Configuration Manual

MSc Research Project
Artificial Intelligence

Vikas Varma Malipeddi
Student ID: 22143335

School of Computing
National College of Ireland

Supervisor: Dr.Anh Duong Trinh (Senja)

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	Vikas Varma Malipeddi		
Student ID:	22143335		
Programme:	MSc in Artificial Intelligence	**Year:**	2023
Module:	MSc Research Method		
Lecturer:	Dr Anh Duong Trinh (Senja)		
Submission Due Date:	31/01/2024		
Project Title:	Optimizing Adversarial Attacks on ML-Powered Malware Detection Systems		
Word Count:	1009	**Page Count:**	12

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: Vikas Varma Malipeddi

Date: 31/01/2024

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Configuration Manual: Optimizing Adversarial Attacks on ML-Powered Malware Detection Systems

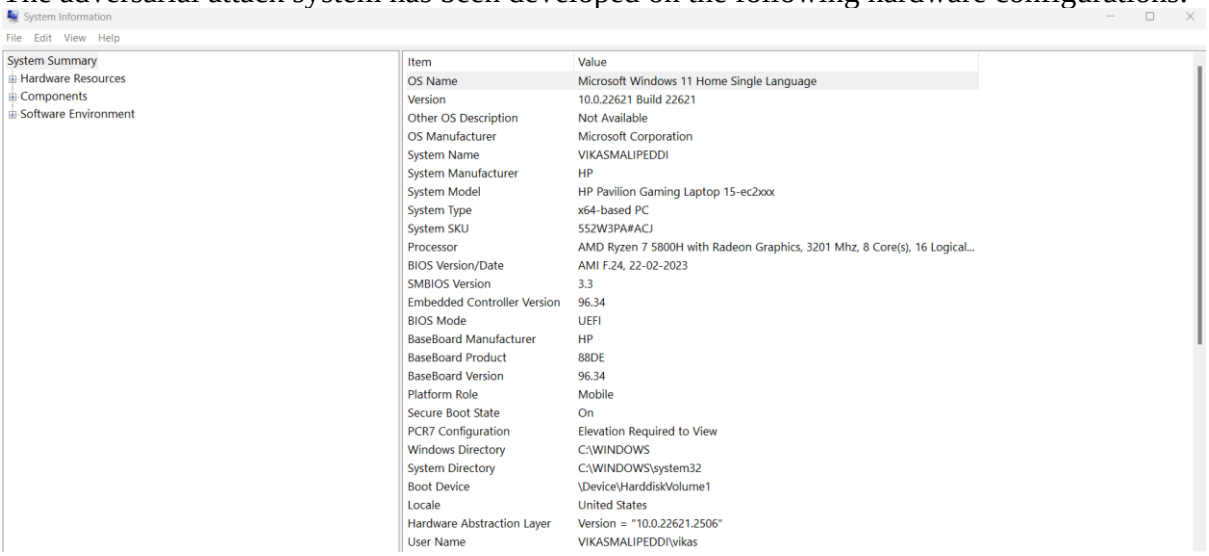
Vikas Varma Malipeddi
Student ID: 22143335

1. Introduction

This manual provides detailed instructions for setting up and executing code related to the implementation of query-efficient adversarial attacks against machine learning models. The focus is on understanding and enhancing the robustness of machine learning models against adversarial attacks. The following sections guide you through the necessary configurations, requirements, and tools.

2. System Specification

The adversarial attack system has been developed on the following hardware configurations:



- Processor: Ryzen 7 5000 series
- Operating System: Windows 11
- Ram: 16 GB (DDR4)
- Storage Hard Drive: 1TB (SSD)

3. Software Used:

The following tools are required for the development and usage of the query-efficient adversarial attack system Pycharm Application below are the imported libraries to the required models to run:

- Torch
- TensorFlow and Keras
- Pandas
- NumPy
- Matplotlib
- Scikit-learn.

4. Installation of the Software:

Python Installation:

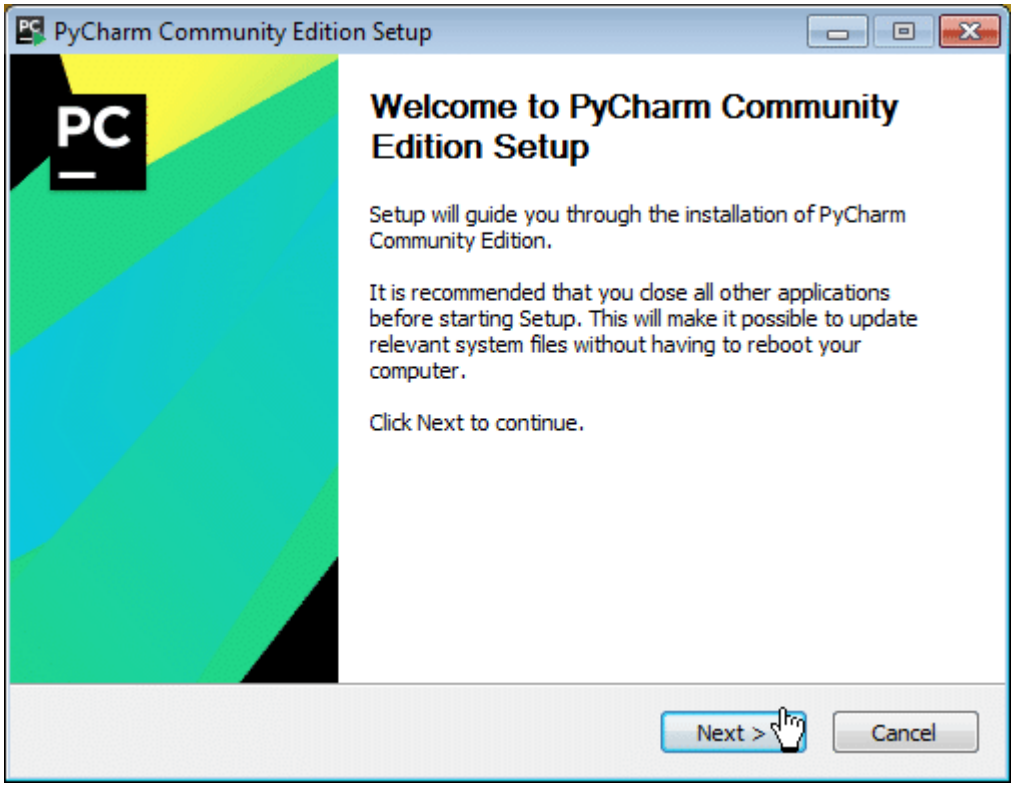
- Download and install Python 3.x from the official website: Python.
- Ensure that Python is added to the system PATH during installation.

Pycharm Installation:

Step 1: To download PyCharm, visit the official website of JetBrains: Download PyCharm

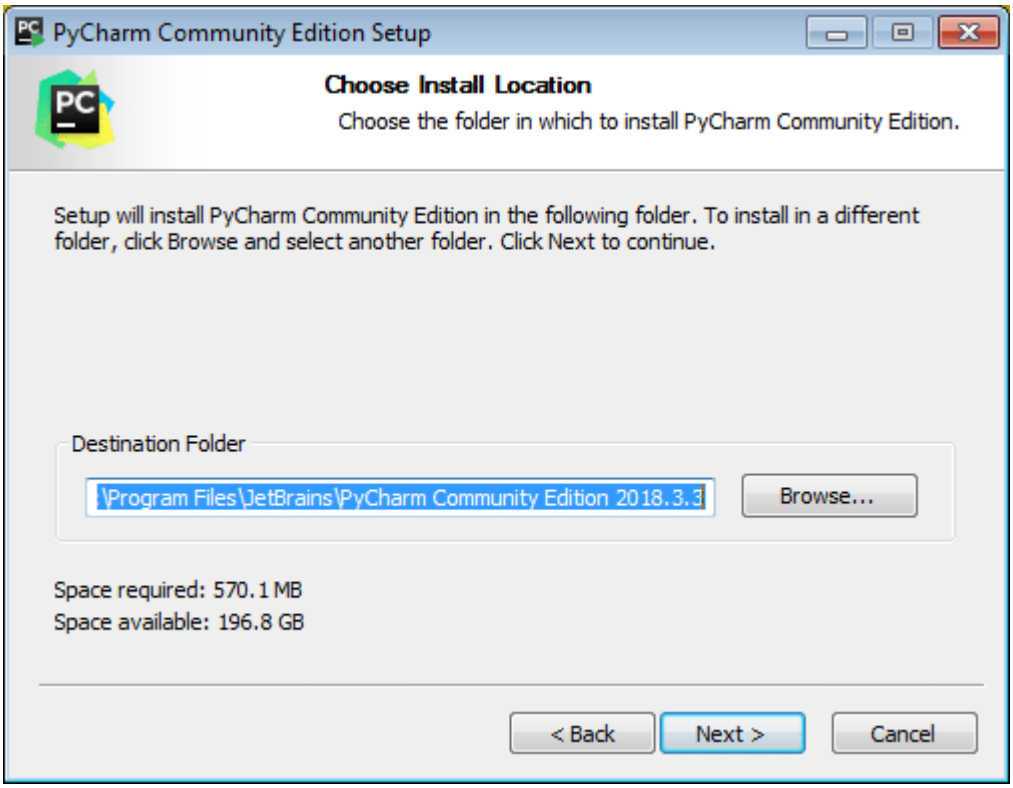
Step 2: After downloading the file, click on it

Step 3: When the following window appears, click on Next and the installation process will start

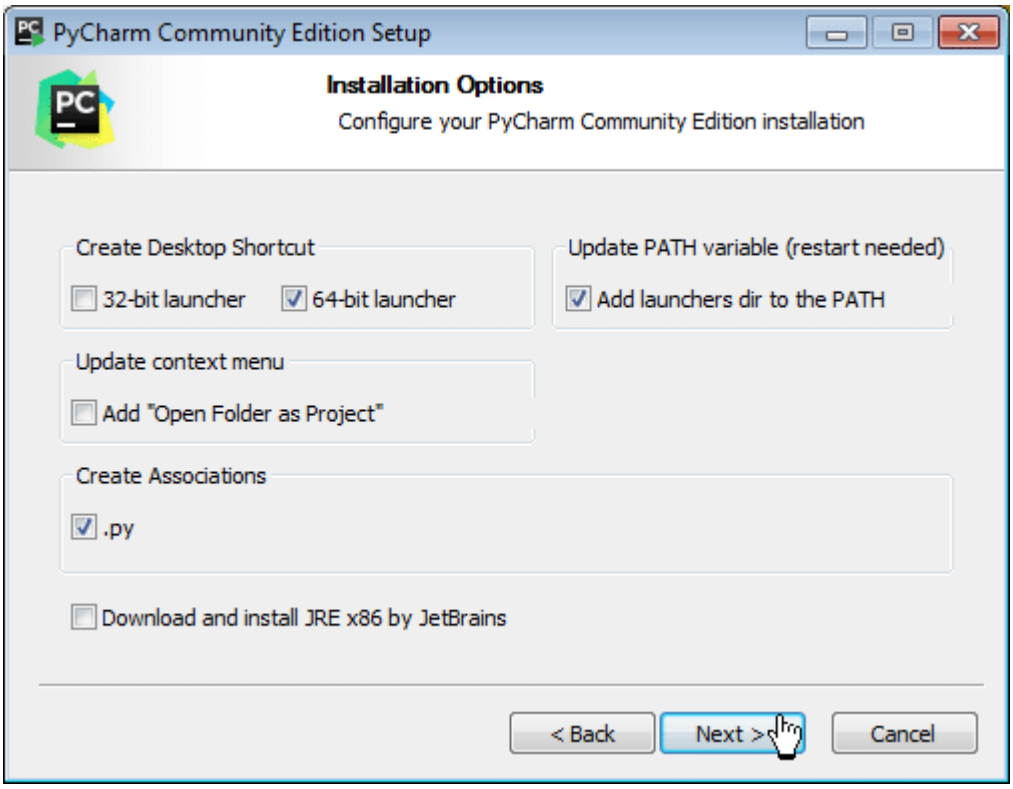


Step 3: After clicking on Next, first, a window for setting up the installation location will appear.

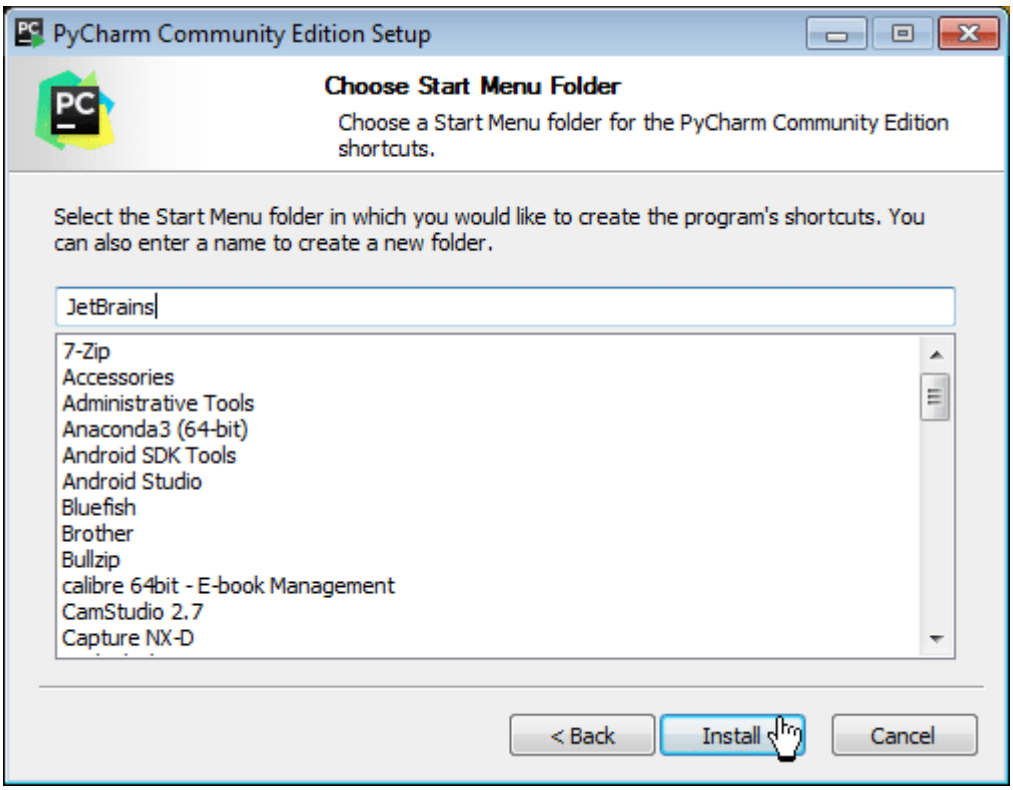
Note: You can either select a folder for the installation location or retain the default path.



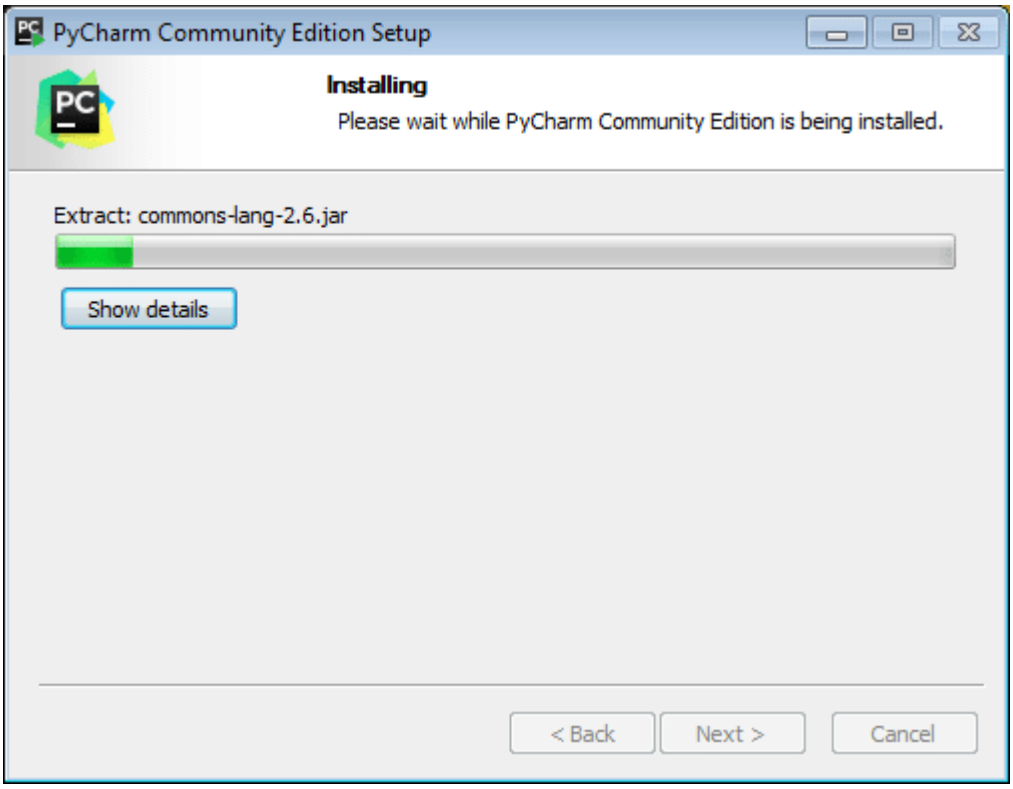
Step 4: In the next step, you can set the Installation Options as per requirements, and then, click on the Next button to proceed.



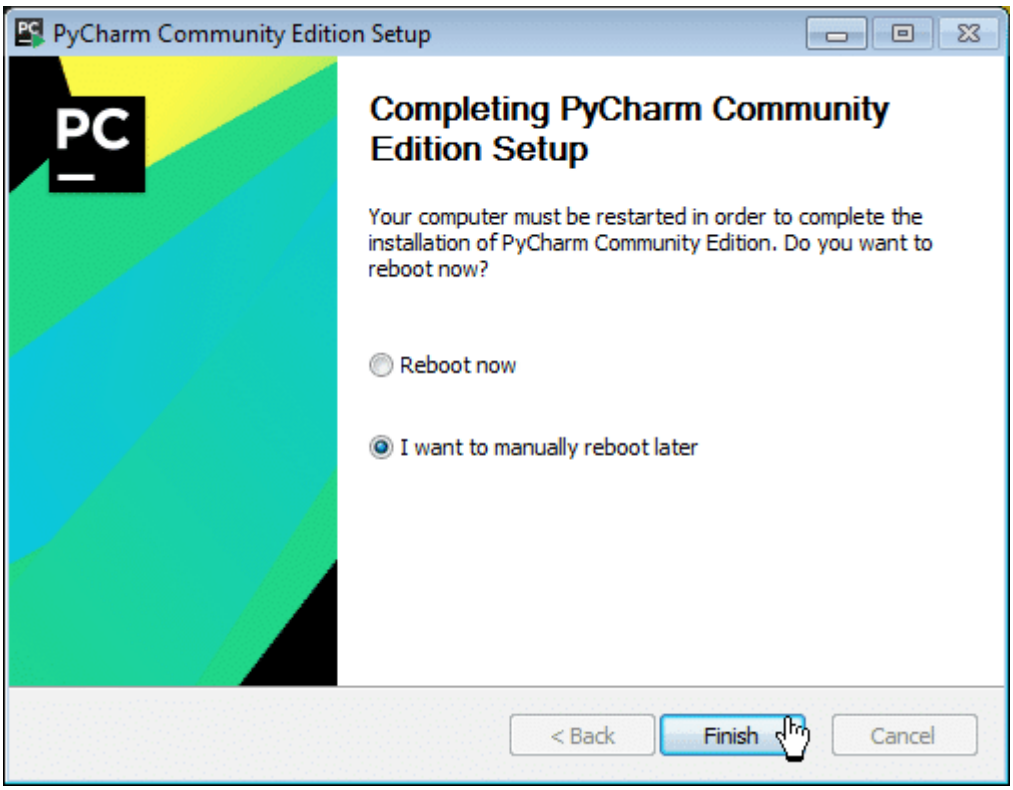
Step 5: Now, you have to select the Start Menu folder, or you can leave it as default



Step 6: After these steps, click on the Install button as above to start the installation process.



Step 7: When you click on the Finish button, your PyCharm installation completes



Now, you have successfully installed PyCharm and Python both in your system.

Virtual Environment Setup:

- Create a new virtual environment for the application.
- Activate the virtual environment and install the required packages using pip.

5. Source Code and Models

Obtain the source code for query-efficient adversarial attacks against machine learning models. The repository may include pre-trained models and scenario scripts. Found on relevant repositories on platforms like GitHub.

6. Code Execution

Open Pycharm and then Python scripts to develop and execute the code. The workflow includes:
Execution Steps:

- Preprocess the Dataset File

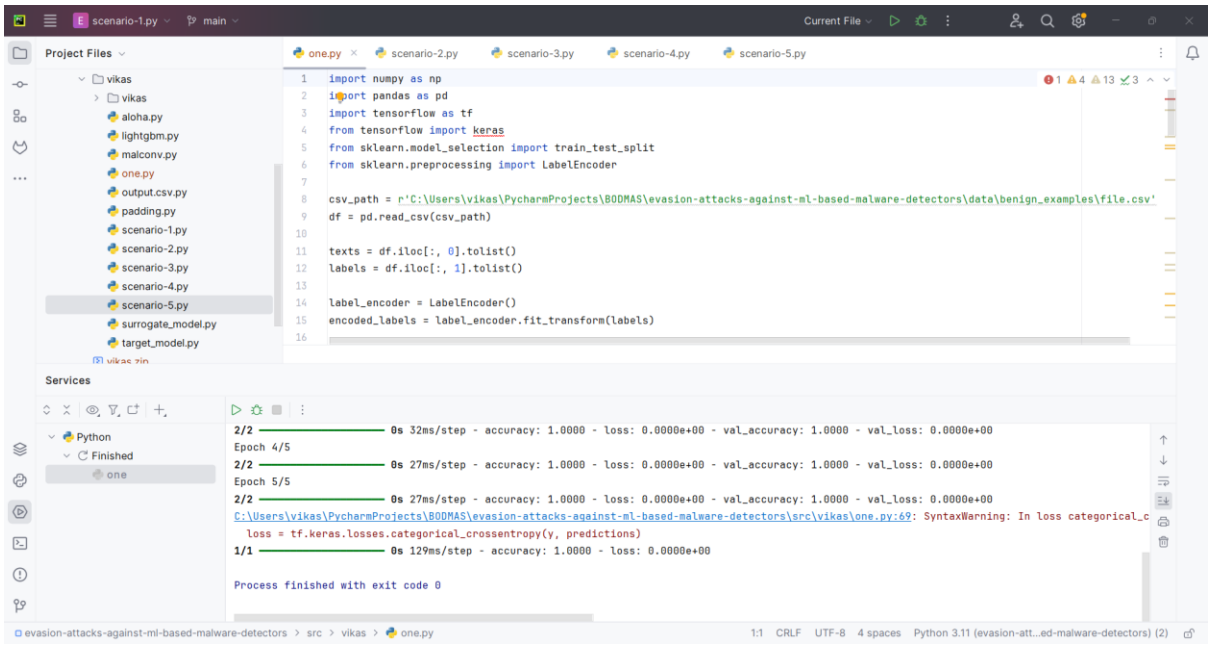
```
C:\Users\vikas\PycharmProjects\BODMAS\evasion-attacks-against-ml-based-malware-detectors\src\checkpoints\end2end\MalConv
-keras-master (1)\MalConv-keras-master>python3 preprocess.py file.csv
Preprocessing ..... this may take a while ...
Finished ..... 44 sec
Preprocessed data store in ../saved/preprocess_data.pkl

C:\Users\vikas\PycharmProjects\BODMAS\evasion-attacks-against-ml-based-malware-detectors\src\checkpoints\end2end\MalConv
-keras-master (1)\MalConv-keras-master>
```

- Perform the Prediction through the scenario 1

Scenario 1: Shared Training Data:

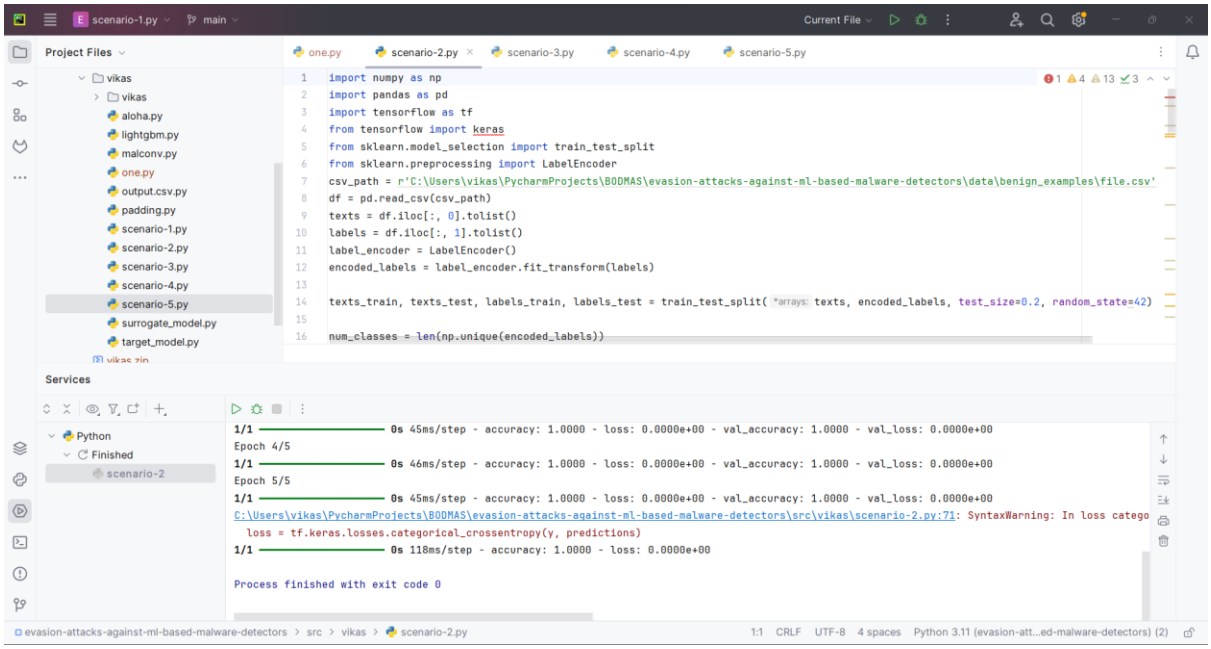
- In this scenario, both the target detection model and the surrogate model have access to the identical training dataset. They are trained on the same set of data samples, allowing for a direct comparison of their performance and vulnerability to adversarial attacks.



- Perform the Prediction through scenario 2.

Scenario 2: Partially Shared Training Data:

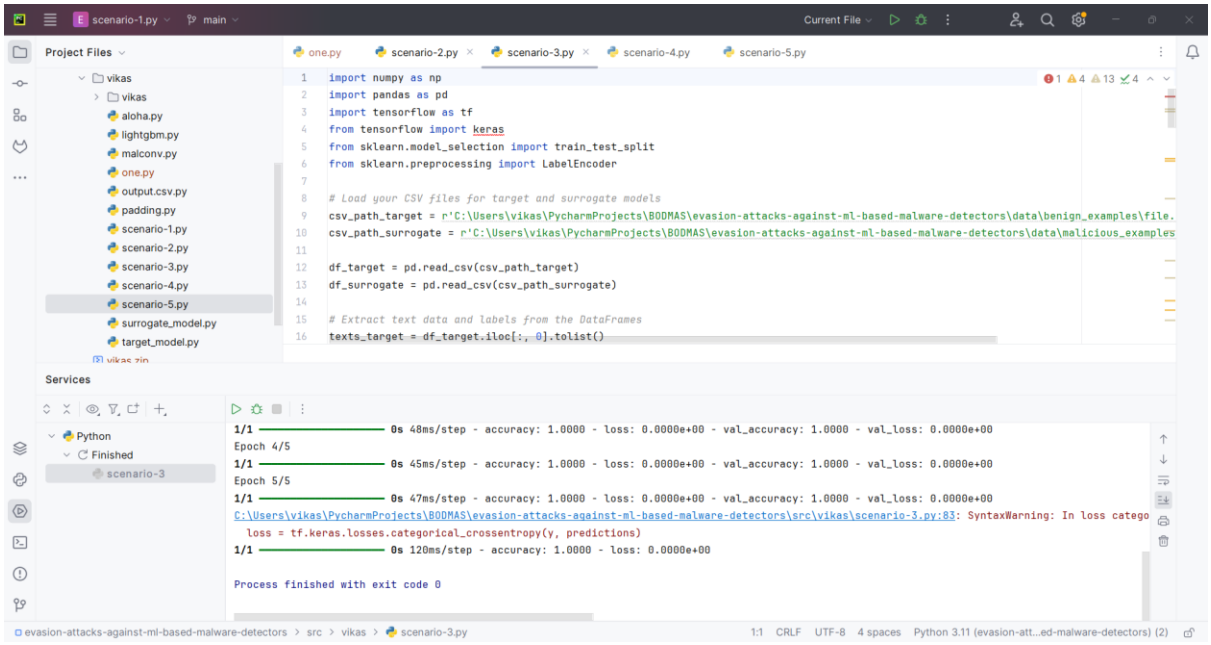
- In this scenario, the target detection model and the surrogate model share only a portion of their training data. While some data samples are common between the two models, they also have distinct training data subsets. This introduces a degree of similarity and divergence in their training experiences.



- Perform the Prediction through scenario 3.

Scenario 3: Non-Shared Training Data:

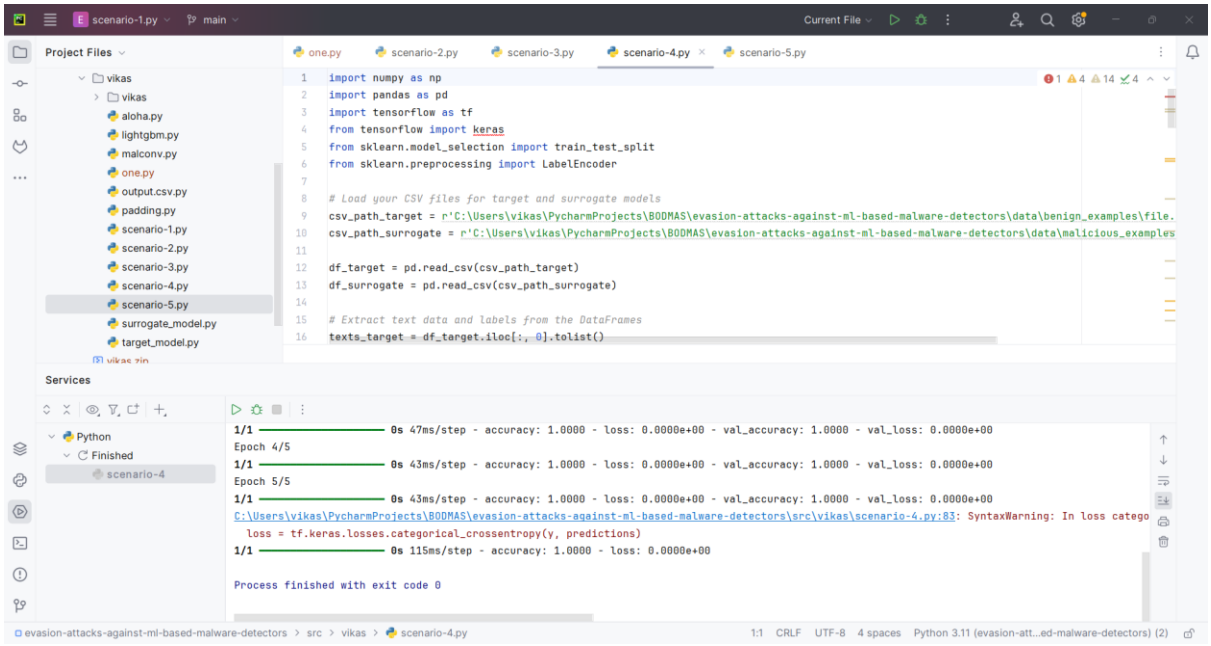
Here, the target detection model and the surrogate model do not share any training data. They are trained independently on entirely separate datasets. This scenario assesses the transferability of adversarial attacks between models that have no common training ground.



- Perform the Prediction through the scenario 4

Scenario 4: Identical Model Architectures:

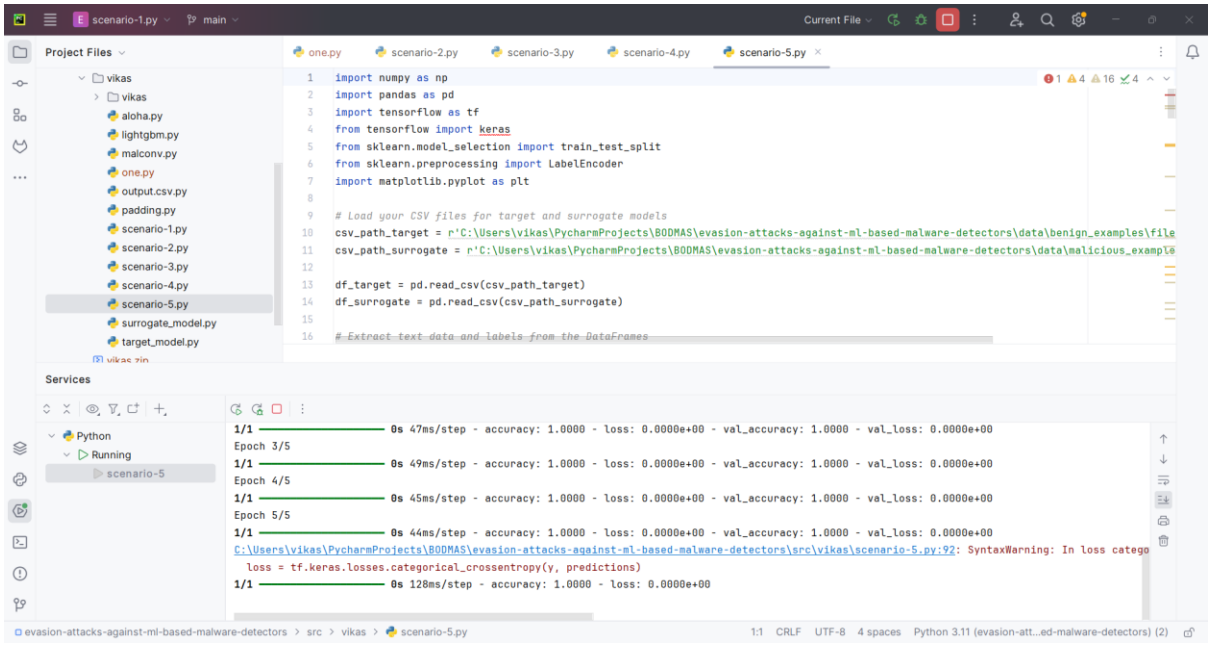
- In this scenario, both the target detection model and the surrogate model have the same architectural design. They share the same model structure, making it a direct architecture-to-architecture comparison.



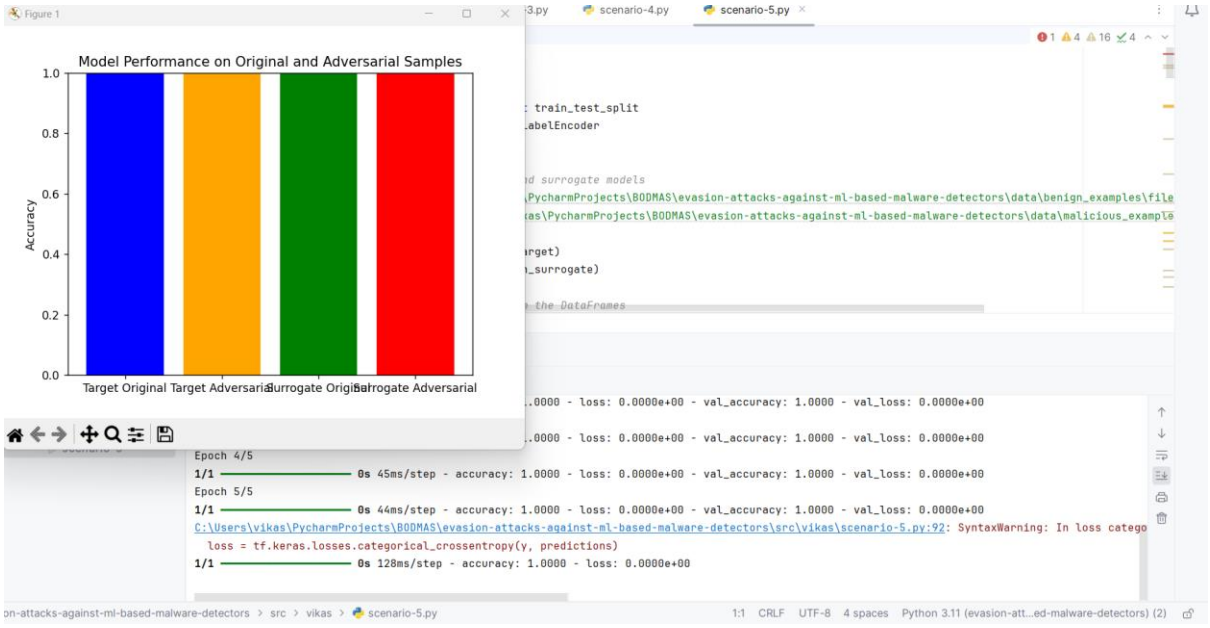
- Perform the Prediction through scenario 5.

Scenario 5: Different Model Architectures:

- This scenario involves target and surrogate models with distinct architectural designs. Examples of these architectures include MalConv. The comparison explores the impact of varying model structures on adversarial attack transferability and effectiveness.



- Perform the Model Evaluation for all the methods.



This manual serves as a comprehensive guide for configuring the installation of the required software/tools for implementing query-efficient adversarial attacks against machine learning models.

References

- Python: (https://www.python.org/)
-Pycharm community available at Download PyCharm: Python IDE for Professional Developers by JetBrains.
- TensorFlow: [TensorFlow Installation Guide](https://www.tensorflow.org/install)