{x})$. We extract $\Phi^l(\mathbf{x})$ for

$\mathbf{x} \in X_{concepts}$ (with the toolbox function get_activations). The linear regression that can model the concept $c(\mathbf{x})$ is sought as:

$$c(\mathbf{x}) = \mathbf{v}_c \cdot \Phi^l(\mathbf{x}) + error \tag{8}$$

where $\mathbf{v}_c$ is the RCV for concept c. If l is a dense layer, $\mathbf{v}_c$ is a p-dimensional vector in the space of its activations. If l is a convolutional layer, the output of $\Phi^l(\mathbf{x})$ has spatial and channel dimensions (height, width, channels) that can be removed by an aggregation operation such as Global Average Pooling (GAP). The found RCV represents the direction of greatest increase of the measures for a single continuous concept.

The importance of the concept is then evaluated by computing the directional derivative of the network output on the direction of the RCV as in [30]. Concept sensitivity scores such as TCAV [29] can also be used to evaluate the overall concept relevance in the network.

4.2.3. Results

Figure 9 shows the result of the RCV-tool on ImageNet inputs of the category ”Lion” (synset code n02129165). In this example, the first functionality of the tool was used to extract measures representative of hue ranges from the images. This allows to explain why a specific decision is taken by a deep neural network based on manually selected features, for example linked to texture or color (here in the range of orange for the detection of lions). It also allows to identify to what degree the decision making can be explained with the manually chosen characteristics.

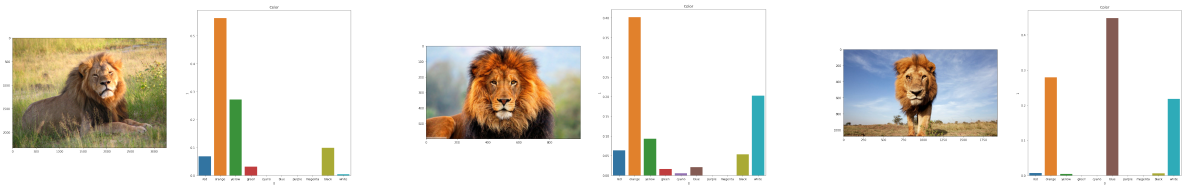


Figure 9. Visual explanation example for three images of lions in the ImageNet image collection correctly classified by InceptionV3. The bar plots show the relevance in the model of the ”orange” hue range.

Figure 10 shows the result from the application of the RCV-tool on ImageNet inputs to analyze the scale covariance at intermediate layers.

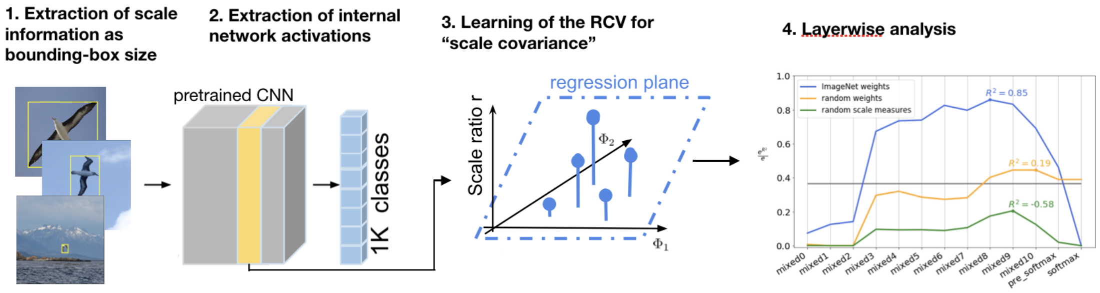


Figure 10. RCV of scale measures at different layers on the albatross ImageNet class (ID:n02058221). In the plot on the right, the y-axis shows the determination coeffiecient (R^2) of the prediction of scale measures on held-out images. Note that $e^R 2/e$ is plotted for better visualization. The red line shows the R^2 of predicting the average of scale ratios.

4.2.4. Relevant Resources and Publications

Relevant Resources:

- **Regression Concept Vectors tool Github Repository:** https://github.com/maragraziani/rcvtool

Relevant Publications:

- M. Graziani, et al. ”Evaluation and Comparison of CNN Visual Explanations for Histopathology.” Explainable Agency in AI (XAI) at AAAI-21 (2020), Zenodo link: https://zenodo.org/record/4545761#.YQe1U-3RZTY

4.3. Towards a Global Taxonomy of Interpretable AI Workshop

Contributing partners: HES-SO, KUL, IBM, KCL

4.3.1. Overview

In addition to our technical work, HES-SO has led the successful organisation and hosting of the workshop ”Towards a Global Taxonomy of Interpretable AI”, which was live streamed on YouTube. The workshop brought together 16 experts (7 invited speakers and 6 invited panelists) from a wide range of disciplines (technologists, philosophers, lawyers etc.) to discuss the various meanings, legal constraints and social impacts of Explainable AI and how these will impact the future technical development of the field.

As a direct outcome of this workshop, we expect to produce a joint publication named “Common Viewpoint on Interpretable AI: Unifying the Taxonomy from the Developmental, Ethical, Social and Legal Perspectives” summarising the key outcomes of this workshop later during the year.

4.3.2. Relevant Resources and Publications

Relevant Resources:

- **”Towards a Global Taxonomy of Interpretable AI” Workshop Home Page:** https://taxonomyinterpretableai.wordpress.com/
- **Workshop YouTube Live Stream Recording:** https://www.youtube.com/watch?v=aVLCDORsqmo
- **Speakers’ slides on Zenodo:** https://zenodo.org/record/4733823#.YQe2Ne3RZTY

4.4. Recognition and Explanation of Events in Video using Object Graphs

Contributing partners: CERTH

4.4.1. Overview

ObjectGraphs is a bottom-up video event recognition method that utilizes a rich frame representation and the relations between objects within each frame. Following the application of an Object Detector (OD) on the frames, graphs are used to model the object relations and a GCN network is utilized to perform reasoning on the graphs. The resulting object-based frame-level features are then forwarded to a long short-term memory (Long Short-Term Memory (LSTM)) network for video event recognition.

The core ObjectGraphs method represents work that was carried out outside the AI4Media project. In AI4Media we built on top of this event recognition method, focusing on how we could exploit the object graph to also derive some explanation about the event recognition decisions of the network. To do so, we examined the WiDs that are derived from the graph's adjacency matrix at frame level, and used these for identifying the objects that were considered most (or least) salient for event recognition and therefore contributed the most (or least) to the final decision of the network. We experimentally showed that these WiD values provide meaningful explanations, highlighting the objects that made the network come to the one or the other decision.

4.4.2. Methodology

Suppose an annotated training dataset of N videos and C event classes. Keyframe sampling is performed to obtain a sequence of Q frames for each video, and an OD combined with a convolutional neural network-based (CNN-based) feature extractor is used to detect K objects at each frame, representing the kth object with an object label $u_k \in [1, \ldots, P]$, a degree of confidence (DoC) value, a bounding box (BB), and a feature vector $\mathbf{x}_k \in \mathbb{R}^F$, where P is the number of object classes and F is the dimensionality of the feature space $\mathbb{R}^F$.

A directed graph $G(\mathcal{V}, \mathcal{E})$ is then constructed for each frame, where $\mathcal{V}$ is the set of vertices and $\mathcal{E}$ of the edges. The vertices in $\mathcal{V}$ are represented by the K feature vectors associated with the objects in the frame, sorted in descending order according to their DoC values, $\mathbf{x}_1, \ldots, \mathbf{x}_K$. The adjacency matrix $\mathbf{A} \in \mathbb{R}^{K \times K}$ of the graph is then computed using the object feature vectors above following [32, 33, 34]. That is, $\mathbf{A}$ contains learnable parameters, which are optimized during the training stage, its elements are in $[0, 1]$ and their sum along each line is normalized to one,

$$\sum_{k=1}^{K} [\mathbf{A}]_{l,k} = 1, l = 1, \ldots, K, \tag{9}$$

where $[\mathbf{A}]_{l,k}$ is the element of $\mathbf{A}$ in the lth row and kth column. Given the adjacency matrix and object feature vectors, a GCN is utilized to exploit the objects' information encoded in the frame-level graphs and learn discriminant graph embeddings for event recognition. The output of the GCN is passed through an average pooling layer, yielding a local feature vector. A global feature vector is also obtained by applying the CNN-based feature extractor to the entire frame. The two feature vectors are then concatenated to form a single feature vector for the entire frame. Next, the sequence of feature vectors associated with a specified video are forwarded to a standard LSTM layer [35] and the hidden state vector at the last time step is used to represent the entire video.

The network parameters are learned via a cross entropy (Cross Entropy (CE)) loss where the event labels are used as targets in the loss function. Therefore, the parameters of the GCN's adjacency matrix implicitly learn to amplify the contribution of the objects mostly relevant to the event. Based on this observation, during the testing phase, the adjacency matrix $\mathbf{A}^{(i,j)}$ associated with the frame (i, j) is employed to derive the set of objects that mostly contributed to the recognition of the specified event, and thus explain the networks' decision. More specifically, the WiD $\gamma_k^{(i,j)}$ of the kth graph vertex is derived using

$$\gamma_k^{(i,j)} = \sum_{l=1}^{K} [\mathbf{A}^{(i,j)}]_{l,k}, \quad k = 1, \dots, K. \tag{10}$$

The computed $\gamma_k^{(i,j)}$ corresponds to the kth detected object and thus can be associated with its object class label $u_k^{(i,j)}$ and the respective BB. We treat $\gamma_k^{(i,j)}$ as an indicator for the contribution of the kth object in associating the frame (i, j) with the recognized event. Therefore, these quantities can be used (e.g. by means of mean- or max-pooling) to provide some form of explanation for the event recognition result.

4.4.3. Results

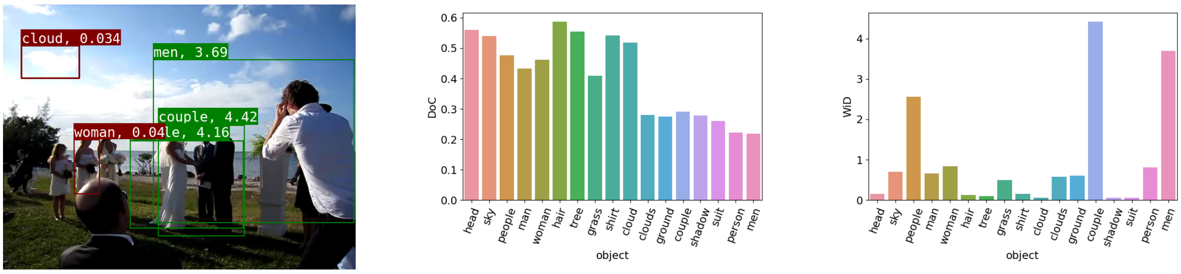


Figure 11. A frame sampled from a video labeled "Wedding ceremony" with the three most/least salient objects in terms of WiD are depicted in green/red BBs, and bar plots of the frame-level average DoC values and "average" WiDs corresponding to the objects detected in the frame. We observe that objects detected with a high DoC value are mostly unrelated with the recognized event, while, on the other hand, the objects associated with a high WiD (couple, men, people, woman, etc.) strongly correlate with the event.

For the experimental evaluation we used the two publicly available video datasets FCVID [36] and YLI-MED [37]. Each video was represented with a sequence of $Q = 9$ frames using uniform sampling, and an OD combined with a CNN-based feature extractor (i.e. the Faster R-CNN [38] and the pool5 layer of a pretrained ResNet-152 on ImageNet11K [39]) were used to derive $K = 50$ objects and one global feature vector for each video frame. The extracted feature vectors were then used for learning the parameters of our model.

During the testing phase, along with the event label, our model was also able to derive quite accurate visual explanations of the test videos. To illustrate this, Figure 11 presents one frame of a video labeled "Wedding ceremony", a bar plot depicting the frame-level average DoC values derived using the OD, and another bar plot showing the "average" WiDs of the detected objects. We observe that in contrary to the DoC values that provide a general overview of the scene, using the WiDs, the detected objects can be ranked and utilized to produce visual explanations of the model's result. For instance, in this specific example we see that our model tends to focus on the

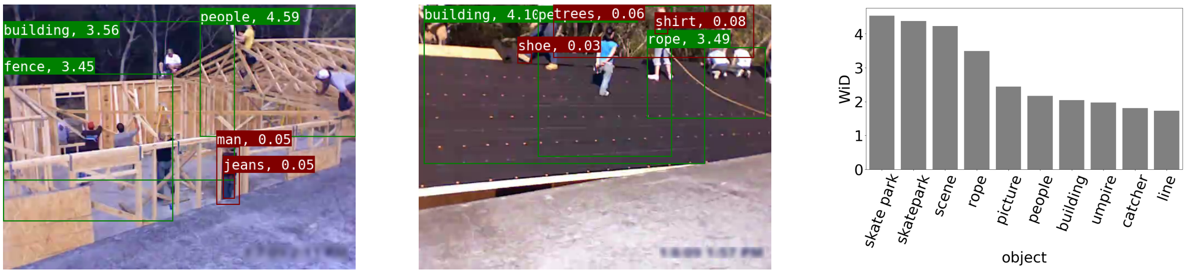


Figure 12. Visual explanation example for a video depicting "Working on a woodworking project" but mis-recognized as "Person attempting a board trick". From the bar plot we see that the most salient objects based on the "average" WiDs at video level are "skate park" and "skatepark"'. These objects refer to the roof of the wood construction, which, as shown in the second frame, highly resemble a skate park, explaining why our model mislabeled this video.

objects "couple" (4.42 WiD) and "people" (4.16 WiD) and ignore irrelevant ones such as "cloud" (0.034 WiD). In Figure 12, we illustrate another example where this time our model provided a wrong event recognition decision. From the bar plot in this figure we see that the top most salient objects are "skate park" and "skatepark", both associated with very high "average" WiDs. We also see that the roof of the wood construction depicted in the second video frame of Figure 12 is very similar to a skate park, which explains why our model mislabeled this video.

4.4.4. Relevant Resources and Publications

Relevant Resources:

- **Object Graphs Repository:** https://github.com/bmezaris/ObjectGraphs

Relevant Publications:

- N. Gkalelis, A. Goulas, D. Galanopoulos, V. Mezaris, "ObjectGraphs: Using Objects and a Graph Convolutional Network for the Bottom-up Recognition and Explanation of Events in Video", Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2nd Int. Workshop on Large Scale Holistic Video Understanding (HVU) 2021, Zenodo link: https://zenodo.org/record/4963588

4.5. Multi-ML eXplainable AI Tool

Contributing partners: 3IA-UCA

4.5.1. Overview

Decision-making Management is a growing field dealing with companies operation to "discover, model, analyze, measure, improve, optimize, and automate decisions" and it can nowadays benefit from advances in technology and artificial intelligence. However, the massive adoption of AI in decision automation is hindered by mistrust and risk-aversion [40]. This is mainly due to the lack of explanation of model predictions, which makes the results difficult to understand for business users, who are typically experts in their domain, but are not data scientists: this is the "Last Mile Problem" [41] in data science.

State-of-the-art models are actually *black-box* and even a data scientist could not explain why a decision-making system made a specific decision. In addition to the complexity of machine learning models, in practice companies rarely make decisions using a single model. Instead, the concrete reality of business decision-making services is that of a collection of models (typically two or three), each predicting key quantities for the problem at hand, which are then agglomerated by a decision tree to produce the final decision. In this context, it is crucial for business users to understand why a certain decision has been made and, possibly, to have information simple enough to be able to provide explanations to the final user in a way she or he can understand.

Our method provides local post-hoc explanations for such decision-making systems, based on feature importance and sensitivity analysis.

4.5.2. Methodology

The intuition behind this approach is that we need to agglomerate the explanations of individual models with a mechanism similar to that in which their outputs are agglomerated. Making the appropriate assumptions, we can see a decision-making system as a decision tree where the nodes are machine learning models.

A first distinction of interpretability methods [42] is between:

- *Intrinsically interpretable methods*, referring to machine learning models that are interpretable due to their simple structure, such as linear regression or decision trees;
- *Post-hoc methods*, analyzing the model after training. LIME [43] and SHAP [44] are examples of those.

We therefore combine an ad-hoc method for the interpretability of the (intrinsically interpretable) decision tree, with a model-agnostic method for the machine learning components. To do so, we first perform two parallel steps:

- Explain the decision rules: we obtain the importance of each variable directly involved in the system policies through a geometric approach extending the work of [45] for the explainability of the decision tree;
- Explain the results of the models: get the importance of each input feature for each machine learning model through a post-hoc interpretability method.

Then, we combine these partial explanations in order to fairly attribute them to the input features by weighing the importance of the features in a model by the importance of the output of that model in the decision rules.

4.5.3. Results

We are implementing this method in a Python library. Our tool is designed to support developers of decision-making systems to gain insights into how the system works, and for operators to gain explanations to support the decision, so that they can assess whether or not to trust a decision in an informed way and if required report simple explanations to the final user under review.

Experiments with benchmark data are promising. In particular, we show that by considering the whole decision-making system as a whole black-box model, post-hoc methods produce poor results. In this configuration, there is indeed knowledge that we can exploit: we know which variables are involved in the decision policy and we know its rules. It is therefore worth exploiting this knowledge. When we do so, we are able to provide a full explanation for a complex system combining different ML approaches.

4.6. Exploring Generative Models Latent Spaces

Contributing partners: CEA

4.6.1. Overview

Generative models learn the distribution induced by the samples (images) used at training. Hence, the training images are considered as an independent and identically distributed (i.i.d) samples from an unknown distribution that has been estimated by the generative models. At inference, the generated images thus look similar to the training images but are not strictly identical to any of them [46, 47, 48]. However, if the images of the training set have some particular features of interest, one expects to be able to infer any combination of these, even if the wanted combination is not actually present in the training dataset. For example, if there are green cars and trucks in the dataset as well as red cars, one would expect to be able to generate red trucks. In an ideal case, we want the generative model, and its latent space in particular, to allow vector arithmetic on the features of interest, similar to that met with the word embeddings in natural language processing [49]: "red trucks" = "red cars" - "green cars" + "green trucks". That is, one wants to identify some axes in the latent space that correspond to the features of interest, then sample the latent code in the region at the intersection of what the user wants. In addition to providing mode control over data generation tasks, the ability to do so can provide direct cues to explain how the neural generative model produce some new images. In the example above, one would estimate the axis "red *vs* green" and the axis "cars *vs* trucks" from the training data, then expect that the translation of the second axis in the "red trucks" makes sense. It would be obvious in a vectorial space, but unfortunately, the latent space of generative models seems usually highly non-linear.

Controlling data generation thus consists in being able to explore the latent space of the generative models, to determine the regions that may correspond to features of interest for the user. The ability to do so can thereafter be used as a basis for explainable generational models. In addition, one may want to adapt the training process or generative model itself to get a better *disentanglement*, meaning that a user could modify an attribute while keeping the other constant. A challenge for this task is to minimize the need in annotated data [46], since training data for generative models do not need to be annotated and that the annotation is well-known to be a costly process.

4.6.2. Methodology

The method developed in this task will thus be able to identify such directions of interest and learn latent spaces that allow more disentangled directions. Given a training dataset and a list of features of interest, we expect that a user will be able to edit any new image (quite similar to the training ones) and change the feature of interest of its choice without modifying the others. Such an exploration of the latent space will thus provide direct cues to explain how the neural generative model produces some new images. Indeed, in an ideal case, the method will provide several axes in the latent space that are associated to an explicit semantics, that is to say that carry a direct meaning for a human. Hence, if a given point of the latent space is projected on these axes, its coordinates would correspond to a quantification of each meaning over each axis. Through such a

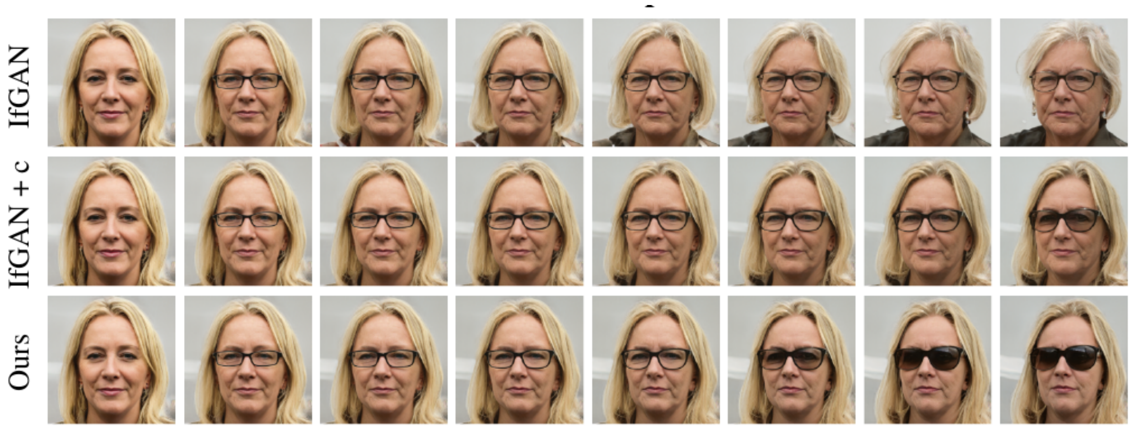


Figure 13. A comparison of our method to InterfaceGAN for the attribute "glasses", controlled in the $\mathcal{Z}$ latent space of StyleGAN trained on FFHQ.

method any point of the latent space will thus be explainable in terms of the semantics associated to the axes.

In the vein of the recent literature [47], we proposed a learning-based supervised method that consists on sampling a set of latent codes, then labelling the latent codes from the corresponding images using pre-trained image classifiers and finally, extracting the directions. As GANs learn to approximate the real data distribution that carries different kinds of biases, the sampling stage also leads to generating biased datasets that can, in turn, affect the semantic directions. In particular, if some attributes are strongly correlated in the training dataset, it will result into a corresponding entanglement in the estimated directions. A classical example for images of faces is the fact that the age is often correlated to the presence of glasses, since older people tend to more often carry glasses than young ones in the usual datasets. The core of our proposal thus consists in modifying this second stage of sampling in order to fix the intrinsic bias carried in the training dataset. Getting latent codes that are unbiased allows to estimate directions that are less entangled, since these codes are the only information used to estimate the directions. Indeed, the third stage is usually implemented as a linear classifier trained to separate latent codes corresponding to images with a desired attribute (positive samples) from those corresponding to images without the desired attribute (negative samples). The direction controlling the attribute is then taken as the vector orthogonal to the classifier's decision boundary. In the literature, a support vector machine (SVM) with a linear kernel is usually used for this stage, but we showed that a simpler method can lead to slighly better results. Indeed, a stronger regularization (larger SVM margin) tends to produce directions that allow more disentangled edits. If the linear SVM has a very large margin, the decision boundary becomes orthogonal to the line connecting the centroids of the two classes. Hence such a direction, that is faster to compute, can also be used directly.

4.6.3. Results

First results of the proposed method is illustrated in Figure 13. This example focuses on the control of the attribute "glasses", that is to say the ability to add/remove glasses to a human face, while avoiding to change other attributes such the age and the gender of the person, or the fact that the person smiles or not. Following the recent literature, we used a StyleGAN neural network trained on the FFHQ dataset [55], and applied our method to the $\mathcal{Z}$ latent space. Similar results are obtained in the $\mathcal{W}$ space, as well as in the $\mathcal{Z}$ latent space of a PGGAN [56] pre-trained on the CelebAHQ dataset [57]. On the first line of Figure 13, the InterfaceGAN method [47] produces a face image with a controlled presence of glasses, there is an obvious entanglement with the attribute “age”, reflecting that in the training dataset, glasses are usually worn by older persons. It can be fixed with an ad-hoc post-processing called conditional manipulation (second row) consisting in forcing orthogonality of the found directions in the latent space. Our method (third line) provides similar results without such artifact, providing interesting insights on the nature of entanglement.

4.6.4. Relevant Resources and Publications

Relevant Publications:

- P. Doubinsky , N.Audebert , M. Crucianu, H. Le Borgne, ”Multi-attribute balanced sampling for disentangled GAN controls”, (2021) (Submitted)

5. Privacy Preserving AI Toolset

Data is the new oil. Never before, so much personal data has been collected and evaluated. Never before, so many technologies are available to analyze the data and combine this into new insights.

All these advances in AI have the important downside that breaching individuals' privacy at scale is also as easy as never before. The European legislation reacted with the General Data Protection Regulation (GDPR) regulating what is allowed and what not. However, this suggests a trade off between AI performance and privacy. But instead of drawing things black and white, making data privacy a natural enemy of progress, it is important to take a look on technologies that allow the processing of personal data without sacrificing sensitive information held by individuals and organisations. More often than not, cleverly anonymised data is enough.

Within this task (T4.4) we create tools that help protecting private data, while making data analysis required by the AI4Media use cases possible.

5.1. Overview of Privacy AI contributions

A common technique to model user and usage data is using graphs, connecting user and item nodes with edges. For example, if different users watched the same movie, this can be modeled by a graph connecting those users to a single movie node. This data structure can also be fed directly into a machine learning model using Graph Neural Networks. In Section 5.2, IDIAP provides a tool to secure privacy within such neural networks.

Differential Privacy is a popular approach to mathematically guarantee data privacy. However, beyond the academic literature there is a lack of software libraries that provide robust and easy to use implementations of this technique. In Section 5.3, IBM describes DiffPrivLib, an open source python library to overcome these issues.

Federated Learning is an emerging technology that allows the training of neural networks without having to aggregate the training data of all participants at a single server / machine; instead, model updates are calculated locally, and only these updates are shared with the central entity. This can help address concerns regarding unintended exposure of private or otherwise sensitive data, but only if Federated Learning is complemented with other tools that prevent attacks gaining knowledge about participants based on shared information. In Section 5.4, FhG-IDMT describes techniques to secure Federated Learning, including e.g. Full Homomorphic Encryption.

Adversarial machine learning describes the process of altering content in a way that there is no difference to the human observer, but the output of a machine learning model gets manipulated. In Section 5.5, AUTH proposes an adversarial attack inspired by K-anonymity that triggers misclassifications by a neural network, therefore protecting the data to be analyzed, but still retains human utility.

5.2. Privacy-Preserving Graph Neural Networks

Contributing partners: IDIAP

5.2.1. Overview

GNNs have shown superior performance in solving the problems formulated as a machine learning task over graphs, such as node classification, link prediction, and graph classification, in various disciplines from social network analysis and recommendation services to drug discovery and medical diagnosis. However, the graphs used to train such models could be sensitive and contain personal information, and this information can be leaked through the model's output, when the GNN is released publicly, or when it is offered as a service [58, 59, 60]. For example, a GNN trained over a social network for friendship recommendation may reveal the graph's linkage information through its predictions. As another example, a GNN trained over the social graph of COVID-19 patients to predict the spread of the disease could be used as a service by government authorities, but an adversary might be able to recover the private graph used for training.

This research aims to prevent the information leakage of the underlying graph in GNNs using privacy-enhancing technologies, such as Differential Privacy (DP). As the first step, we used local differential privacy to protect the privacy of node-level data, such as node features and node labels, when accessed by untrusted third-parties, such as social networks, who already have the linkage information but require node features/labels to train a GNN. In the next step, our goal will be to propose a privacy-preserving GNN framework based on the specific notions of DP for graphs, such as edge-DP, to also guarantee the indistinguishability of individual edges of the graph by trying to keep the GNN's output distribution nearly the same when an arbitrary edge is added to or removed from the graph.

5.2.2. Methodology

In order to preserve the privacy of node data, we proposed the Locally Private Graph Neural Network (LPGNN), a novel privacy-preserving GNN learning framework for training GNN models with private node data. Our method has provable privacy guarantees based on Local Differential Privacy (LDP), can be used when either or both node features and labels are private, and can be combined with any GNN architecture independently.

To protect the privacy of node features, we proposed an LDP mechanism, called the *multi-bit mechanism*, through which the graph nodes can perturb their features that are then collected by the server with minimum communication overhead. These noisy features are then used to estimate the first graph convolution layer of the GNN. Given that graph convolution layers initially aggregate node features before passing them through non-linear activation functions, we benefit from this aggregation step as a denoising mechanism to average out the differentially private noise we have injected into the node features. To further improve the effectiveness of this denoising process and increase the estimation accuracy of the graph convolution, we prepend a simple yet effective graph convolution layer based on the multi-hop aggregation of node features, called KProp, to the backbone GNN.

Finally, since the node labels are also considered private, we need another LDP mechanism to collect them privately. To this end, we use the generalized randomized response algorithm [61], which randomly flips the correct label to another one with a probability that depends on the privacy budget. However, learning with perturbed labels introduces extra challenges, as the label noise could significantly degrade the generalization performance of the GNN. To this end, we propose a robust training framework, called Drop (label **d**enoising with p**rop**agation), which incorporates the graph structure for label correction, and at the same time does not rely on any form of clean data (features or labels), neither for training nor validation. Given that nodes with similar labels tend to connect together more often, we utilize the graph topology to predict the label of a node by estimating the label frequency of its neighboring nodes. Still, if we rely on immediate neighbors, the true labels could not be accurately estimated due to insufficient neighbors for many nodes. Again, our key idea is to exploit KProp's denoising capability, but this time on node labels, to estimate the label frequency for each node and recover the true label by choosing the most frequent one.

The overview of our framework is depicted in Figure 14.

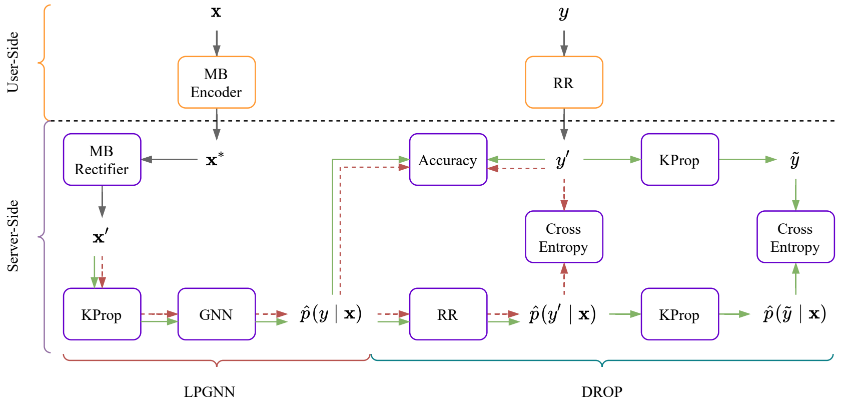


Figure 14. Overview of the locally private GNN training framework, featuring the multi-bit mechanism (MB Encoder and MB Rectifier), randomized response (RR), KProp layers, and Drop training. Users run multi-bit encoder and randomized response on their private features and labels, respectively, and send the output to the server, after which training begins. Green solid arrows and red dashed arrows indicate the training and validation paths, respectively.

5.2.3. Results

We evaluated how our privacy-preserving LPGNN method performs under varying feature and label privacy budgets. We changed the feature privacy budget ϵ_x in $\{0.01, 0.1, 1, 2, 3, \infty\}$ and the label privacy budget ϵ_y within $\{1, 2, 3, \infty\}$. The cases where $\epsilon_x = \infty$ or $\epsilon_y = \infty$, are provided for comparison with non-private baselines, where we did not apply the corresponding LDP mechanism (multi-bit for features and randomized response for labels) and directly used the clean (non-perturbed) values. We performed this experiment using GCN, Graph Attention Network (GAT), and GraphSAGE as different backbone GNN models and reported the node-classification accuracy in Figure 15.

We can observe that all the three GNN models demonstrate robustness to the perturbations, especially on features, and perform comparably to the non-private baselines. For instance, on the Cora dataset [9], both GCN and GraphSAGE could get an accuracy of about 80% at $\epsilon_x = 0.1$ and $\epsilon_y = 2$, which is only 6% lower than the non-private ($\epsilon = \infty$) method. On the other three datasets, we can decrease ϵ_x to 0.01 and ϵ_y to 1, and still get less than 10% accuracy loss compared to the non-private baseline. We believe that this is a very promising result, especially for a locally private model perturbing hundreds of features with a low privacy loss.

5.2.4. Relevant Resources and Publications

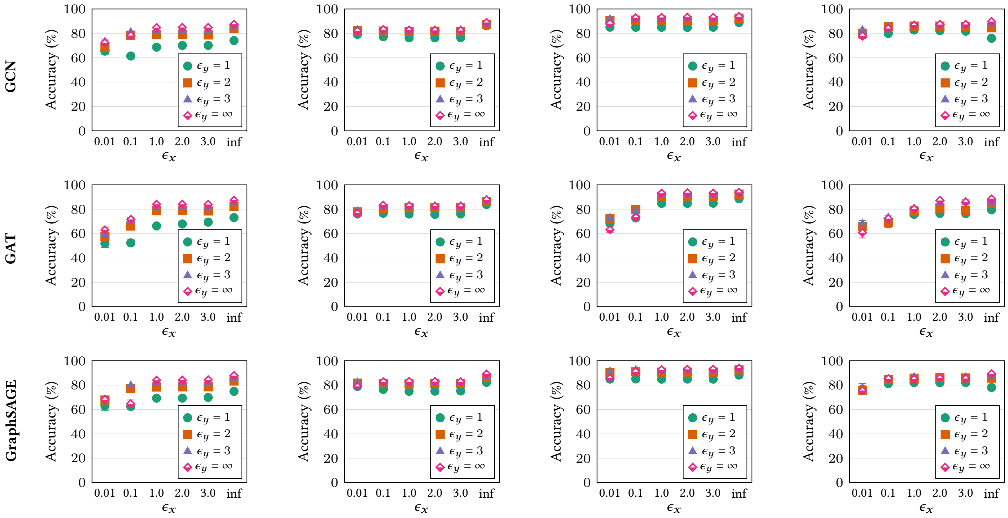
Relevant Resources:

- **Privacy Preserving AI Toolset Repository:** https://github.com/sisaman/LPGNN

Relevant Publications:

- S. Sajadmanesh, D. Gatica-Perez, "Locally Private Graph Neural Networks." Conference in Computer and Communication Security (CCS) (2021) Zenodo link: https://zenodo.org/record/5081878

[9] https://graphsandnetworks.com/the-cora-dataset/



(a) Cora **(b) Pubmed** **(c) Facebook** **(d) LastFM**

Figure 15. Comparison of LPGNN's performance with different GNN models under varying feature and label privacy budgets.

5.3. DiffPrivLib - A Differential Privacy Library

Contributing partners: IBM

5.3.1. Overview

Owing to its robust mathematical guarantees, generalised applicability and rich body of literature, differential privacy has emerged as the defacto standard in data privacy. Over the last 10 years, mechanisms have been investigated to optimise the process of achieving differential privacy in an ever-widening field of topics with various data types and scenarios.

However, in most cases, such research has so far led to the creation of only disparate code bases. This requires the community to tediously combine various heterogeneous implementations of differential privacy mechanisms on an ad-hoc basis when applying such techniques in the real world applications.

As a consequence, the DiffPrivLib library [62] was created as a central repository of differential privacy mechanisms readily available to apply and combine in various application use cases, in conjunction with state of the art standard machine learning practices and tools. The aim is to enable users to i) experiment with differential privacy, ii) explore the impact such techniques can have on machine learning accuracy and iii) build commercial grade applications with differential privacy mechanism integrated from their inception onwards.

DiffPrivLib is a general purpose, open source python library, which includes a host of mechanisms, the building blocks of differential privacy, alongside a number of applications to machine learning and other data analytics tasks.

5.3.2. Methodology

Simplicity and accessibility has been prioritised in developing the library, making it suitable to a wide audience of users, from those using the library for their first investigations in data privacy, to the privacy experts looking to contribute their own models and mechanisms for others to use.

The Diffprivlib library is comprised of four major components namely i) Mechanisms, ii) Models, iii) Tools and iv) Accountants. DiffPrivLib mechanisms (e.g.: Laplace, Gaussian, Exponential functions etc.) consist of differential privacy building blocks which can be integrated directly within various machine learning models. Since regular machine learning models do not support differential privacy mechanisms by default, the models offered by DiffPrivLib consist of standard machine learning models (e.g.: Logistic Regression, KMeans etc.) engineered from their inception with differential privacy capabilities in mind. These include clustering, classification, regression and dimensionality reduction and pre-processing models. The library also provides users with various generic tools for differentially private data analysis. These include histograms functions as well as simple statistical functions both mirroring and leveraging the functionality of their NumPy counterparts with differential privacy. Finally, an accountant module is also provided which can be used to track privacy budget and privacy loss while combining the various mechanisms available.

5.3.3. Results

As an example of DiffPrivLib capabilities, we present the results of differential privacy being applied to a naive Bayes classifier. Naive Bayes is a probabilistic classifier that learns the means and variances of each feature (assumed independent) for each label, allowing Bayes theorem to be applied to classify unseen examples. Differential privacy is applied by adding noise to the means and variances that the model learns, ensuring a decoupling of the model from the data upon which it was trained.

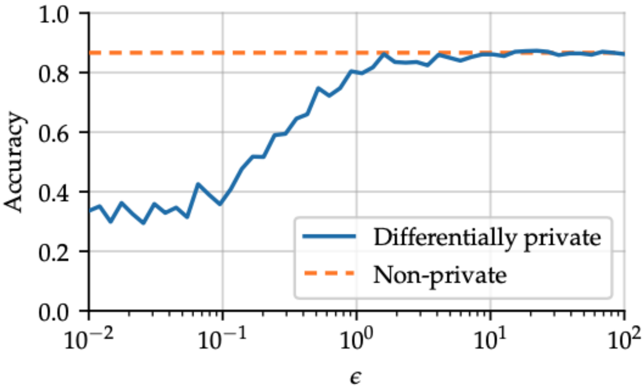


Figure 16. Comparison of accuracy versus ϵ for a differentially private naive Bayes classifier on the Iris dataset.

As can be seen in Figure 16, DiffPrivLib can be used to assess accuracy levels of our Naive Bayes model with respect to various values of ϵ privacy guarantees. In this example, the model was trained upon the Iris flower dataset [10],which results in high threshold values for ϵ needed in order to maintain accuracy. The DiffPrivLib library developed as part of this deliverable builds on top of existing work carried out prior to the AI4Media project. In particular, on-going contributions include the implementation of mechanisms which deal with statistical attacks that differential privacy

[10] https://www.kaggle.com/arshid/iris-flower-dataset

mechanisms are known to be vulnerable to. These vulnerabilities are caused by the approximation of real values to floating point numbers.

Our current investigation focuses on practical solutions to the finite-precision floating point vulnerability, where the inverse transform sampling of the Laplace distribution can itself be inverted, thus enabling an attack where the original value can be retrieved with non-negligible advantage. Such a solution would the advantage of being (i) mathematically sound, (ii) generalisable to any infinitely divisible probability distribution, and (iii) simple implementation in modern architectures. Additional, this solution would have the added benefit of making side channel attack infeasible, due to the inherently exponential domain sizes of brute force attacks.

5.3.4. Relevant Resources and Publications

Relevant Resources:

- **DiffPrivLib Repository:** https://github.com/IBM/differential-privacy-library

5.4. Tools for secure federated learning

Contributing partners: FhG-IDMT

5.4.1. Overview

Usually, a machine learning model is trained at a central server, with all training data being aggregated from participants / clients prior to the training process. From the perspective of model performance, this approach is ideal, but depending on the application, it can come with serious drawbacks: Providing all training data to a central entity may come with significant cost (think of big media archives and large amounts of data, or small sensor devices with little bandwidth). More importantly, participants may not want to share their data with a central entity at all, due to privacy and confidentiality considerations.

The idea of federated learning is that the training data stays with the participants / clients, and only the model weights are shared: Every client performs a local training, resulting in a local model update. Then, all models are sent to the central server, and the aggregate model is sent back to the clients. While this approach is likely to come with a decrease in performance, ideally, it still performs well, and all clients benefit from each other without having the need to transmit their training data to other actors.

While Federated learning is very helpful when it comes to data reduction, it is however not guaranteed that it avoids privacy and confidentiality concerns, because the trained models can still reveal a lot of information about the clients and their training data. The original federated learning paper mentions this in a footnote [63, p. 2]: "For example, if the update is the total gradient of the loss on all of the local data, and the features are a sparse bag-of-words, then the non-zero gradients reveal exactly which words the user has entered on the device. In contrast, the sum of many gradients for a dense model such as a CNN offers a harder target for attackers seeking information about individual training instances (though attacks are still possible)." Not surprisingly, reconstruction attacks on CNNs turned out to be feasible and effective [64].

Hence, it is clear that there is a need for securing federated learning, and depending on the given use case and attacker model, there are several candidates technologies regarding privacy enhancement technologies for federated learning:

- Differential Privacy

- Fully Homomorphic Encryption (FHE)
- Secure Multiparty Computation (SMPC)

The secure federated learning tool will consist of a set of modules that allows incorporating selected PET in federated learning frameworks.

5.4.2. Methodology

Depending on the use case, it is important to specify attacker models and define the level of trust put into other participants, and select appropriate security measures based on that. For instance, there might be scenarios where only the (cloud) server is untrusted, while other scenarios will also require protecting against other clients. A basic FHE scheme where all clients share the same key can prevent the server to learn anything on the model data, thereby addressing the former type of scenario, but it will not prevent a malicious client to spy on others, which is necessary for the latter scenario.

In the domain of federated learning, the notion of an *honest, but curious* attacker is common, emphasizing the privacy aspect. Participants in the federated learning systems are suspected to get as much out of the data as they can (or lose them after being hacked), but to do so *passively*. In contrast to an *active* attacker, the aggregator can run model inversion attacks on the individual model updates it receives, but it will not send faulty data to individual clients, which could compromise the overall system performance.

From all possible participant / attack mitigation combinations, we will start with differential and FHE for Federated Averaging: Differential Privacy (DP) can be applied to the training data directly but DP can also be applied to the resulting model weights of the local training *before* sending it to the aggregator, which is specially useful within the AI4Media context, and will therefore be supported by the tool. This will include investigation of the trade-off between performance degradation and resilience regarding model inversion attacks.

The benefits of FHE come with significant cost in terms of computing time and memory requirements (by orders of magnitudes). This makes direct neural network computations, e.g. encrypted inference, infeasible for many problems relevant for AI4Media. However, a central part of Federated Learning is the computation of the aggregated model, which can be as simple as calculating an average. These operations are a good fit for FHE, and support will be provided by the tool for popular Federated Learning (FL) frameworks like flower[11]. Practical security issues like key exchange and modifications to the Federated Averaging will be investigated as well.

5.4.3. Results

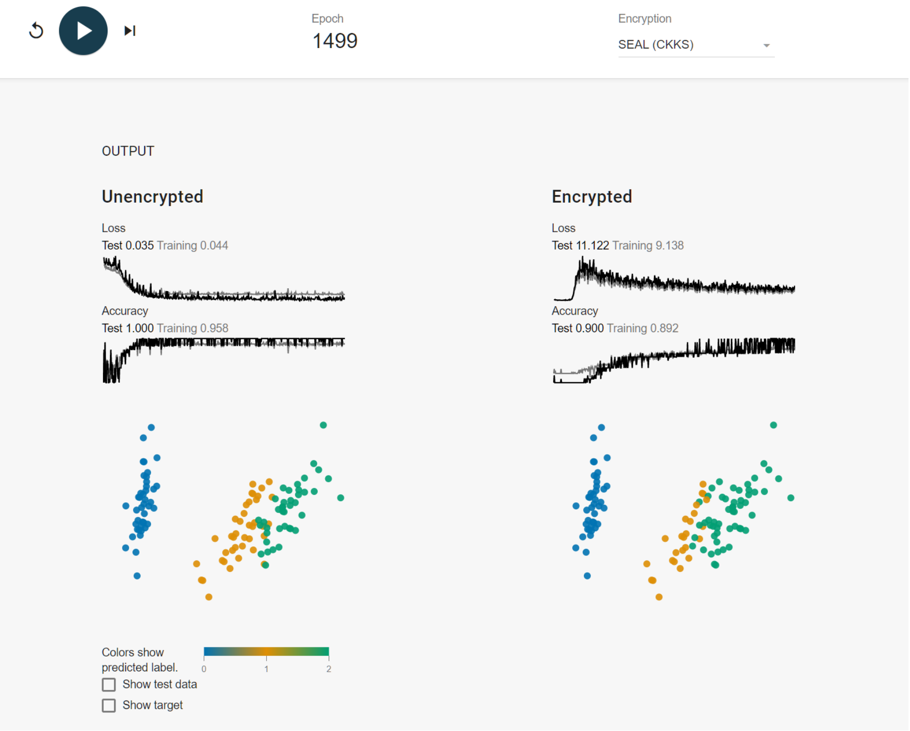
First experiments with FHE for the federated averaging turned out to be promising. Furthermore, we created a proof-of-concept of machine learning in the encrypted domain by implementing encrypted inference for the (very simple) classification task of the IRIS dataset. See Figure 17 and 18 for some details.

5.4.4. Relevant Resources and Publications

Relevant Resources:

- **TenSEAL Python library for FHE:** https://github.com/OpenMined/TenSEAL
- **Flower Federated Learning Framework:** https://flower.dev

[11] https://flower.dev



Epoch 1499

Encryption SEAL (CKKS)

OUTPUT

Unencrypted

Loss
Test 0.035 Training 0.044

Accuracy
Test 1.000 Training 0.958

Encrypted

Loss
Test 11.122 Training 9.138

Accuracy
Test 0.900 Training 0.892

Colors show predicted label.

☐ Show test data

☐ Show target

Figure 17. A screenshot of our demo application for encrypted inference. Complex FHE operations add noise to the data, which explains why the encrypted model converges slower than the unencrypted one

5.5. Introducing K-anonymity principles to adversarial attacks for privacy protection in image classification problems

Contributing partners: AUTH

5.5.1. Overview

The network output activation values for a given input can be employed to produce a sorted ranking. Adversarial attacks typically generate the least amount of perturbation required to change the classifier label. In that sense, generated adversarial attack perturbation only affects the output in the 1st sorted ranking position. We argue that meaningful information about the adversarial examples i.e., their original labels, is still encoded in the network output ranking and could potentially be extracted, using rule-based reasoning. To this end, we introduce a novel adversarial attack methodology inspired by the K-anonymity principles, that generates adversarial examples that are not only misclassified by the neural network classifier, but are uniformly spread along K different positions in the output sorted ranking. This tool can thereby be employed for privacy protection against automated analysis tools, while maintaining human utility (e.g., for de-identifying personal photos in social media).

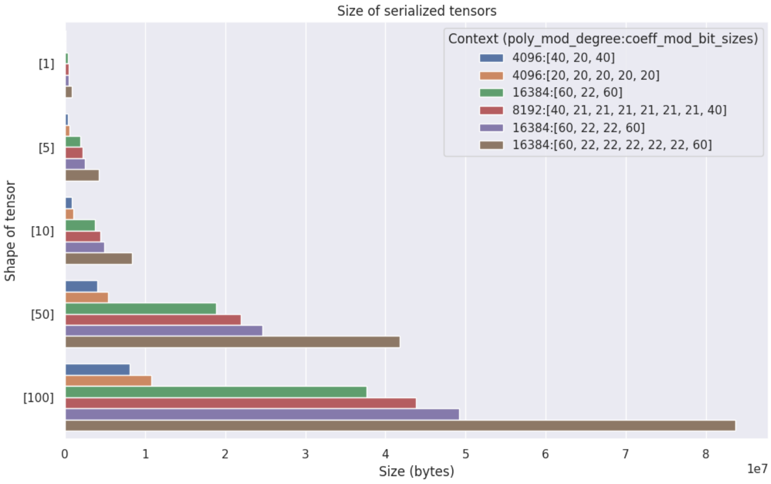


Figure 18. The size increase of encrypted tensors depending on the used FHE settings, in this case parameters of the CKKS FHE scheme.

5.5.2. Methodology

K-anonymity is a generic privacy protection concept that suggests that the maximum probability of identifying an individual in a specific set must be lower than $1/K$ [65]. In order to quantify privacy protection introduced by adversarial attacks according to the K-anonymity principles, one might employ the class identification probabilities (e.g., classification rate) of a set of adversarial examples $\tilde{\mathcal{X}}$ produced by some attack against the classifier decision function $f(\tilde{\mathbf{x}};\boldsymbol{\theta}) = y$ that has been attacked. However, such a definition does not examine the overall neural network output activation values. More specifically, according to the perspectives of Label Ranking [66] and Multi-Label Classification [67], the network output activation values contain an underlying strict ordered ranking over the finite label set $\mathcal{Y} = \{\ell_i, \dots, \ell_C\}$, where C is the total number of classes supported by the model and $\ell_i \succ_{\mathbf{x}} \ell_j$ denotes that for a given data example $\mathbf{x}$, label ℓ_i is a more probable output classification label than label ℓ_j. For simplicity reasons, we denote the output ranking with a vector function $\mathbf{r}(\mathbf{x}) = [r_{\mathbf{x}}(1), \dots, r_{\mathbf{x}}(C)]^T$, such that the output classification label of sample $\mathbf{x}$ by the deep neural network model is given in the 1st ranking position $r_{\mathbf{x}}(1)$. We argue that the ranking obtained for any sample $\mathbf{x}$ may encode underlying data properties, that may expose information about the class of interest (e.g., we assume that in most cases, the true label of misclassified samples may be obtained by $r_{\mathbf{x}}(2)$).

Taking the above into consideration, we design an adversarial attack methodology in order to preserve anonymity (hide the true label) of every sample $\mathbf{x} \in \mathcal{X}$ against the neural network according to the K-anonymity constraints, taking into consideration the whole network output layer. To this end, we argue that the appropriate mapping $\mathcal{X} \mapsto \tilde{\mathcal{X}}$ should achieve two conditions:

$$r_{\tilde{\mathbf{x}}}(1) \neq y, \quad \forall \tilde{\mathbf{x}} \in \tilde{\mathcal{X}}, \tag{11}$$

$$p(i) = \begin{cases} P\left(r_{\tilde{\mathbf{x}}}(i) = y\right) \leq 1/K & , \forall i \in \{1, \dots, C\} \\ 0 & , \text{otherwise}, \end{cases} \tag{12}$$

where $p(\cdot)$ is the probability mass function of its argument, containing the probability of retrieving the true label of the adversarial example $\tilde{\mathbf{x}}$ in the $i-$th position of the output ranking, e.g., $P\left(r_{\tilde{\mathbf{x}}}(1)\right)$ is equal to the classification rate of the model. K is a hyperparameter denoting the $K-$anonymity protection level, e.g., 5-Anonymity. Condition (11) is the adversarial attack objective, i.e., disabling correct classification. To this end, we demand that the true labels of exactly K data groups, each containing N/K samples of $\tilde{\mathcal{X}}$, are not retrieved in at least $k \in \mathcal{K} = \{2, \ldots, K+1\}$ sorted ranking positions, relevant to K. Assuming e.g., $K = 5$, then 5 such data groups must be formed, demanding that the true labels of the first group are not retrieved in ranking positions $r_{\mathbf{x}}(1), r_{\mathbf{x}}(2)$, while the labels in the 5-th group are not retrieved in any position of $r_{\mathbf{x}}(i), \forall i \leq 6$.

The proposed methodology is designed to achieve more difficult constraints when compared to standard adversarial attacks. It should be expected that increased perturbation may be generated to the crafted adversarial examples. To this end, We introduce a visual similarity term to our proposed optimization problem, namely the Complex Wavelet Structural Similarity (CW-SSIM) loss function [68] $s(\mathbf{x}, \tilde{\mathbf{x}})$ between the initial samples and the crafted adversarial examples, guiding the optimization problem towards solutions that regulate the amount of perturbation generated by the adversarial attack. Maintaining the assumptions of white-box attacks i.e., access to a continuous loss function L_f associated with f, we propose the following optimization problem:

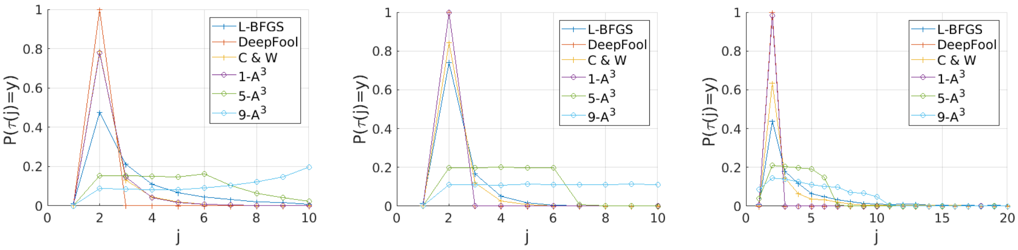
$$\min_{\mathbf{p}}: \quad \|\mathbf{p}\|_2 + (1 - s(\mathbf{x}, \tilde{\mathbf{x}})) + \sum_{i=2}^{k} L_f(f(\tilde{\mathbf{x}}; \boldsymbol{\theta}), r_{\mathbf{x}}(i)), \tag{13}$$

until the ranking obtained for $\tilde{\mathbf{x}}$ by the neural network architecture satisfies the constraint $r_{\tilde{x}}(i) \succ_{\tilde{\mathbf{x}}} y, \forall i \in \mathcal{K}$.

5.5.3. Results

In order to evaluate the performance of the proposed method, we have employed the MNIST (digit classification), CIFAR-10 (object recognition) and Yale (face recognition) datasets. A CNN architecture, namely the LeNet5 (MNIST-LeNet) [69] was trained from scratch in the MNIST dataset. In the CIFAR-10 dataset, we have trained the MobileNetV2 architecture [70] (CIFAR-10-MobileNetV2). In the Yale dataset, we fine-tuned a 9-Layer LightCNN architecture [71], that had been pre-trained using more than 1.5M facial images from CelebA dataset (Yale-LightCNN), totaling 3 architecture-dataset combinations. All conducted experiments were implemented using PyTorch. The proposed method was employed to attack each architecture for different values of $K = 1, 5, 9$. For comparison reasons, we have also employed the L-BFGS [72], DeepFool [73] and the C & W [74] attack with L_2 distance.

The datasets obtained by each method were evaluated in terms of satisfying K-anonymity principles, as have been defined by equation (12). That is, we have tried to retrieve the original dataset labels using the architecture $\boldsymbol{\theta}$ to obtain ranked label outputs for each adversarial sample. We have determined the probability mass functions $p(i)$ for obtaining the ground truth label at the $i-$th ranking position, plotted in Figure 19. As can be observed, the datasets obtained by employing L-BFGS, DeepFool, C & W, and the variant of the proposed method with $K = 1$ do not satisfy the K-anonymity requirements, since $P(r_{\tilde{\mathbf{x}}}(2) = y) > 1/K$ for every $K > 1$. On the other hand, the probabilities of the adversarial examples crafted by the proposed method with $K = 5$ and $K = 9$, satisfy $P(r_{\tilde{\mathbf{x}}}(j) = y) \leq 1/5$ for $i = 2, \ldots, 6$ and $P(r_{\tilde{\mathbf{x}}}(j) = y) \leq 1/9$ for $j = 2, \ldots, 10$ respectively in almost every case, or lie really close.



(a) MNIST *(b) CIFAR* *(c) Yale*

Figure 19. Probability mass functions of recovering the original labels y in the j-th sorted ranking position $P(r_{\tilde{\mathbf{x}}}(j) = y)$ generated by L-BFGS, DeepFool, C & W, and the proposed methods. L-BFGS, DeepFool, C & W, and the variant 1-A^3 of the proposed method do not satisfy the K-anonymity requirements, since in most cases, the original label y can be recovered by retrieving the label ranked 2nd, in constrast with the variants 5-A^3 and 9-A^3.

5.5.4. Relevant Resources and Publications

Relevant Publications:

- V. Mygdalis, A. Tefas, I. Pitas, ”Introducing K-anonymity principles to adversarial attacks for privacy protection in image classification problems”, Technical Report, (submitted as conference paper), Zenodo link: https://zenodo.org/record/5137317#.YP6R_Y4zabg

6. Fair AI Toolset

As machine learning models are fast becoming critical components of every decision making process essential for our society (mortgage lending, prison sentencing etc), it becomes crucial to guarantee that these models do not privilege specific groups or individuals at the disadvantage of others. These models are constructed upon the statistical analysis and properties of training data, which may contain biases due to existing prejudice and/or inaccurate sampling. Hence, if left unchecked unwanted biases can emerge from these models with significant societal consequences.

AI Fairness is typically evaluated either on a group or individual level. When addressing group fairness, a population is divided into groups based on a set of protected attributes (gender, ethnicity etc.). A fair ML model within this context is a model which seeks some statistical measure to be equal across such groups. On the other hand, when addressing individual fairness, ML models seek to treat individuals similarly regardless of their protected attributes.

Algorithms and metrics designed to address biases in ML models can operate on the training data itself as well as on the trained model. Moreover, they can also occur at various points in the machine learning lifecycle whether at a pre-processing, in-processing or post-processing phase. This work-package seeks to apply AI fairness algorithms and metrics at a group and individual levels and at various points in the AI lifecycle.

6.1. Overview of our AI Fairness contribution

In this section, we present the work carried out within the dimension of AI Fairness. As pointed out in Section 2.2, the contributions presented in this section consists of preliminary work which will be used as a foundation for the next deliverable. In particular, an initial research requirement analysis and AI4Media-wide workshop activities are presented in Section 6.2.

6.2. AI Fairness 360 Toolkit

Contributing partners: IBM

6.2.1. Overview

As machine learning models are used to support decision making in high-stakes applications, fairness is fast becoming an increasingly important concern. In order to meet these concerns, an increasing number of algorithms and metrics have been developed within the research community which can either detect, quantity and/or rectify biases introduced by ML models at various point in the AI lifecycle pipeline. Various toolkits [12] and libraries have recently emerged to consolidate these advances in AI Fairness and provide the required tools for the public to scrutinise and/or address any bias in the models used within the industry.

As a preliminary step to this research, we performed a review and analysis of existing frameworks available to the community and assessed them with respect to the upcoming AI4Media use case requirements (see Section 7), as well as the broader community at large.

6.2.2. Methodology

In our attempt to identify an AI Fairness framework suitable as a basis for our contributions to this project, various requirements were derived based on the type of bias to be assessed, as well

[12] Eg: Aequitas: https://github.com/dssg/aequitas, AIF360: https://github.com/Trusted-AI/AIF360, FairLearn: https://github.com/fairlearn/fairlearn

as the various WP8 use cases definition workshop sessions carried out throughout the first year of this project (see Section 7).

Such a framework should first of all provide a wide diversity of state of the art algorithms and metrics capable of analysing and evaluating fairness both at a group and individual level. Ideally, a subset of these algorithms should also provide the ability to correct some of the identified biases. Bias detection should be performed in both the ML models themselves as well as the datasets they are trained on. A subset of these algorithms should be applicable at various points in the lifecycle of the machine learning pipeline. Namely, data scientists should have algorithms at their disposal capable of detecting and correcting bias at the pre-processing, in-processing or post-processing phases of the pipeline. Finally, following the various use case definition workshops carried out in collaboration with WP8, it became clear that the output of such analysis will need to satisfy the needs of a diverse set of users. In other words, the reports summarising the results of fairness analysis should be accessible not only to data scientists and machine learning experts but also to non technical professionals (eg: journalists, educators etc.). Finally, these reports should provide the ability to interact with the analysis results as a means of exploring the various components leading to biases.

Based on our analysis, the AI Fairness 360 toolkit (AIF360) appears to be the framework which meets most of the requirements list above and which will be the most amenable to extensions and modifications in the following iterations of this contribution.

This toolkit is an extensible open-source library containing techniques developed by the research community to help detect and mitigate bias in machine learning models throughout the AI application lifecycle. It provides the whole spectrum of metrics and state of the art algorithms to mitigate bias in machine learning models at various stages of the AI application lifecycle. Additionally, fairness analysis produced by this framework can be combined with explainer modules, which consist in components leveraging Explainable AI algorithms (e.g. LIME [43]) to provide insights as to how a particular algorithm reached the conclusion that a model was biased. Such a framework could form a reliable basis to facilitate the transition of fairness research algorithms for use in an industrial setting by providing a common framework across existing fairness algorithms, which should assist the research community to share and evaluate algorithms.

The AIF360 toolkit has been engineered as an end-to-end workflow enabling users to start from raw data, gradually developing fair AI models as easily as possible. For this reason, the design of the AIF360 toolkit relies heavily on standard paradigms used in data science. Figure 20 presents the general architecture and workflow supported by AIF360. As can be seen, fairness algorithms can be applied in the various phases of the ML pipeline. Three main paths can be used in order to improve the fairness of AI model predictions, namely fair pre-processing, fair in-processing, and fair post-processing. Each of these corresponds to a category of bias mitigation algorithms implemented in AIF360. In each of these three cases, however, algorithms supported can act on an input dataset and produce an output dataset.

While many other instantiations are also possible, Figure 20 describes one typical generic pipeline. Every output in this process (rectangles in the figure) is a new dataset that shares, at least, the same protected attributes as other datasets in the pipeline. Every transition is a transformation that may modify the features or labels or both between its input and output. Trapezoids represent learned models that can be used to make predictions on test data. In this example pipeline, data is loaded into a dataset object, transformed into a fairer dataset using a fair pre-processing algorithm. A classifier can thereafter be learned from this transformed dataset so as to produce fair predictions. At this point , metrics can be calculated on the original, transformed, and predicted datasets, as well as between the transformed and predicted datasets.

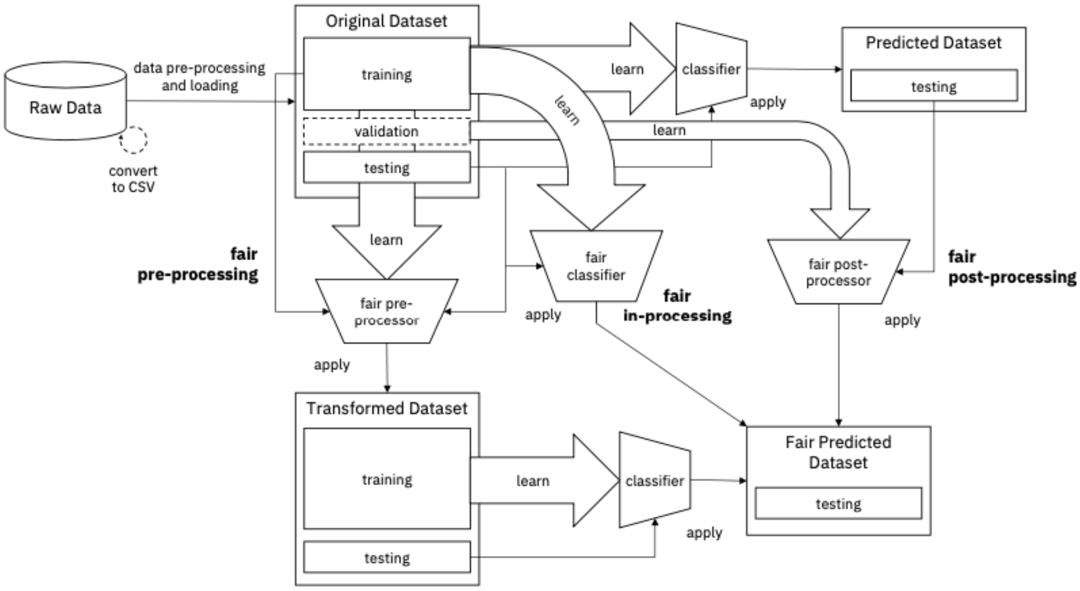


Figure 20. AIF360 Architecture

6.2.3. Results

As part of this preliminary analysis of AI Fairness requirements and frameworks available to the community, the AIF360 toolkit was selected as the most amenable platform to rely on in the next steps of our contribution to AI Fairness.

The toolkit meets most of the requirements set out prior to this analysis. It includes a comprehensive set of fairness metrics for datasets and models, explanations for these metrics, and algorithms to mitigate bias in datasets and models. Despite all of these features however, the toolkit could be enhanced with respect to two important requirements. Although it does provide explainer modules, the reporting produced by this framework is mostly consumable only by data scientists or machine learning engineers. The reporting produced is inadequate to non technical professionals such as journalists, educators etc. An adaptable reporting tool, which could dynamically adjust the level of detail and insight in reports produced by this toolkit could be very beneficial to deliver an understanding of biases within ML models and datasets to the wider community while leveraging the existing reports produced by the framework. Such a module could for example combine reports from this toolkit with the IBM FactSheet initiative [13]. This would enable both technologists such as data scientists to more easily manage the governance and lifecycles of their AI models by leveraging a dynamic reporting tool consumable by a wider set of stakeholders. Finally, while the reporting tools currently produced by AIF360 provide reliable detailed summaries of any biases detected within models and datasets, model probing capabilities are currently missing. Linking existing reporting tools with computational neural graphs would enable data scientists to interactively explore data point boundaries of a given dataset or model produced from within any workflow.

In summary, we plan to use the AIF360 framework as a basis for our on-going collaboration with the relevant WP8 use cases (see Section 7). As it stands, many algorithms contained within

[13] https://www.ibm.com/blogs/research/2020/07/aifactsheets/

this toolkit could already be of use to these use cases. Our intention is to explore the use of these algorithms by AI4Media partners, as part of the WP8 use case vehicles. This will also enable us to explore and develop extensions to the framework to support dynamic reporting and explorative capabilities for the various stakeholders involved in each use case.

In parallel to this analysis and in anticipation to the use of AI Fairness tools across the AI4Media partners, we already dedicated various sessions during plenary presentations to focus on the topic of AI Fairness. The intention was to inform the various AI4Media partners involved in WP8 use cases about the broad principles of AI Fairness and the various technical implications involved when interpreting the various fairness metrics which can be used. An entire session was also used, as part of a WP4 technical workshop event organised by this work-package (see Section 8), for the purpose of AI Fairness in which Samuel Hoffman, the technical lead of the AIF360 toolkit, was invited as an external guest speaker to present the framework to AI4Media partners.

6.2.4. Relevant Resources and Publications

Relevant Resources:

- **AIF360 Toolkit:** https://github.com/Trusted-AI/AIF360

7. Contributions to the AI4Media WP8 Use Cases

Each individual piece of research presented in this deliverable will be made available to WP8 as modules ready to be incorporated, with a view to support various use case scenarios or demonstrators. WP8 operates 7 use cases, which defined their user needs for AI functions in the format of user stories. The user stories (described in detail in deliverable D8.1) [75] were grouped into ”Features” (sub use cases) and ”Epics” (groups of related user stories). Each Epic has its own ID (e.g. Epic 1E4) (see Table 4).

Table 4. WP8 Use case Epics in which modules from WP4 partners will contribute to

WP8 Use Case Epics			
Use Cases	**Features**	**Epic ID**	**Requirement Title**
AI Against Disinformation	Detection/Verification of Synthetic Media	1A2	Synthetic Image Detection/Verification
	Capability for Trustworthy AI	1E2 1E3 1E4	Capability for Explainability Information about Bias Mitigation (Fairness/Accuracy) Information about Robustness (Safety/Security)
AI in Vision	Management of unexpected event occurrence	3C1	Just-In-Time Content Verification
AI for Social Science and Humanity	AI Trustworthiness	4A1 4A3 4A6	Representation in training sets AI Transparency Safety & security
AI for Content Organisation and Moderation	AI Trustworthiness	7C1 7C2	Representation in datasets AI robustness

As part of various WP8 use case mapping sessions, WP4 partners analysed which Epics could benefit from the research conducted across WP4 tasks and related to the Trustworthy AI dimensions. These sessions eventually led to the participation of WP4 partners to two internal user requirement workshop events organised by WP8, in which details about individual use cases and WP4 modules were clarified and aligned. Finally, a dedicated session as part of the WP4 technical workshop (see Section 8) was also used to discuss the various WP4 specific technical coordination required among partners to deliver these modules.

As a result of these meetings, we have identified various Epics in which WP4 partners technical output will be beneficial. For example, the Epics 1E4, 4A6 and 7C2, will benefit from the optimization measures developed by AUTH (see Section 3.2) to avoid adversarial attacks as well as from the extensive set of algorithms to attack and defence AI models available within the Adversarial Robustness Toolkit (see Section 3.3) developed by IBM.

Epics 1A2, 4A3 and 1E2 will be able to benefit from of the capabilities provided by the RCV-tool developed by HES-SO (see Section 4.2) to understand the underlying features that caused the detection of synthetic images and by providing explainability for models with multi-modal inputs. These Epics will also benefit from the Object Graphs method developed by CERTH (see Section 4.4) by enabling users to understand the underlying processes within a video annotation method and how these produce their results on the basis of supplied input data. In addition, the work carried out by CEA with regard to the exploration of Generative Models Latent Spaces (see Section 4.6) will enable these Epics to acquire direct cues to explain how the neural generative model produces some new images.

The AI Fairness 360 toolkit (see Section 6.2) will provide many functionalities to Epics 1E3, 4A1, 7C1, including dataset and models bias detection/correction algorithms as well as various metric tools to measure the extend to which such datasets or models are biased.

Finally, Epic 3C1-1 will benefit from the research modules produced by AUTH as part of their work on K-anonymity principles (see Section 5.5) to de-identify images from automated analysis/detection/classification tools, in a humanly imperceptible manner while maintaining utility of images for human users.

8. Conclusion

This deliverable presented an overview of the current research carried out as part of WP4 focusing on Trustworthy AI. As can be seen in each section of the document, partners have been active in addressing numerous challenges presented across the various dimensions of Trustworthy AI.

Within the dimension of **Adversarial AI**, new methods aimed at increasing the robustness of ML models have been proposed. To achieve this goal, two methods involve the use of novel attacks leveraging excess capacities as well domain shift vulnerabilities common in many AI models. While the latter techniques can be applied to trained models, a third method focuses on strengthening the robustness of models during their training through the use of homeomorphic topological spaces. These techniques have been tested and can be applied on discriminative, generative and re-identification models.

Novel approaches within the field of **Explainable AI** have also been presented. Two techniques enhancing interpretability mechanisms available for deep neural models were proposed. The first enables vector space arithmetic capabilities to be used within the latent space of generational models, while the second provides the ability to interpret high level concepts of internal features present within deep neural networks by monitoring their changes with respect to a given input. Additionally, two methods presented as part of this deliverable focused on providing explainability capabilities to specific domains namely video event recognition and decision-making system.

Within the dimension of **AI Privacy**, two techniques focusing on enhancing privacy within specific scenarios were presented. The first introduces local differential privacy within graph neural networks at the node-level while the other protects data to be analysed by purposely triggering classifications in neural networks. Two additional earlier stage contributions were provided as part of this deliverable within the area of privacy. The first provided an analysis of tools available and requirements needed for the development of widespread privacy within the context AI modelling while the other provided an analysis of techniques available for secure federated learning.

As part of this first WP4 deliverable, an initial contribution to the **AI fairness** dimension of Trustworthy AI has also been provided through the analysis of existing toolsets providing broad AI Fairness capabilities to the ML community, with respect to the specific fairness requirements of AI4Media use cases.

Although the research presented in this deliverable only covers the first year of the project, partners were nevertheless already successful in publishing 6 conference articles over the course of this time as well as submitting 1 journal paper and 2 conference papers. In addition to each partner's individual research contributions, joint collaboration across partners is already underway. WP4 partners were for example successful in driving the organisation of a public workshop focused on Explainable AI, which brought together experts in a wide range of fields (technologists, philosophers, lawyers etc.) to discuss the future development and implications of Explainable AI within our society. A second technical WP4 workshop was also successfully organised on June 2nd 2021, in which partners had the opportunity to present their individual research in detail. The workshop showcased the technical contributions of 8 WP4 partners. As part of this workshop, we also invited Samuel Hoffman as an external guest speaker from IBM Yorktown who provided an overview of the field of AI Fairness with the various technical nuances which researchers should be aware of, as well as an in-depth description of how the AI Fairness Toolkit addresses these needs.

Additionally, following the development of initial contributions to WP4, technical integration between partners has also been initiated. The "Hypespherical class prototypes for adversarial robustness" tool (Section 3.2), for example, developed by AUTH is currently being integrated as part of the Adversarial Robustness Toolbox (Section 3.3) developed by IBM. This integration represents a very good example of mutually beneficial collaboration between partners which enables the technical abilities as well as dissemination efforts of our work to be pooled together.

In addition to joint collaboration within WP4, partners involved in this work-package have also been actively collaborating with WP8. As a result of various cross work-package workshop sessions, partners have successfully mapped each of their module's technical contributions to the needs of specific WP8 use-case requirements.

Finally, in addition to AI4Media collaboration, we also intend to make our work available more broadly as part of the AI4EU platform. Several partners involved in WP4 are already in the process of integrating their modules within the AI4EU module catalogue. E.g. an application to include the work carried out by CERTH on ”Recognition and Explanation of Events in Video using Object Graphs” (see Section 4.4) has already been submitted and is being processed. Within the second year of this project, we also plan to prioritise the integration of other modules to become part of the overall AI4EU catalogue, such as the ‘Adversarial Robustness 360 Toolkit‘ developed by IBM (see Section 3.3), the ”RCV-tool” developed by HES-SO as well as the ”Hypespherical class prototypes for adversarial robustness” method by AUTH .

In summary, the research activities carried out in the first phase of WP4 have been very intense and successful with 6 conference papers already published in top conferences, 1 journal paper submitted, 2 conference papers submitted, as well as 2 technical workshops successfully organised. As the number and venues of publications already accepted suggest, the research contributions of WP4 partners is of very good quality and demonstrate our dedication towards fulfilling the goals of this work-package. Although this report only covers the initial results obtained so far, the research contributions presented in this document nevertheless represent a very solid foundation to build on for the next phases of our research collaborations. The current trend convincingly demonstrates the good continuation of the work carried out so far, on track with the original research plans of this work-package.

The updated version of D4.1 will be provided in M36 (D4.5 – Intermediate toolset for robust, explainable, fair, and privacy-preserving AI) and will include the outcomes of the ongoing as well as additional investigations regarding the tasks covered in this deliverable.

References

[1] L. Zheng, Y. Yang, and A. G. Hauptmann, “Person re-identification: Past, present and future,” *arXiv preprint arXiv:1610.02984*, 2016.

[2] Y. Sun, L. Zheng, Y. Yang, Q. Tian, and S. Wang, “Beyond part models: Person retrieval with refined part pooling (and a strong convolutional baseline),” in *ECCV*, pp. 480–496, 2018.

[3] H. Wang, G. Wang, Y. Li, D. Zhang, and L. Lin, “Transferable, controllable, and inconspicuous adversarial attacks on person re-identification with deep mis-ranking,” in *CVPR*, 2020.

[4] J. Li, R. Ji, H. Liu, X. Hong, Y. Gao, and Q. Tian, “Universal perturbation attack against image retrieval,” in *ICCV*, 2019.

[5] N. Papernot, P. McDaniel, A. Sinha, and M. Wellman, “Towards the science of security and privacy in machine learning,” 2016.

[6] X. Wang, J. Li, X. Kuang, Y. an Tan, and J. Li, “The security of machine learning in an adversarial setting: A survey,” *Journal of Parallel and Distributed Computing*, vol. 130, pp. 12–23, 2019.

[7] V. Mygdalis, A. Iosifidis, A. Tefas, and I. Pitas, “Graph embedded one-class classifiers for media data classification,” *Pattern Recognition*, vol. 60, pp. 585–595, 2016.

[8] V. Mygdalis, A. Iosifidis, A. Tefas, and I. Pitas, “Kernel subclass support vector description for face and human action recognition,” in *2016 First International Workshop on Sensing, Processing and Learning for Intelligent Machines (SPLINE)*, pp. 1–5, IEEE, 2016.

[9] V. Mygdalis, A. Iosifidis, A. Tefas, and I. Pitas, “Semi-supervised subclass support vector data description for image and video classification,” *Neurocomputing*, vol. 278, pp. 51–61, 2018.

[10] V. Mygdalis, A. Tefas, and I. Pitas, “K-anonymity inspired adversarial attack and multiple one-class classification defense,” *Neural Networks*, vol. 124, pp. 296–307, 2020.

[11] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

[12] A. Krizhevsky, G. Hinton, *et al.*, “Learning multiple layers of features from tiny images,” 2009.

[13] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng, “Reading digits in natural images with unsupervised feature learning,” 2011.

[14] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “Imagenet: A large-scale hierarchical image database,” in *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255, Ieee, 2009.

[15] A. Mustafa, S. H. Khan, M. Hayat, R. Goecke, J. Shen, and L. Shao, “Deeply supervised discriminative learning for adversarial defense,” *IEEE transactions on pattern analysis and machine intelligence*, 2020.

[16] Y. Wen, K. Zhang, Z. Li, and Y. Qiao, “A discriminative feature learning approach for deep face recognition,” in *European conference on computer vision*, pp. 499–515, Springer, 2016.

[17] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, “Towards deep learning models resistant to adversarial attacks,” in *International Conference on Learning Representations*, 2018.

[18] I. J. Goodfellow, J. Shlens, and C. Szegedy, “Explaining and harnessing adversarial examples,” *arXiv preprint arXiv:1412.6572*, 2014.

[19] A. Kurakin, I. Goodfellow, S. Bengio, *et al.*, “Adversarial examples in the physical world,” 2016.

[20] Y. Dong, F. Liao, T. Pang, H. Su, J. Zhu, X. Hu, and J. Li, “Boosting adversarial attacks with momentum,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 9185–9193, 2018.

[21] M.-I. Nicolae, M. Sinn, M. N. Tran, B. Buesser, A. Rawat, M. Wistuba, V. Zantedeschi, N. Baracaldo, B. Chen, H. Ludwig, I. Molloy, and B. Edwards, “Adversarial robustness toolbox v1.2.0,” *CoRR*, vol. 1807.01069, 2018.

[22] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, “”gradient-based learning applied to document recognition.”,” in *Proceedings of the IEEE*, 1998.

[23] K. Alex, “”learning multiple layers of features from tiny images”,” in *Technical Report - University of Toronto*, 2009.

[24] L. Zheng, L. Shen, L. Tian, S. Wang, J. Wang, and Q. Tian, “Scalable person re-identification: A benchmark,” in *CVPR*, 2015.

[25] E. Ristani, F. Solera, R. Zou, R. Cucchiara, and C. Tomasi, “Performance measures and a data set for multi-target, multi-camera tracking,” in *European Conference on Computer Vision*, pp. 17–35, 2016.

[26] Z. Zheng, L. Zheng, and Y. Yang, “Unlabeled samples generated by gan improve the person re-identification baseline in vitro,” in *ICCV*, pp. 3754–3762, 2017.

[27] L. Wei, S. Zhang, W. Gao, and Q. Tian, “Person transfer gan to bridge domain gap for person re-identification,” in *CVPR*, pp. 79–88, 2018.

[28] X. Sun and L. Zheng, “Dissecting person re-identification from the viewpoint of viewpoint,” in *CVPR*, pp. 608–617, 2019.

[29] B. Kim, M. Wattenberg, J. Gilmer, C. Cai, J. Wexler, F. Viegas, *et al.*, “Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav),” in *International conference on machine learning*, pp. 2668–2677, PMLR, 2018.

[30] M. Graziani, V. Andrearczyk, and H. Müller, “Regression concept vectors for bidirectional explanations in histopathology,” in *Understanding and Interpreting Machine Learning in Medical Image Computing Applications*, pp. 124–132, Springer, 2018.

[31] R. M. Haralick, K. Shanmugam, and I. H. Dinstein, “Textural features for image classification,” *IEEE Transactions on systems, man, and cybernetics*, no. 6, pp. 610–621, 1973.

[32] X. Wang, R. Girshick, A. Gupta, and K. He, “Non-local neural networks,” in *Proc. IEEE CVPR*, (Salt Lake City, Utah, USA), pp. 7794–7803, June 2018.

[33] X. Wang and A. Gupta, “Videos as space-time region graphs,” in *Proc. ECCV*, vol. 11209, (Munich, Germany), pp. 413–431, Sept. 2018.

[34] J. Yang, W. S. Zheng, *et al.*, “Spatial-temporal graph convolutional network for video-based person re-identification,” in *Proc. IEEE CVPR*, (Seattle, WA, USA), pp. 3286–3296, June 2020.

[35] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural Computation*, vol. 9, no. 8, 1997.

[36] Y.-G. Jiang, Z. Wu, *et al.*, “Exploiting feature and class relationships in video categorization with regularized deep neural networks,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 2, pp. 352–364, 2018.

[37] J. Bernd, D. Borth, *et al.*, “The YLI-MED corpus: Characteristics, procedures, and plans,” *CoRR*, vol. abs/1503.04250, 2015.

[38] S. Ren, K. He, *et al.*, “Faster R-CNN: Towards real-time object detection with region proposal networks,” in *Proc. NIPS*, vol. 28, 2015.

[39] O. Russakovsky, J. Deng, *et al.*, “ImageNet large scale visual recognition challenge,” *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.

[40] S. T. Jan, V. Ishakian, and V. Muthusamy, “AI trust in business processes: the need for process-aware explanations,” in *AAAI Conference on Artificial Intelligence*, pp. 13403–13404, 2020.

[41] G. Ottosson, “Solving machine learning’s ‘last mile problem’ for operational decisions,” *Towards Data Science*, 2019.

[42] C. Molnar, *Interpretable Machine Learning*. Christoph Molnar, 2020. https://christophm.github.io/interpretable-ml-book/.

[43] M. T. Ribeiro, S. Singh, and C. Guestrin, “”Why should I trust you?” Explaining the predictions of any classifier,” in *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 1135–1144, 2016.

[44] S. M. Lundberg and S.-I. Lee, “A unified approach to interpreting model predictions,” *Advances in Neural Information Processing Systems*, vol. 30, pp. 4765–4774, 2017.

[45] I. Alvarez, “Explaining the result of a decision tree to the end-user,” in *ECAI*, vol. 16, p. 411, 2004.

[46] A. Plumerault, H. L. Borgne, and C. Hudelot, “Controlling generative models with continuous factors of variations,” in *International Conference on Learning Representations*, 2020.

[47] Y. Shen, C. Yang, X. Tang, and B. Zhou, “InterFaceGAN: Interpreting the disentangled face representation learned by GANs,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2020.

[48] A. Plumerault, H. Le Borgne, and C. Hudelot, “Avae: Adversarial variational auto encoder,” in *International Conference on Pattern Recognition (ICPR)*, 2020.

[49] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, “Distributed representations of words and phrases and their compositionality,” in *NIPS*, 2013.

[50] Y. Le Cacheux, H. Le Borgne, and M. Crucianu, “Using sentences as semantic representations in large scale zero-shot learning,” in *TASK-CV Workshop - ECCV 2020*, (Online), 2020.

[51] Y. Le Cacheux, A. Popescu, and H. Le Borgne, “Webly supervised semantic embeddings for large scale zero-shot learning,” in *Asian Conference on Computer Vision (ACCV)*, 2020.

[52] O. Adjali, r. Besancon, o. Ferret, H. Le Borgne, and B. Grau, “Multimodal entity linking for tweets,” in *European Conference on Information Retrieval (ECIR)*, (Lisbon, Portugal), april 2020.

[53] O. Adjali, r. Besancon, o. Ferret, H. Le Borgne, and B. Grau, “Building a multimodal entity linking dataset from tweets,” in *International Conference on Language Resources and Evaluation (LREC)*, (Marseille, France), may 2020.

[54] Y. Le Cacheux, H. Le Borgne, and M. Crucianu, “Modeling inter and intra-class relations in the triplet loss for zero-shot learning,” in *International Conference on Computer Vision (ICCV)*, (Seoul, Korea), 2019.

[55] T. Karras, S. Laine, and T. Aila, “A style-based generator architecture for generative adversarial networks,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[56] 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.

[57] Z. Liu, P. Luo, X. Wang, and X. Tang, “Deep learning face attributes in the wild,” in *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.

[58] V. Duddu, A. Boutet, and V. Shejwalkar, “Quantifying privacy leakage in graph embedding,” *arXiv preprint arXiv:2010.00906*, 2020.

[59] X. He, J. Jia, M. Backes, N. Z. Gong, and Y. Zhang, “Stealing links from graph neural networks,” in *30th {USENIX} Security Symposium ({USENIX} Security 21)*, 2021.

[60] I. E. Olatunji, W. Nejdl, and M. Khosla, “Membership inference attack on graph neural networks,” *arXiv preprint arXiv:2101.06570*, 2021.

[61] P. Kairouz, K. Bonawitz, and D. Ramage, “Discrete distribution estimation under local privacy,” in *International Conference on Machine Learning*, pp. 2436–2444, PMLR, 2016.

[62] N. Holohan, S. Braghin, P. M. Aonghusa, and K. Levacher, “Diffprivlib: The ibm differential privacy library,” 2019.

[63] 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* (A. Singh and J. Zhu, eds.), vol. 54 of *Proceedings of Machine Learning Research*, (Fort Lauderdale, FL, USA), pp. 1273–1282, PMLR, 20–22 Apr 2017.

[64] J. Geiping, H. Bauermeister, H. Dröge, and M. Moeller, “Inverting gradients–how easy is it to break privacy in federated learning?,” *arXiv preprint arXiv:2003.14053*, 2020.

[65] L. Sweeney, “k-anonymity: A model for protecting privacy,” *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 10, no. 05, pp. 557–570, 2002.

[66] J. Fürnkranz, E. Hüllermeier, E. Loza Mencía, and K. Brinker, “Multilabel classification via calibrated label ranking,” *Machine Learning*, vol. 73, pp. 133–153, Nov 2008.

[67] J. Zhu, S. Liao, Z. Lei, and S. Z. Li, “Multi-label convolutional neural network based pedestrian attribute classification,” *Image and Vision Computing*, vol. 58, pp. 224–229, 2017.

[68] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, “Image quality assessment: from error visibility to structural similarity,” *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, 2004.

[69] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[70] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, “Mobilenetv2: Inverted residuals and linear bottlenecks,” *arXiv preprint arXiv:1801.04381*, 2018.

[71] X. Wu, R. He, Z. Sun, and T. Tan, “A light cnn for deep face representation with noisy labels,” *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 11, pp. 2884–2896, 2018.

[72] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, “Intriguing properties of neural networks,” *arXiv preprint arXiv:1312.6199*, 2013.

[73] S.-M. Moosavi-Dezfooli, A. Fawzi, and P. Frossard, “Deepfool: a simple and accurate method to fool deep neural networks,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2574–2582, 2016.

[74] N. Carlini and D. Wagner, “Towards evaluating the robustness of neural networks,” in *2017 IEEE Symposium on Security and Privacy (SP)*, pp. 39–57, IEEE, 2017.

[75] B. Gray, S. Tzoannos, L. Overmeire, R. Bauwens, F. Negro, M. Montagnuolo, P. Kemenade, *et al.*, “D8.1 use case definition and requirements,” *AI4Media – A European Excellence Centre for Media, Society and Democracy*, 2021.