Are Your Sensitive Attributes Private? Novel Model Inversion Attribute Inference Attacks on Classification Models

Shagufta Mehnaz, *The Pennsylvania State University;* Sayanton V. Dibbo and Ehsanul Kabir, *Dartmouth College;* Ninghui Li and Elisa Bertino, *Purdue University*

https://www.usenix.org/conference/usenixsecurity22/presentation/mehnaz

This artifact appendix is included in the Artifact Appendices to the Proceedings of the 31st USENIX Security Symposium and appends to the paper of the same name that appears in the Proceedings of the 31st USENIX Security Symposium.

August 10–12, 2022 • Boston, MA, USA

978-1-939133-31-1

Open access to the Artifact Appendices to the Proceedings of the 31st USENIX Security Symposium is sponsored by USENIX.

A Artifact Appendix

A.1 Abstract

Our artifact primarily implements and evaluates our proposed model inversion attack strategies– LOMIA and CSMIA. It includes our codebase for the above two attack strategies, datasets, and APIs to query target and attack models. While our codebase is heavily Python-dependent, it can run without any specific hardware requirements. In our *requirements* file, we list packages used in our codebase, and in the installation guideline, we describe details of the installation guideline and also include a *readme* file, where we add details step by step procedure to run and evaluate our artifact, i.e., each of the claims we make. We provide the APIs of the target as well as attack ML models, with instructions to perform each attack leveraging these APIs and training datasets. We expect to reproduce the results shown in the paper, although the attack models trained on different accounts with their optimization technique might cause slightly different results.

A.2 Artifact check-list (meta-information)

- **Algorithm: N/A**
- **Program: N/A**
- **Compilation: N/A**
- **Transformations: N/A**
- **Binary: N/A**
- **Model: Decision Tree (DT), Deepnet (DNN)**
- **Data set: Adult, GSS, FiveThirtyEight**
- **Run-time environment: Python (3.7.11)**
- **Hardware: N/A**
- **Run-time state: N/A**
- **Execution: N/A**
- **Security, privacy, and ethical concerns: N/A**
- **Metrics: Precision, Recall, Accuracy, F1 score, G-mean, MCC, FPR**
- **Output:**
- **Experiments: LOMIA, CSMIA (we report median of 5 runs, expected variation 2-3% for GSS, and 5-7% for FiveThirtyEight datasets)**
- **How much disk space required (approximately)?: N/A**
- **How much time is needed to prepare workflow (approximately)?: N/A**
- **How much time is needed to complete experiments (approximately)?: N/A**
- **Publicly available (explicitly provide evolving version reference)?: N/A**
- **Code licenses (if publicly available)?: N/A**
- **Data licenses (if publicly available)?: N/A**
- **Workflow frameworks used?: N/A**
- **Archived (explicitly provide DOI or stable reference)?: N/A**

A.3 Description

A.3.1 How to access

This artifact codebase is shared via Github. The codebase can be downloaded from there. Also, we share the target model training datasets for our experiments, and associated attack datasets as well as attack model APIs. Our target models (DT, DNN) and attack models can be accessed via APIs provided on the Github.

A.3.2 Hardware dependencies

No specific hardware is required to run this code.

A.3.3 Software dependencies

Our codebase can be run in the python environment using any python package manager. For ease of use, we have provided instructions on how to set up the environment using Anaconda.

A.3.4 Data sets

We use three publicly available datasets: General Social Survey (GSS), Adult, and Fivethirtyeight. We perform pre-processing on each dataset and do train-test splits (datasets available on Github). Details descriptions about each dataset can be found in Section 5.1 in the main manuscript. We provide the training datasets in our shared Github repository.

A.3.5 Models

We consider two different target models: decision tree (DT), and deepnet (DNN). We use the 'ensemble' model as the attack model. All models are trained on BigML with default features. Details about model training can be found in Section 5.2 in the main manuscript. All our ML models both target and attack, trained on each dataset, can be accessed via our provided APIs.

A.3.6 Security, privacy, and ethical concerns

Our artifact does not have security, privacy, and ethical concerns.

A.4 Installation

This artifact is dependent on the python environment. Required packages have to be installed before running the codebase. A list of requirements packages are in the *requirements* file. In our Github link, we provide step-by-step installation guidelines with Conda environment creation and installation of the dependencies. Also, procedures to run the codebase to produce outputs are all described in the GitHub readme file: https://github.com/smehnaz/black-boxMIAI

A.5 Experiment workflow

A.6 Evaluation and expected results

The following are the main claims that are supported by the artifact we submitted.

- We demonstrate two new proposed black-box model in- version attacks: (1) confidence score-based attack (CSMIA) and (2) label-only attack (LOMIA) outperforms existing FJRMIA

- Our proposed attacks can achieve better performance while estimating both binary (Table 12-13) and multi-valued (Table 10) and also multiple sensitive attributes (Table 15-16)
- We empirically show that model inversion attacks have disparate vulnerability property (Figure: 4b, 9, 10)
- We also evaluate partial knowledge attack scenarios of a target record and demonstrate that our attacks' performance is not impacted significantly in those scenarios (Figure: 5, 11-13)
- We also experiment on distributional privacy leakage and show that these attacks can also breach the privacy of datasets outside training but drawn from the same distribution. (Figure: 4a, 8)

In Sections 5.4.1, 5.4.2, and 5.4.3, we present our key comparisons of our proposed LOMIA and CSMIA attack performances compared to existing FJRMIA. This shows on different datasets, and target models our attacks outperform existing attacks in different performance metrics. To reproduce the results on LOMIA or CSMIA strategy, one has to run each attack particular strategy in our codebase described in the Github and also added end of this section. Other existing technique strategies are explained in the manuscript. In Tables 4-9, we provide target model confusion matrices and Fig. 2 in the manuscript shows the comparisons in GSS and Adult datasets. Tables 12 and 13 show performance comparisons on GSS, and Adult datasets.

For the second claim, we estimate both binary ('alcohol') and multi-valued ('age') in the FiveThirtyEight dataset (details in Section 5.4.3). We also estimate multiple attributes (inferring 'age' along with 'alcohol' (Table 16) and inferring 'alcohol' along with 'age' (Table 15)). To experiment with disparate vulnerability in model inversion attack, we query each attack model on specific subgroup instances of the training dataset, as presented in Section 5.7 of the manuscript. In the partial attack experiment, we perform the attack for estimating sensitive attributes with gradually missing more non-sensitive attributes in the training data. We present the results in Section 5.8 of the paper. We present the distributional privacy leakage experiment results in Fig. 8.

All steps for each experiment and reproduction steps are added to the Github repository. We experiment with our proposed CSMIA, LOMIA as well as baseline FJRMIA to compare performances. The different kinds of attack experiments that we perform using LOMIA, CSMIA, and FJRMIA are as follows:

- Inferring a single binary sensitive attribute
- Inferring a single multi-valued sensitive attribute
- Inferring multiple sensitive attributes
- Inferring sensitive attributes when one or more non-sensitive attributes are unknown
- Inferring sensitive attributes from data that was not originally on the training set (distributional privacy leakage)
- Analyzing disparate vulnerability of model inversion attack on different subgroups

Now we list out how these experiments' results can be reproduced one by one. One way is to use the configuration files we provided to reproduce results as a figure or a table presented in the paper. If the configuration file name is "config_x.yaml", then one only has to run the following command in the terminal "python main.py −−param config_x.yaml". Another way is to write down the configuration .yaml file and use it from the terminal in the same way.

Inferring a single binary sensitive attribute: For the Adult dataset we infer the marital attribute, and for the GSS the xmovie attribute. We use all combinations of DT and DNN models and both LOMIA and CSMIA attacks. One can use the built-in configuration files from the table in the GitHub readme file. For example: To infer marital from Adult Dataset and DT model using LOMIA attack, one can use the configuration file "configs/table_13/lomia_dt.yaml". Then one can compare the results with Table 12 and Table 13 of the paper.

Inferring a single multi-valued sensitive attribute: For the 538 dataset, we infer the multi-valued age attribute. We use the DT model and both LOMIA and CSMIA attacks. One can use the built-in configuration files from the table in the GitHub readme file. For example: To infer age using the CSMIA, one can use the configuration file "configs/table_10/csmia.yaml". Then one can compare the results with Table 10 of the paper. Because 538 is a very small dataset, in many case3 instances the target models confidence values are the same and the CSMIA chooses the sensitive attribute randomly which is the reason behind the deviation from the paper result. For LOMIA, the training of the ensemble attack model introduces the variation in the experiment result. We discuss these at the end of this section.

Inferring multiple sensitive attributes: For the 538 dataset, we infer both alcohol and age attributes. We use the DT model and both LOMIA and CSMIA attacks. One can use the built-in configuration files from the table in the GitHub readme file. For example: To infer age using the CSMIA, one can use the configuration file "configs/table_15_16/csmia.yaml". Then the results can be compared with Tables 15 and 16 from the paper. The same reason holds for the slight variation between the outputs.

Inferring sensitive attributes when one or more non-sensitive attributes are unknown: For LOMIA, we infer sensitive attributes when 1-9 non-sensitive attributes are missing in order of their importance. The details of this experiment can be found in section 5.8 of the paper. We perform the attack on both the Adult and GSS datasets

and both DT and DNN models. One can use the built-in configuration files from the table in the GitHub readme file. For example: To perform the partial knowledge attack on Adult DT, the following configuration file may be used "configs/figure_5/dt.yaml". The output can be compared with Figures 5, 11, and 12 from the paper.
For CSMIA, we infer sensitive attributes when 1-2 non-sensitive attributes are unknown. We only attack Adult DT for this setting. One can use the built-in configuration files from the directory mentioned in the GitHub readme file. For example: To perform the partial knowledge attack when occupation and capital-gain are unknown, the following configuration file may be used "configs/figure_13/occupation_capgain.yaml". The outputs can be compared with figure 13 from the paper.

Disparate Vulnerability Experiment: In this experiment, we estimate the disparate vulnerability of subgroups using the APIs on specific subgroup instances. We have to define the followings: dataset: Adult/GSS (Depends on which dataset being attacked), attack_type: *LOMIA*,*target_model_type*: DT/DNN (Depends on which target model is being attacked sensitive_attributes: ['marital']/['xmovie'] (marital for Adult, xmovie for GSS), missing_nonsensitive_attributes: [], attack_category: 'disparate_vulnerability', extra_field_for_attack_category: x (The vulnerable subgroup (one of the fields on the dataset, e.g., *male/female*)). We can also use specific built-in configuration files. For example, the following file can be used for Adult DNN sex subgroups: "configs/figure_4b/sex.yaml" One can compare the outputs with the ones presented in the paper in figure 4b, 9, and 10.

Distributional Privacy Experiment: For this experiment, a similar setup with the above code snippet can be used with attack_category: 'distributional_privacy_leakage' to get the result of dataset from the same distribution but not training data. Also, as an alternative to this, different files can be used as mentioned in the readme for this attack. For example in DNN CSMIA this can be used to get results on distributional privacy leakage: "configs/figure_8/adult_csmia_dnn_on_DSd.yaml". The outputs can be compared with Figures 4a and 8 from the paper.

LOMIA Attack Dataset Preparation: First, to build the attack dataset, one needs to query the target model. Then build the attack models with those datasets to perform the attack. We provide the attack models. Therefore, by querying the attack model, one can perform the attack. However, the attack dataset can be generated with configuration file names provided in the readme file. For example to generate dataset on Adult DT model, while inferring *marital* sensitive attribute, following configuration file can be used "configs/table_3/adult_dt.yaml".

We expect to have similar results in all experiments as presented in the empirical results. However, since the attack models are trained on a different BigML account, and BigML applies its optimization techniques while generating the attack models (ensembles), there might be a slight variation in results produced by this artifact.

A.7 Experiment customization

A.8 Notes

A.9 Version

Based on the LaTeX template for Artifact Evaluation V20220119.