Active Membership Inference Test (aMINT): Enhancing Model Auditability with Multi-Task Learning

Daniel DeAlcala, Aythami Morales, Julian Fierrez, Gonzalo Mancera, Ruben Tolosana, Javier Ortega-Garcia
Biometrics and Data Pattern Analytics Lab, Universidad Autonoma de Madrid, Spain
daniel.dealcala@uam.es aythami.morales@uam.es julian.fierrez@uam.es

Abstract

Active Membership Inference Test (aMINT) is a method designed to detect whether given data were used during the training of machine learning models. In Active MINT, we propose a novel multitask learning process that involves training simultaneously two models: the original or Audited Model, and a secondary model, referred to as the MINT Model, responsible for identifying the data used for training the Audited Model. This novel multi-task learning approach has been designed to incorporate the auditability of the model as an optimization objective during the training process of neural networks. The proposed approach incorporates intermediate activation maps as inputs to the MINT layers, which are trained to enhance the detection of training data. We present results using a wide range of neural networks, from lighter architectures such as MobileNet to more complex ones such as Vision Transformers, evaluated in 5 public benchmarks. Our proposed Active MINT achieves over 80% accuracy in detecting if given data was used for training, significantly outperforming previous approaches in the literature. Our aMINT and related methodological developments contribute to increasing transparency in AI models, facilitating stronger safeguards in AI deployments to achieve proper security, privacy, and copyright protection[1].

1. Introduction

The rapid evolution of Artificial Intelligence (AI) in recent years has motivated legal frameworks to safeguard citizens' rights. Institutions like the European Union (EU) have taken a proactive stance, proposing regulations that establish rules and responsibilities for AI developers, for example, the AI Act introduced in June 2024 [50]. These regulations aim to ensure lawful compliance, protect fundamental rights, and achieve transparent, fair, and reliable AI technology. In particular, among all regulations, the EU imposes companies the registration of the trained AI models in an EU-managed database, allowing for external audits and oversight. This trend in legislating AI deployments is not only observed at the EU level. Recently, in October 2024, the White House raised similar concerns through a memo that underscored AI as a matter of national security, advocating for stronger oversight and access to AI technologies to protect citizens [49]. This initiative emphasizes the need for stronger security measures, allowing supervision to prevent possible abuses of citizens' rights.

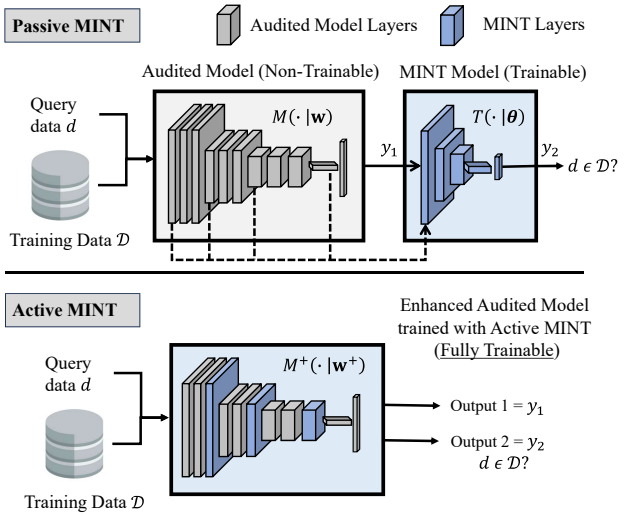

Figure 1. Differences between Passive MINT and Active MINT approaches. The Enhanced Audited model trained with the Active MINT approach (M^+) presents two outputs: 1) the primary task output y (e.g., image classification label); 2) the MINT output (probability that data d has been used during the training process of the model).

The Membership Inference Test (MINT) [8] emerged in 2024 as a research area focused on detecting whether specific data were used to train an AI model, with the aim of exposing unauthorized use of sensitive data, such as biometric information [34]. MINT builds on the field of Membership

[1]Code available in https://github.com/DanieldeAlcala/Membership-Inference-Test.git

Inference Attacks (MIAs) [45], which attempt to extract private information (e.g. medical or consumer data) from trained models, revealing privacy vulnerabilities. These two fields (MIA and MINT) operate under different environmental conditions, resulting in differences in methodologies and outcomes. However, they share similarities, and research in one area can inspire progress in the other.

The main difference between these two lines of research, which leads to different environmental conditions, is that MIAs are attacks on the AI/ML model, meaning that collaboration from the model developer should not be considered. In contrast, MINT, as an auditing tool, allows for a certain level of collaboration with the model developer, such as limited access to the original model. This is supported by current and ongoing legislation, such as that of the White House [49] and the EU [50], which impose model registration in accessible databases for authorized oversight. This is also supported by other legal frameworks such as the General Data Protection Regulation (GDPR) [13] and the California Consumer Privacy Act (CCPA) [3].

Previous MINT research has assumed certain access to the model trained after it has been trained, focusing on analyzing patterns in the model's activations using the so-called MINT Model [7, 8, 10]. A similar approach is carried out in MIAs; however, since MIA is considered an attack, this is carried out on what are known as "shadow models", i.e. models that replicate the original model and over which complete control and access are available [22, 45]. In this work, we explore a scenario where the developer actively engages in the development of the MINT Model, which we term Active MINT (see Fig. 1). This approach contrasts with the previous studies where the model developer did not participate in the MINT Model's development, which we refer to as Passive MINT. In Active MINT, the owner of the model actively improves the auditability of the model by introducing MINT as an objective in his learning.

The main contributions can be summarized as follows.

- We propose Active MINT (aMINT), a novel multitask learning scheme in which two models are trained simultaneously: the Audited Model (based on neural network architectures) and a MINT Model that detects whether specific data were used during the Audited Model training.
- We present an extensive experimental section, evaluating models from lightweight architectures like MobileNet to more complex ones like ViT, across various image datasets, from simple digits to complex face images, demonstrating that aMINT improves auditability over existing MINT approaches.
- We compare our method with existing techniques, further extend their results, and develop new architectures.

The article is structured as follows. Sect. 2 provides a review of the state of the art. Sect. 3 describes the main concepts of the proposed Active MINT and the specific details compared to the previous Passive MINT. The databases, AI models, and experimental protocols are described in Sect. 4. Results are discussed in Sec. 5, whereas the discussion and conclusion are finally drawn in Sects. 6 and 7.

2. Related Works

Membership Inference is a line of research introduced in 2017 by Shokri *et al*. [45] under what they called Membership Inference Attacks (MIAs). This involved attempting to extract sensitive information that the model had been trained on. Later, in 2024, DeAlcala *et al*. introduced MINT [8], focusing on auditing models to ensure that they have been trained with legal data.

2.1. Membership Inference Attacks (MIAs)

Neural networks are often trained on sensitive or private data, posing a potential privacy risk [51] to individuals if such information can be extracted in some way from trained models. MIAs were developed for this purpose, starting with Shokri *et al*. [45], who showed that training data can be extracted under specific conditions. Their approach involved training what they called "Shadow Models" that mimic the original model's behavior while granting full control over training data and unrestricted model access. Training these shadow models required detailed knowledge of the architecture of the original model, the training algorithm, and the dataset statistics. The statistics of the original dataset allowed them to replicate the dataset and use these dataset replicas to train shadow models with the same architecture and training algorithm. A binary classifier trained on the output classification vector of these shadow models then identifies data that are used in training or not. The underlying concept was that these shadow models would generalize to the original model, allowing the binary classifier to detect training data in the original model by extension.

Building on the work of Shokri *et al*., the research line known as MIAs emerged, exploring various strategies for executing these attacks. Some studies, such as [54], investigated applying a threshold directly to the loss values of individual data points rather than training a binary classifier, while others, such as [40, 46], employed a threshold in the prediction output.

Based on the level of access to the model, Nasr *et al*. [36] defined two types of MIAs: Black-box, where only the model's output is available, and White-box, where internal information such as intermediate activations or gradients can be accessed. In their work, Nasr *et al*. introduced a White-box architecture that provided access to the model's intermediate activations and even gradients. They demonstrated that access to intermediate activations did not offer improvements over Black-box architectures. The best results were achieved using gradients.

Another relevant study is that of Nasr *et al*. [35], which served as partial inspiration for our work. In their approach, the authors trained the original model to optimize its primary task while simultaneously regularizing its output to prevent MIAs from identifying training data.

In recent years, numerous studies have been presented that address the intrinsic complexity of the task [39, 42]. These works also demonstrate that the results reported in the literature are often optimistic compared to reality because of insufficient experimental protocols. Furthermore, these concepts have been extended to other intriguing fields such as audio [32, 43], generative models [20], diffusion models [12], and LLM/NLP [33, 44].

2.2. Membership Inference Test (MINT)

Building on the ideas of MIAs, MINT introduces a new perspective [8, 10]. While MIA is considered an attack, MINT is framed as an audit tool. This change in environmental conditions removes the need for shadow model training and allows the possibility of requesting information from the model developer. These differences in environmental conditions lead to different methods and results. The authors demonstrate that their "MINT models" can identify training data with high precision. They conducted experiments on facial recognition [8] and general object classification models [29], exploring different MINT model architectures with different available data. Unlike MIAs, they show that White-box architectures in MINT significantly outperform Black-box architectures.

The same authors expanded on their research with another study that examines the factors influencing the detection performance of MINT Models. In particular, they highlight the options available to the model developer to positively or negatively affect the detection accuracy [7].

Subsequent work in MINT has demonstrated the benefits of considering gradients (gMINT) [9], which are especially useful when auditing LLMs [28].

3. Active MINT: Concept and Architecture

The driving force behind AI is data, and today it is common to train models with large amounts of data [1]. However, access to these data is often restricted by licenses or data protection laws [3, 13]. Data owners have the right to decide how their data are used. A model developer might use data without the necessary permissions, and a tool is needed to detect such misuse [49, 50]. This is the role of MINT.

In Fig. 1, a diagram is presented that illustrates the functioning of Passive MINT is presented. The auditing entity (whether it be an international organization like the EU or the White House or a smaller private entity such as a company) gains access to the Audited Model after it has been trained. With this access, the auditor trains the so-called MINT Model, which is responsible for determining whether specific data was used or not for training. This approach is what we have referred to as Passive MINT (pMINT). For more information on pMINT, see the full paper [8].

In the present work, we will explore a different perspective. Instead of training the MINT Model a posteriori (Passive MINT), here we consider the training of the MINT Model concurrently with the Audited Model. We will refer to this as Active MINT (aMINT). In Figure 1, we can see a schematic representation of this idea.

3.1. Terminology

We consider that the Audited Model M is trained with data $\mathcal{D}$ for a specific task. For a sample d that originates from training data $\mathcal{D}$ or external data $\mathcal{E}$ ($\mathcal{E} \notin \mathcal{D}$) the model M generates an output $y_1 = M(d|\mathbf{w})$ based on d and the trained model parameters $\mathbf{w}$. Intermediate outputs, known as Auxiliary Auditable Data (AAD $= N(d|\mathbf{w}')$) can also be obtained from the input d and a subset of parameters $\mathbf{w}' \subset \mathbf{w}$. The AAD is used as input in the MINT Model that generates the output $y_2 = T(N(d|\mathbf{w}')|\boldsymbol{\theta})$. This terminology applied to our use case is represented in Fig. 2.

3.2. Active MINT: Multi-task Learning Approach

Fig. 2 illustrates the full training workflow along with the implementation details. The goal of Active MINT is to create an Enhanced Audited Model M^+ that improves auditability, allowing to detect whether the input data d belong to $\mathcal{D}$ or $\mathcal{E}$. This Enhanced Audited Model M^+ comprises the Audited Model M, trained using the dataset $\mathcal{D}$ (samples available to train Model M) for the audited task (e.g. image classification) and the MINT model T, which is trained on the AAD for the MINT task. M^+ is defined by its parameters $\mathbf{w}^+ = \{\mathbf{w} \cup \boldsymbol{\theta}\}$. The AAD is generated by feeding both $\mathcal{D}$ and $\mathcal{E}$ (external data not available to train the Audited Model) through the model M.

In this work, AAD is extracted from two points in the network, though extraction from one or more than two points is also possible. The model starts with shared layers, which then split at the point where the AAD is extracted. The initial layers, shown in orange, are shared and need to be optimized for both the Audited Model task and the MINT task. Beyond this point, the remaining layers of model M (in gray) are optimized solely for the Audited Model task, while the layers of model T (in blue) are trained specifically for the MINT task. This shared optimization of the initial layers is a key distinction between Active MINT and Passive MINT, as Active MINT introduces common layers that are trained simultaneously for both tasks.

The batches contain samples from both datasets $\mathcal{D}$ and $\mathcal{E}$. The samples from $\mathcal{E}$ follow the path of the MINT Model (green lines in Fig. 2), while the samples from $\mathcal{D}$ traverse both the MINT Model path and the Audited Model path (gray lines in Fig. 2). Each path has its own loss func-

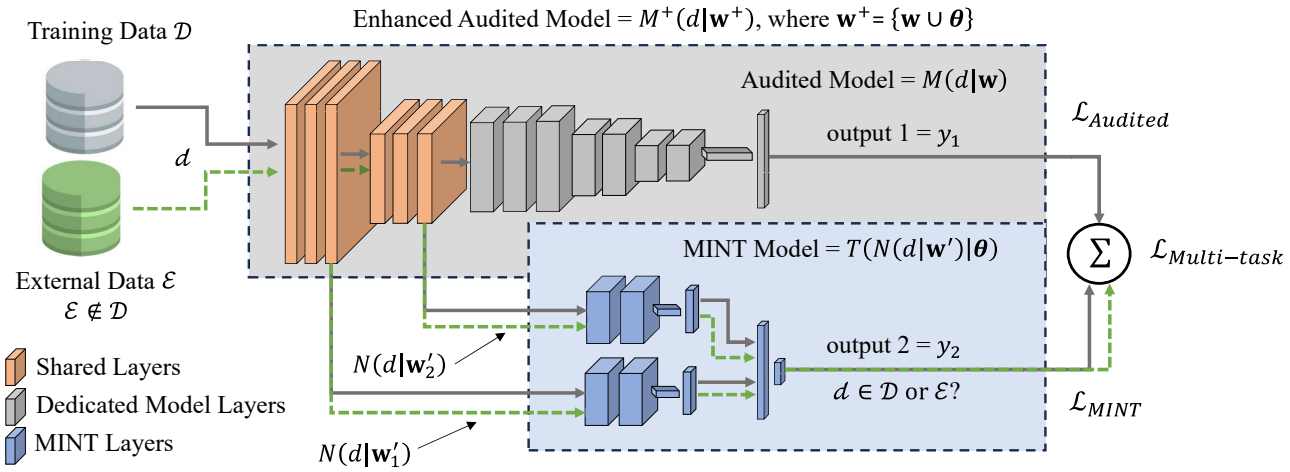

Figure 2. Full workflow of the Active MINT approach to train an Enhanced Audited Model (M^+), where the Audited Model (M) and the MINT Model (T) are jointly trained as a single multitask model that optimizes both tasks simultaneously.

tion, and the losses from each path are combined to jointly train the model. For the model to train effectively, several considerations are essential when designing the multi-task loss function: both losses (the Audited Loss and the MINT Loss) must be normalized to remain within the same range, and a regularizer is included to balance the influence of each loss depending on the architecture and dataset:

$$\mathcal{L}_{\text{Multi-task}} = \lambda_1 \frac{\mathcal{L}_{\text{Audited}}}{\|\mathcal{L}_{\text{Audited}}\|} + \lambda_2 \frac{\mathcal{L}_{\text{MINT}}}{\|\mathcal{L}_{\text{MINT}}\|} + R(\mathbf{w}^+), \quad (1)$$

where $R(\mathbf{w}^+)$ is an L2 regularizer applied to the weights of the Enhanced Audited model, λ_1 and λ_2 are weighting factors to control the convergence of multitask learning, $\mathcal{L}_{\text{Audited}}$ is the loss function for the Audited Model (i.e., it depends on the model and the task for which the audited model is trained), and $\mathcal{L}_{\text{MINT}}$ is a binary cross-entropy loss.

3.3. Active MINT vs Passive MINT

In both Active and Passive MINT, the MINT Model T uses information on how the Audited Model M processes the data d. In Passive MINT, the MINT Model T is trained once the Audited Model M has already been trained, requesting access to the developer only at that point. In contrast, the Active MINT approach proposed here requires the developer to train the MINT Model T alongside their Audited Model M, thereby creating an Enhanced Audited Model M^+. It is crucial that this joint training does not significantly degrade the performance of the Audited Model M, as that would reduce its usefulness compared to Passive MINT strategies. In Sect. 4, we compare this original performance of the Audited Model with that of the Audited Model trained using the Active MINT approach.

Both approaches present advantages and drawbacks. In Passive MINT, the developer must allow access to the Audited model and provide some of the data used for training in order to train the MINT Model. In Active MINT, the developer does not need to disclose training data or grant model access, which can be crucial given the potential risks and the developers' preferences regarding sharing this information, even with regulatory bodies like the EU or US. However, unlike Passive MINT, the developer must actively participate in training the MINT Model.

3.4. Active MINT: Deployment Considerations

Although Active MINT requires active developer participation, it does not mean that the auditor must fully trust them or lose control of the process. Several strategies can ensure that training is performed correctly and verifiably. One option is to provide the developer with a script that logs each training step (e.g., epochs, losses, and parameters) and digitally signs the logs using auditor-controlled keys. These logs could also be automatically uploaded to a remote server in real time for added transparency. Another strategy involves packaging the model, training code, and data into a closed Docker container, which can then be validated by comparing the hash of the final tarball to ensure that the exact intended setup was used. More advanced solutions, such as Multiparty Computation (MPC), can also be explored, for example, letting the developer train the Audited Model while the auditor retains control over the MINT Model.

Although the focus of this work is on presenting the core theoretical framework of aMINT, and not on deployment strategies, we believe it is important to briefly discuss these strategies to demonstrate the feasibility of applying aMINT in real-world, regulation-driven scenarios.

4. Experimental Setup

4.1. Database and Models

In Active MINT, unlike Passive MINT, the Audited Model must be trained alongside the MINT Model, creating the Enhanced Audited Model (Fig. 2). For our experiments, without loss of generality, we have selected to experiment with the task of Image Recognition as it offers a manageable environment for conducting a variety of experiments. Furthermore, most of the work on Membership Inference Attacks (MIAs) [40, 46, 52–54] has been conducted in this domain. Although MIAs and MINT differ in methodology and consequently in results, we have compared our results with recent MIA approaches in order to position our work in the state of the art. We present experiments with various architectures and datasets that we will mix together:

- For the datasets, we have chosen: MNIST (60K training samples and 10K test samples) [25], CIFAR-10 (50K training samples and 10K test samples) [24], GTSRB (39K training samples and 13K test samples) [47], and finally Tiny Imagenet (100K training samples and 20K test samples) [5]. MNIST contains simple images of 10 numbers with very low resolution (28×28). This resolution slightly increases in CIFAR-10, with also 10 classes where the realism grows significantly (32×32). Next, we have GTSRB, which consists of fine-grained class data, differing more from the previous datasets and also featuring an increase in image resolution (from 15×15 the smallest to 222×293 the biggest) and number of classes (43 classes). Finally, Tiny Imagenet contains many more classes (200 classes) and images with higher resolution (64×64). On the other hand, for the Face Recognition domain we have the CASIA WebFace dataset [55], which consists of 500K facial images (250×250).
- Regarding architectures, we present experiments with MobileNet [21], ResNet50 [19], ResNet101 [19], DenseNet121 [23], Xception [4], and ViT [2].

4.2. Experimental Protocol

In our experiments, we compare the proposed Active approach with existing Passive MINT approaches. The Audited Model and the general architecture of the MINT Model (which is optimized through parameter tuning, as outlined in 4.3) will be the same in both approaches: Active MINT and Passive MINT. In the original work, the authors explored different forms of AAD to train the MINT Models. Their primary finding was that using all activation maps as AAD, combined with a CNN-based MINT Model to analyze them, yielded the best results. They tested AAD from various depths of the network, closer to both the input and output layers. Generally, they found that AAD closer to the input yielded better results, although this varied slightly depending on the model.

In their paper, the authors did not explore the combination of AAD from different depths. In our approach, we combine two activation maps as AAD and analyze both using a CNN-based MINT Model, as shown in Fig. 2. We implement this combination in Passive MINT, which also means extending the method proposed by the authors [8]. This allows the MINT Model to leverage more information and potentially achieve better results by selecting the most useful data for the MINT task. Although each model architecture (e.g., ResNet, DenseNet, etc.) is different, the overall process remains the same: selecting two activations maps and analyzing them with a CNN-based MINT Model. It is worth noting that for the ViT model, the activation maps generated are sequences, and thus, the MINT Model is comprised solely of Fully Connected layers.

Although in Passive MINT we know from DeAlcala *et al*. that activation maps closer to the model input generally perform better, in this new Active MINT approach, further experimentation is required to verify this behavior. Therefore, we present experiments using activation maps closer to the input layer (referred to as the Entry Setup), closer to the output layer (Output Setup), and finally, two intermediate activation maps (Middle Setup). This varies depending on the architecture, but to be more specific, these architectures are generally composed of different convolutional blocks, each containing multiple convolutional layers. In the Entry Setup, AAD is extracted from the last two layers of the first convolutional block; in the Output Setup, it is obtained from the last two layers of the last convolutional block; and in the Middle Setup, it comes from the last layers of two intermediate convolutional blocks.

The training subsets of the datasets used are divided into two parts: 50% to train the original model, $\mathcal{D}$, and the other 50% as external data, $\mathcal{E}$. This reduced training size explains the lower performance of M compared to the state-of-the-art models trained on 100% of the data. The performance of the Audited Model M trained alone with this 50% of the dataset is detailed in Sect. 5, allowing a direct comparison with the performance of the Audited Model in this Active MINT setup. All results in Sect. 5 are calculated using the evaluation subsets of these datasets that guarantee no overlap between the training and evaluation data.

4.3. Hyperparameter Tuning

In both aMINT and pMINT architectures, various parameters require tuning to optimize the results. As mentioned, the MINT Models use a CNN architecture to analyze activation maps (AAD in this work are two activation maps of size $H \times W \times C$), except in the case of ViT, where a fully connected network is used due to the sequential nature of its activation maps. The activation map sizes vary widely between experiments. Due to this diversity and the nature of multitask training, hyperparameters can vary significantly.

To illustrate, we compare two examples of MINT Model architectures: an entry setup using ResNet50 for MNIST (E1) and an entry setup using Xception for Tiny-Imagenet (E2). The dimensions of the activation map are $7 \times 7 \times 64$ and $7 \times 7 \times 128$ in E1, while in E2 they are both $16 \times 16 \times 728$.

In E1, due to the lower input dimensionality and the size of the dataset, the MINT architecture has fewer parameters to prevent overfitting, consisting of a single convolutional layer per path with 256 channels, 3×3 filter size, and a stride of 1. Then global grouping is applied and the two outputs are concatenated into a vector of size 512, which is processed through two linear layers with a dropout value of 0.4 in between. In E2, with higher dimensionality and a larger dataset, each path includes two convolutional layers, the first with 1024 filters and the second with 2048 both with 3×3 filter size and a stride of 1. Global pooling is applied, producing a concatenated 4096-dimensional vector, analyzed by two linear layers with a 0.2 dropout.

Furthermore, the Audited Task of E1 is simpler than that of E2, so the ratio $\lambda_2 \div \lambda_1$ (see Eq. 1) is set to 10 for E1 and 10000 for E2. For the learning rate (LR), E1 uses 10^{-5}, while E2 uses 10^{-4}, as ResNet requires a lower LR than Xception. Lastly, E1's $R(\mathbf{w}^+)$ regularization term (see Eq. 1) is set to 10^{-4} to further mitigate overfitting, while E2 is set to 10^{-5}. Both E1 and E2 followed an early stopping strategy, reaching around 50 epochs for E1 and 100 for E2.

5. Results

The first experiment presented examines the Active MINT results in the three setups presented in 4.2: Entry, Middle, and Output, as shown in Table 1. Active MINT trains the Audited Task and the MINT Task simultaneously (Fig. 2), making it especially relevant to compare the performance across all three setups for both tasks. This experiment is conducted on the Object Recognition datasets, which offer a controlled and extensive environment to explore different dataset complexities (from simpler MNIST datasets to more complex Tiny ImageNet). We include as results: 1) the MINT Accuracy (MINT in Tables 1, 2) which refers to the binary classification accuracy between training data ($\mathcal{D}$) and external data ($\mathcal{E}$); 2) the Audited Model Accuracy (Aud in Tables 1, 2) which refers to the performance of the model for the Audited Model task (e.g., image recognition as developed in our experiments). In Table 1, we can observe several noteworthy results:

- MINT accuracy: Depending on the model and dataset, the Entry or Middle setups yield the best results. With MNIST, both setups achieve similar performance. In CIFAR-10 and Tiny ImageNet, the Middle setup achieves higher MINT accuracy, whereas in a more fine-grained dataset like GTSRB, the Entry setup slightly outperforms the others. However, key observation is that the Output setup does not provide better results in any dataset, with the only case where it matches the other setups being the Xception model, where all three setups show similar performance across datasets.
- Audited accuracy: The results follow a similar trend. Entry and Middle setups perform comparably, although in this case the Entry setup appears slightly advantageous, except for Tiny ImageNet. Again, the Output setup consistently yields the lowest performance.

The main takeaway is that the Output Setup offers no benefit. This outcome can be attributed to the fact that the Audited Task and the MINT Task are fundamentally opposed. The Audited Task aims to generalize well, ensuring similar performance on unseen samples, while the MINT Task works against this generalization, aiming to distinguish between samples used in training and those not. In the Output Setup, both tasks share a substantial portion of the layers (red layers in 2), which complicates the training since these layers must optimize for contradictory goals.

Therefore, in subsequent experiments where we compare this technology, we will use results from either of the other two setups (Entry Setup or Middle Setup). Specifically, we will proceed with the Entry Setup since, while both setups yield similar outcomes, it offers a slight advantage for the Audited Task. As discussed in Sect. 6, minimizing performance loss on the Audited Task is essential for achieving a favorable trade-off for the developer. This advantage offered by the Entry setup regarding the Audited Accuracy is again due to the fact that the MINT Model and the Audited Model share fewer "shared layers" (red layers in Fig. 2).

We present a comparative analysis of Active and Passive MINT in Table 2. As explained in Sect. 4, we expand the Passive MINT approach by using two Activation Maps as AAD, allowing for a consistent MINT Model architecture across Passive and Active MINT, with slight adjustments in parameter tuning for each model M and dataset D. The activation maps in Passive MINT are also drawn from an entry-level setup, as this configuration yields the best results [8]. The Audited Model also remains consistent across methods, ensuring that the comparison is as fair as possible.

In Active MINT, both models are trained jointly, requiring Face Recognition models to be trained from scratch with the MINT Model. It is important to note that the goal is not to develop competitive FR models but to compare Active and Passive MINT in this domain. FR accuracy is reported as the accuracy of identification in the $10,575$ identities used during training (random chance: $1/10575$).

In Table 2, we present the results of the comparison between Active and Passive MINT for the MINT Model Task and the Audited Model Task. Additionally, in the Passive MINT rows, we indicate the performance of the Audited Task when trained independently, allowing us to evaluate any trade-offs in performance when training both tasks simultaneously. As shown, the MINT accuracy is consid-

	MNIST (10 classes)											
Passive MINT	MobileNet		ResNet50		ResNet100		DenseNet121		Xception		ViT	
	MINT	Aud	MINT	Aud	MINT	Aud	MINT	Aud	MINT	Aud	MINT	Aud
Entry Setup	0.86	0.96	0.83	0.97	0.83	0.97	0.82	0.99	0.85	0.98	0.80	0.94
Middle Setup	0.88	0.92	0.82	0.97	0.82	0.96	0.83	0.99	0.83	0.98	0.81	0.95
Output Setup	0.82	0.90	0.80	0.90	0.77	0.80	0.81	0.98	0.80	0.83	0.80	0.93
	CIFAR-10 (10 classes)											
Passive MINT	MobileNet		ResNet50		ResNet100		DenseNet121		Xception		ViT	
	MINT	Aud	MINT	Aud	MINT	Aud	MINT	Aud	MINT	Aud	MINT	Aud
Entry Setup	0.86	0.41	0.86	0.53	0.87	0.58	0.86	0.80	0.86	0.64	0.86	0.19
Middle Setup	0.91	0.41	0.89	0.49	0.87	0.54	0.87	0.80	0.86	0.62	0.87	0.20
Output Setup	0.85	0.33	0.85	0.39	0.85	0.40	0.82	0.76	0.87	0.46	0.88	0.19
	GSTRB (43 classes)											
Passive MINT	MobileNet		ResNet50		ResNet100		DenseNet121		Xception		ViT	
	MINT	Aud	MINT	Aud	MINT	Aud	MINT	Aud	MINT	Aud	MINT	Aud
Entry Setup	0.89	0.89	0.86	0.98	0.87	0.97	0.85	0.99	0.86	0.99	0.80	0.91
Middle Setup	0.87	0.82	0.83	0.98	0.85	0.98	0.84	0.99	0.88	0.98	0.81	0.93
Output Setup	0.83	0.81	0.81	0.95	0.81	0.94	0.83	0.99	0.88	0.95	0.79	0.90
	Tiny ImageNet (200 classes)											
Passive MINT	MobileNet		ResNet50		ResNet100		DenseNet121		Xception		ViT	
	MINT	Aud	MINT	Aud	MINT	Aud	MINT	Aud	MINT	Aud	MINT	Aud
Entry Setup	0.80	0.11	0.80	0.12	0.80	0.11	0.86	0.34	0.88	0.28	0.81	0.17
Middle Setup	0.88	0.10	0.80	0.09	0.86	0.09	0.88	0.34	0.87	0.28	0.81	0.19
Output Setup	0.79	0.12	0.79	0.07	0.79	0.08	0.79	0.33	0.86	0.16	0.80	0.17

Table 1. Enhanced Audited Model performance in terms of MINT and Audited Model accuracies (MINT and Aud respectively) across the three setups (Entry Setup, Middle Setup, and Output Setup) using datasets that range from simpler to more complex patterns, alongside models varying from basic to advanced architectures. Perfect classification accuracy is represented by 1.

	MNIST (10 classes)											
Method	MobileNet		ResNet50		ResNet100		DenseNet121		Xception		ViT	
	MINT	Aud	MINT	Aud	MINT	Aud	MINT	Aud	MINT	Aud	MINT	Aud
Passive MINT	0.52	0.97	0.51	0.97	0.51	0.98	0.52	0.99	0.54	0.99	0.53	0.97
Active MINT	0.86	0.96	0.83	0.97	0.83	0.97	0.82	0.99	0.85	0.98	0.80	0.94
	CIFAR-10 (10 classes)											
Method	MobileNet		ResNet50		ResNet100		DenseNet121		Xception		ViT	
	MINT	Aud	MINT	Aud	MINT	Aud	MINT	Aud	MINT	Aud	MINT	Aud
Passive MINT	0.66	0.41	0.66	0.55	0.69	0.59	0.60	0.80	0.67	0.66	0.60	0.20
Active MINT	0.86	0.41	0.86	0.53	0.87	0.58	0.86	0.80	0.86	0.64	0.86	0.19
	GSTBR (43 classes)											
Method	MobileNet		ResNet50		ResNet100		DenseNet121		Xception		ViT	
	MINT	Aud	MINT	Aud	MINT	Aud	MINT	Aud	MINT	Aud	MINT	Aud
Passive MINT	0.59	0.92	0.61	0.98	0.61	0.98	0.61	0.99	0.59	0.99	0.60	0.96
Active MINT	0.89	0.89	0.86	0.98	0.87	0.97	0.85	0.99	0.86	0.99	0.80	0.91
	Tiny Imagenet (200 classes)											
Method	MobileNet		ResNet50		ResNet100		DenseNet121		Xception		ViT	
	MINT	Aud	MINT	Aud	MINT	Aud	MINT	Aud	MINT	Aud	MINT	Aud
Passive MINT	0.57	0.12	0.59	0.17	0.61	0.17	0.56	0.35	0.65	0.28	0.60	0.20
Active MINT	0.80	0.11	0.80	0.12	0.80	0.11	0.86	0.34	0.88	0.28	0.81	0.17
	CASIA Webface (10,575 classes)											
Method	MobileNet		ResNet50		ResNet100		DenseNet121		Xception		ViT	
	MINT	Aud	MINT	Aud	MINT	Aud	MINT	Aud	MINT	Aud	MINT	Aud
Passive MINT	0.60	0.17	0.61	0.12	0.61	0.14	0.62	0.31	0.63	0.17	0.59	0.10
Active MINT	0.86	0.15	0.80	0.11	0.81	0.13	0.82	0.29	0.84	0.17	0.76	0.09

Table 2. Comparison between Active and Passive MINT in terms of MINT accuracy and Audited Model accuracy (MINT and Aud respectively). Perfect classification accuracy is represented by 1.

Method	ResNet50	
	CIFAR-10	GSTRB
Salem *et al.* MIA [40]	0.61	0.67
Yeom *et al.* MIA [54]	0.64	0.79
Song *et al.* MIA [46]	0.65	0.68
Ye *et al.* MIA [53]	0.52	0.60
Watson *et al.* MIA [52]	0.63	0.79
Passive MINT [7, 10]	0.66	0.61
Active MINT **(Ours)**	**0.86**	**0.86**

Table 3. Comparison between recent works in MIA and our Active MINT using the same ResNet-50 architecture. Perfect classification accuracy is represented by 1.

erably higher in Active MINT, surpassing Passive MINT across all models and datasets. However, this significant improvement in MINT accuracy comes with a drawback, as it slightly reduces the original Audited Task accuracy. As explained previously, this is due to the conflicting objectives of the MINT task, which oppose the generalization goal of the audit task. We discuss this trade-off in Sect. 6.

Due to the limited MINT literature, we only benchmark against [7, 10]. In Table 3, we also compare Active MINT with recent MIAs. As discussed previously, the environmental conditions between MIAs and MINT differ, since MIAs are considered attacks, whereas MINT functions as an auditing tool. Thus, unlike the comparison between Active and Passive MINT, this one occurs under different and often highly specific conditions used in MIA studies. However, this comparison serves to contextualize our results.

6. Discussion

The aim of MINT is to promote auditable AI aligned with emerging legislation and citizens' rights. Regulations in regions such as the EU and the US increasingly require transparent and accountable AI systems [49, 50], requiring tools such as MINT. Achieving this goal requires the collaboration of developers, either voluntarily for public transparency or mandated by legislation. The objective of this work is to provide a tool that enables these goals, recognizing the legislative demand for such resources while avoiding involvement in legal matters.

The existing Passive MINT (pMINT) and the proposed Active MINT (aMINT) enable auditability by determining whether specific data were used for training. pMINT operates post-training and requires access to both the model and part of the training data. aMINT also requires developer collaboration, but does not need access to the model or any training data, thus avoiding the challenges of sharing sensitive information or granting access to private models. Active MINT on the other side requires the developer to train the MINT Model alongside their model. The objective is to make this process as simple as possible, aiming to keep it automated and transparent to the developer.

As shown in Sect. 5, aMINT achieves significantly higher accuracy than pMINT, making it a much more powerful tool for meeting fairness, transparency, and trustworthiness goals, with only a very small decrease in the Audited Model performance. This trade-off is highly favorable, even allowing some Audited Models to maintain their original performance. However, in certain cases, even a small drop in performance may not be acceptable. For these scenarios, it is crucial to ensure that Active MINT does not affect the performance of the Audited Model. As seen in Sect. 5, this can be achieved by reducing the number of 'shared layers' (Fig. 2) between the MINT Model and the Audited Model. Another option would be to use pMINT methods, despite their lower MINT detection accuracy.

As future work, exploring advanced multi-task learning techniques could help further reduce performance degradation in high-stakes settings. Promising strategies include meta-optimization [14], conflict-averse training [27], and knowledge distillation [26] to better balance both tasks and preserve model representations. Adapting these methods to aMINT is a promising direction for future research.

As future work, we also plan to connect MINT with related research in explainable AI [11, 37, 48], privacy- and bias-aware AI methods [6, 30, 38, 41], and cryptographic constructions in pattern recognition [15–18, 31].

7. Conclusion

We proposed a method called Active MINT (aMINT), a novel auditing tool in line with the latest international regulation in trustworthy AI. aMINT enhances model auditability by training a MINT Model alongside the Audited Model (e.g., a classification model) to detect the data used in the training process. This results in an architecture composed of these two components, termed the Enhanced Audited Model. We present an extensive comparison between the proposed method and similar methods in the literature. Our results demonstrate the feasibility of Active MINT in a wide range of scenarios. Active MINT consistently detects the data used in training with a precision greater than 80%, significantly outperforming previous approaches in the literature and opening a novel line of research aimed at improving the trustworthiness of AI models.

Acknowledgement

BBforTAI (PID2021-127641OB-I00 MICINN/FEDER), HumanCAIC (TED2021-131787B-I00 MICINN), M2RAI (PID2024-160053OB-I00 MICIU/FEDER) and Cátedra ENIA UAM-Veridas (NextGenerationEU PRTR TSI-100927-2023-2). DeAlcala funded by FPU21/05785 and Mancera by PRE2022-104499. ELLIS Unit Madrid.

References

[1] Abdulaziz Aldoseri, Khalifa N Al-Khalifa, and Abdel Magid Hamouda. Re-Thinking Data Strategy and Integration for Artificial Intelligence: Concepts, Opportunities, and Challenges. *Applied Sciences*, 13(12):7082, 2023. 3

[2] Dosovitskiy Alexey. An Image is Worth 16×16 Words: Transformers for Image Recognition at Scale. *arXiv preprint arXiv: 2010.11929*, 2020. 5

[3] California State Legislature and the California Civil Code. California Consumer Privacy Act CCPA. AB 375, Chau., 2018. 2, 3

[4] François Chollet. Xception: Deep Learning With Depthwise Separable Convolutions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1251–1258, 2017. 5

[5] Patryk Chrabaszcz, Ilya Loshchilov, and Frank Hutter. A Downsampled Variant of ImageNet as an Alternative to the CIFAR datasets. *arXiv preprint arXiv:1707.08819*, 2017. 5

[6] Daniel DeAlcala, Ignacio Serna, Aythami Morales, Julian Fierrez, and Javier Ortega-Garcia. Measuring bias in AI models: An statistical approach introducing n-sigma. In *IEEE Conf. on Computers, Software, and Applications (COMPSAC)*, pages 1167–1172, 2023. 8

[7] Daniel DeAlcala, Gonzalo Mancera, Aythami Morales, Julian Fierrez, Ruben Tolosana, and Javier Ortega-Garcia. A Comprehensive Analysis of Factors Impacting Membership Inference. In *IEEE/CVF Conf. on Computer Vision and Pattern Recognition Workshops (CVPRw)*, pages 3585–3593, 2024. 2, 3, 8

[8] Daniel DeAlcala, Aythami Morales, Gonzalo Mancera, Julian Fierrez, Ruben Tolosana, and Javier Ortega-Garcia. Is my Data in your AI Model? Membership Inference Test with Application to Face Images. *arXiv preprint arXiv:2402.09225*, 2024. 1, 2, 3, 5, 6

[9] Daniel DeAlcala, Aythami Morales, Julian Fierrez, Gonzalo Mancera, and Ruben Tolosana. gMINT: Gradient-based membership inference test applied to image models. In *IEEE/CVF Conf. on Computer Vision and Pattern Recognition Workshops (CVPRw)*, pages 2781–2790, 2025. 3

[10] Daniel DeAlcala, Aythami Morales, Julian Fierrez, Gonzalo Mancera, Ruben Tolosana, and Ruben Vera-Rodriguez. MINT-Demo: Membership inference test demonstrator. In *AAAI Workshop on AI Governance: Alignment, Morality, and Law (AIGOV)*, 2025. 2, 3, 8

[11] Ivan Deandres-Tame, Ruben Tolosana, Ruben Vera-Rodriguez, Aythami Morales, Julian Fierrez, and Javier Ortega-Garcia. How good is ChatGPT at face biometrics? A first look into recognition, soft biometrics, and explainability. *IEEE Access*, 12:34390–34401, 2024. 8

[12] Jinhao Duan, Fei Kong, Shiqi Wang, Xiaoshuang Shi, and Kaidi Xu. Are Diffusion Models vulnerable to Membership Inference Attacks? In *Proceedings of the International Conference on Machine Learning*, pages 8717–8730, 2023. 3

[13] European Parliament and the Council of the European Union. General Data Protection Regulation GDPR. EU 2016/679., 2016. 2, 3

[14] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks. In *Intl. Conf. on Machine Learning*, 2017. 8

[15] Mahdi Ghafourian, Ruben Vera-Rodriguez, Julian Fierrez, et al. Blockchain and biometrics: Survey, GDPR elements, and future directions. *arXiv:2302.10883v3*, 2025. 8

[16] Marta Gomez-Barrero, Javier Galbally, Aythami Morales, and Julian Fierrez. Privacy-preserving comparison of variable-length data with application to biometric template protection. *IEEE Access*, 5:8606–8619, 2017.

[17] Marta Gomez-Barrero, Emanuele Maiorana, Javier Galbally, Patrizio Campisi, and Julian Fierrez. Multi-biometric template protection based on homomorphic encryption. *Pattern Recognition*, 67:149–163, 2017.

[18] Ahmad Hassanpour, Majid Moradikia, Bian Yang, Ahmed Abdelhadi, Christoph Busch, and Julian Fierrez. Differential privacy preservation in robust continual learning. *IEEE Access*, 10, 2022. 8

[19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016. 5

[20] Benjamin Hilprecht, Martin Härterich, and Daniel Bernau. Monte Carlo and reconstruction Membership Inference Attacks against Generative Models. *Proceedings on Privacy Enhancing Technologies*, 2019. 3

[21] Andrew G Howard. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21–26, 2017. 5

[22] Hongsheng Hu, Zoran Salcic, Lichao Sun, Gillian Dobbie, Philip S Yu, and Xuyun Zhang. Membership Inference Attacks on Machine Learning: A survey. *ACM Computing Surveys*, 54(11s):1–37, 2022. 2

[23] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely Connected Convolutional Networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4700–4708, 2017. 5

[24] Alex Krizhevsky, Geoffrey Hinton, et al. Learning Multiple Layers of Features from Tiny Images. 2009. 5

[25] Yann LeCun, Corinna Cortes, and CJ Burges. MNIST handwritten digit database. *ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist*, 2, 2010. 5

[26] Wei-Hong Li and Hakan Bilen. Knowledge Distillation for Multi-task Learning. In *Proceedings of the European Conference on Computer Vision*, 2020. 8

[27] Bo Liu, Xingchao Liu, Xiaojie Jin, Peter Stone, and Qiang Liu. Conflict-Averse Gradient Descent for Multi-task learning. *Advances in Neural Information Processing Systems*, 2021. 8

[28] Gonzalo Mancera, Daniel DeAlcala, Julian Fierrez, Ruben Tolosana, and Aythami Morales. Is my text in your AI model? Gradient-based membership inference test applied to LLMs. In *arXiv preprint arXiv:2503.07384*, 2025. 3

[29] Gonzalo Mancera, Daniel DeAlcala, Aythami Morales, Ruben Tolosana, and Julian Fierrez. Membership inference

test: Auditing training data in object classification models. In *AAAI Workshop on Deployable AI (DAI)*, 2025. 3

[30] Gonzalo Mancera, Aythami Morales, Julian Fierrez, et al. PBa-LLM: Privacy- and bias-aware NLP using named-entity recognition (NER). In *IAPR Intl. Conf. on Document Analysis and Recognition Workshops (ICDARw)*, 2025. 8

[31] Pietro Melzi, Hatef Otroshi Shahreza, Christian Rathgeb, Ruben Tolosana, Ruben Vera-Rodriguez, Julian Fierrez, Sébastien Marcel, and Christoph Busch. Cancelable face biometrics with soft-biometric privacy enhancement. *IEEE Access*, 13:128420–128431, 2025. 8

[32] Yuantian Miao, Minhui Xue, Chao Chen, Lei Pan, Jun Zhang, Benjamin Zi Hao Zhao, Dali Kaafar, and Yang Xiang. The Audio Auditor: User-Level Membership Inference in Internet of Things Voice Services. *Proceedings on Privacy Enhancing Technologies*, pages 209–228, 2021. 3

[33] Fatemehsadat Mireshghallah, Kartik Goyal, Archit Uniyal, Taylor Berg-Kirkpatrick, and Reza Shokri. Quantifying Privacy Risks of Masked Language Models Using Membership Inference Attacks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, page 8332–8347, 2022. 3

[34] Aythami Morales, Julian Fierrez, Ruben Vera-Rodriguez, and Ruben Tolosana. SensitiveNets: Learning Agnostic Representations with Application to Face Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(6): 2158–2164, 2021. 1

[35] Milad Nasr, Reza Shokri, and Amir Houmansadr. Machine Learning with Membership Privacy using Adversarial Regularization. In *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security*, page 634–646, 2018. 3

[36] Milad Nasr, Reza Shokri, and Amir Houmansadr. Comprehensive Privacy Analysis of Deep Learning: Passive and Active White-box Inference Attacks against Centralized and Federated Learning. In *Proceedings of the IEEE Symposium on Security and Privacy*, pages 739–753, 2019. 2

[37] Alfonso Ortega, Julian Fierrez, Aythami Morales, Zilong Wang, Marina Cruz, Cesar L. Alonso, and Tony Ribeiro. Symbolic ai for xai: Evaluating lfit inductive programming for explaining biases in machine learning. *Computers*, 10 (11):154, 2021. 8

[38] Alejandro Peña, Julian Fierrez, Aythami Morales, Gonzalo Mancera, Miguel Lopez, and Ruben Tolosana. Addressing bias in LLMs: Strategies and application to fair AI-based recruitment. In *AAAI/ACM Conf. on AI, Ethics, and Society (AIES)*, 2025. 8

[39] Shahbaz Rezaei and Xin Liu. On the difficulty of Membership Inference Attacks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7892–7900, 2021. 3

[40] Ahmed Salem, Yang Zhang, Mathias Humbert, Pascal Berrang, Mario Fritz, and Michael Backes. Ml-Leaks: Model and data independent Membership Inference Attacks and defenses on Machine Learning Models. In *Proc. Annual Network and Distributed System Security Symposium*, 2018. 2, 5, 8

[41] Ignacio Serna, Aythami Morales, Julian Fierrez, Manuel Cebrian, Nick Obradovich, and Iyad Rahwan. Algorithmic discrimination: Formulation and exploration in deep learning-based face biometrics. In *AAAI Workshop on Artificial Intelligence Safety (SafeAI)*, pages 146–152, 2020. 8

[42] Avital Shafran, Shmuel Peleg, and Yedid Hoshen. Membership Inference Attacks Are Easier on Difficult Problems. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14820–14829, 2021. 3

[43] Muhammad A. Shah, Joseph Szurley, Markus Mueller, Thanasis Mouchtaris, and Jasha Droppo. Evaluating the vulnerability of end-to-end automatic speech recognition Models to Membership Inference Attacks. In *Proceedings of Interspeech*, 2021. 3

[44] Virat Shejwalkar, Huseyin A Inan, Amir Houmansadr, and Robert Sim. Membership inference attacks against nlp classification models. In *Proceedings of the NeurIPS 2021 Workshop Privacy in Machine Learning*, 2021. 3

[45] Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership Inference Attacks against machine learning Models. In *Proceedings of IEEE Symposium on Security and Privacy*, pages 3–18, 2017. 2

[46] Liwei Song and Prateek Mittal. Systematic Evaluation of Privacy Risks of Machine Learning Models. In *Proceedings of the USENIX Security Symposium*, pages 2615–2632, 2021. 2, 5, 8

[47] Johannes Stallkamp, Marc Schlipsing, Jan Salmen, and Christian Igel. The German Traffic Sign Recognition Benchmark: A multi-class classification competition. In *Proceedings of the International Joint Conference on Neural Networks*, pages 1453–1460, 2011. 5

[48] Javier Tello, Marina de la Cruz, Tony Ribeiro, Julian Fierrez, et al. Symbolic AI (LFIT) for XAI to handle biases. In *European Conf. on AI Workshops (ECAIw)*, 2023. 8

[49] The White House. Memorandum on Advancing the United States' Leadership in Artificial Intelligence., 2024. 1, 2, 3, 8

[50] Europen Union. Artificial Intelligence Act. *European Parliament: European Parliamentary Research Service*, 2024/1689, (Updated June 2024). 1, 2, 3, 8

[51] R. Veldhuis et al. *Privacy and Security Matters in Biometric Technologies*. Springer, 2025. 2

[52] Lauren Watson, Chuan Guo, Graham Cormode, and Alex Sablayrolles. On the Importance of Difficulty Calibration in Membership Inference Attacks. *Proc. of the Intl. Conf. on Learning Representations*, 2022. 5, 8

[53] Jiayuan Ye, Aadyaa Maddi, Sasi Kumar Murakonda, Vincent Bindschaedler, and Reza Shokri. Enhanced Membership Inference Attacks against Machine Learning Models. In *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security*, pages 3093–3106, 2022. 8

[54] Samuel Yeom, Irene Giacomelli, Matt Fredrikson, and Somesh Jha. Privacy risk in Machine Learning: Analyzing the connection to overfitting. In *Proceedings of the IEEE Computer Security Foundations Symposium*, pages 268–282, 2018. 2, 5, 8

[55] Dong Yi, Zhen Lei, Shengcai Liao, and Stan Z Li. Learning Face Representation from Scratch. *arXiv preprint arXiv:1411.7923*, 2014. 5