Configuration Manual

Evaluating the Effectiveness of Neural Networks for Cyber Attack Detection in IoT Ecosystems

MSc Research Project
MSc Cloud Computing

Lyuble Daniel
Student ID: x23249463

School of Computing
National College of Ireland

Supervisor: Dr. Jorge M. Cortés-Mendoza

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name: .... Lyuble Daniel ....

Student ID: .... x23249463..

Programme: ...... MSc Cloud Computing .... **Year:** ...2024-2025.....

Module: .... Research Project ....

Lecturer: .... Dr. Jorge M. Cortés-Mendoza ....

Submission Due Date: ....15 – 09 - 2025...

Project Title: Evaluating the Effectiveness of Neural Networks for Cyber Attack Detection in IoT Ecosystems

Word Count: ......477........... **Page Count:** ....7.............................

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: .... Lyuble Daniel ....

Date: ....15-09-2025...

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

Lyuble Daniel
Student ID: x23249463

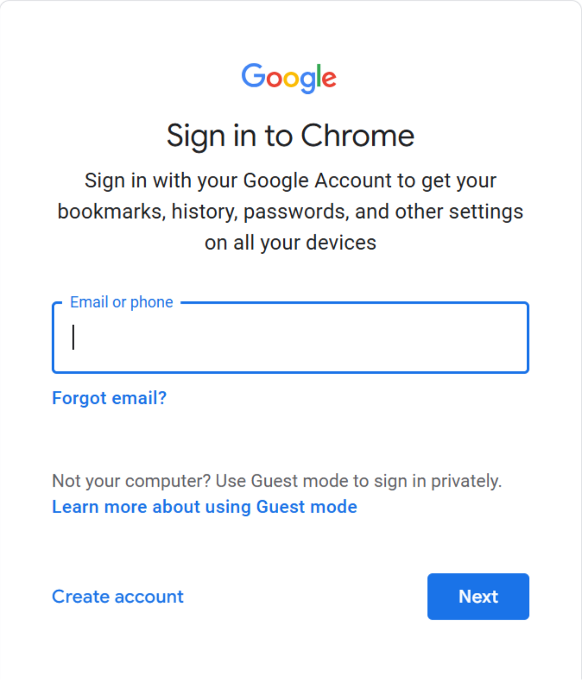
Development Environment and Tools

The research was carried out in Python on Google Colab with GPU. Key libraries were Pandas and NumPy (Pandas, n.d.; NumPy, n.d.) for data analysis, Matplotlib and Seaborn (Matplotlib, n.d.; Seaborn, n.d.) for the creation of visualizations, Scikit-learn and XGBoost (Scikit-learn, n.d.) for ML and TensorFlow/Keras (TensorFlow, n.d.; Keras, n.d.) for deep learning models. The cloud environment of Google Colab allowed us to train and experiment effectively and convenient storage integration with Google Drive.

Setup Procedure and Implementation Workflow

Step 1: Set Up a Google Account

Begin by creating a Google account, this will give access to Google Drive, Google Colab, and other essential tools. Visit https://accounts.google.com to sign up.



Google

Sign in to Chrome

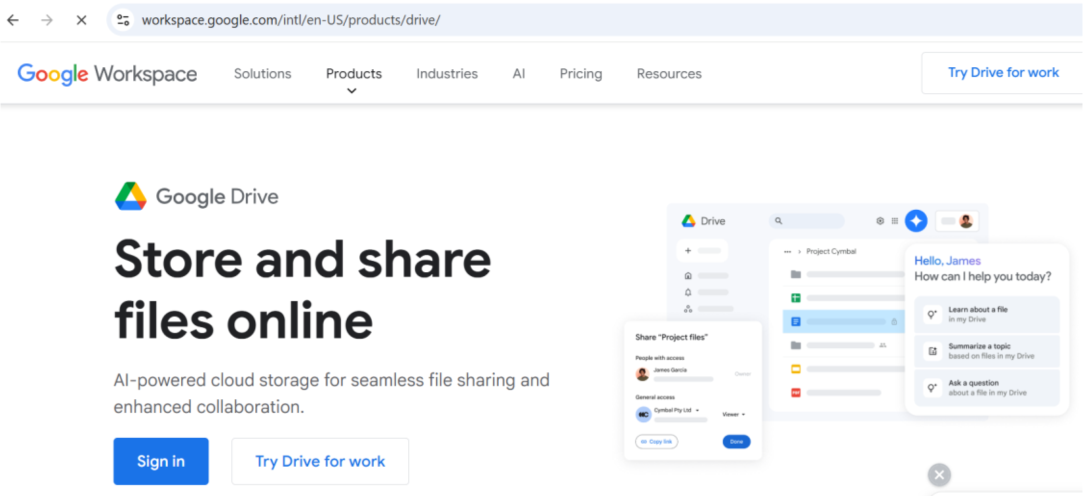
Sign in with your Google Account to get your bookmarks, history, passwords, and other settings on all your devices

Not your computer? Use Guest mode to sign in privately.

Step 2: Access Google Drive

Log in to https://drive.google.com and organize workspace:

- Create a new folder, **"Evaluating the Effectiveness of Neural Networks for Cyber Attack Detection in IoT Ecosystems"**.
- Upload dataset file and any other supporting files into this folder.



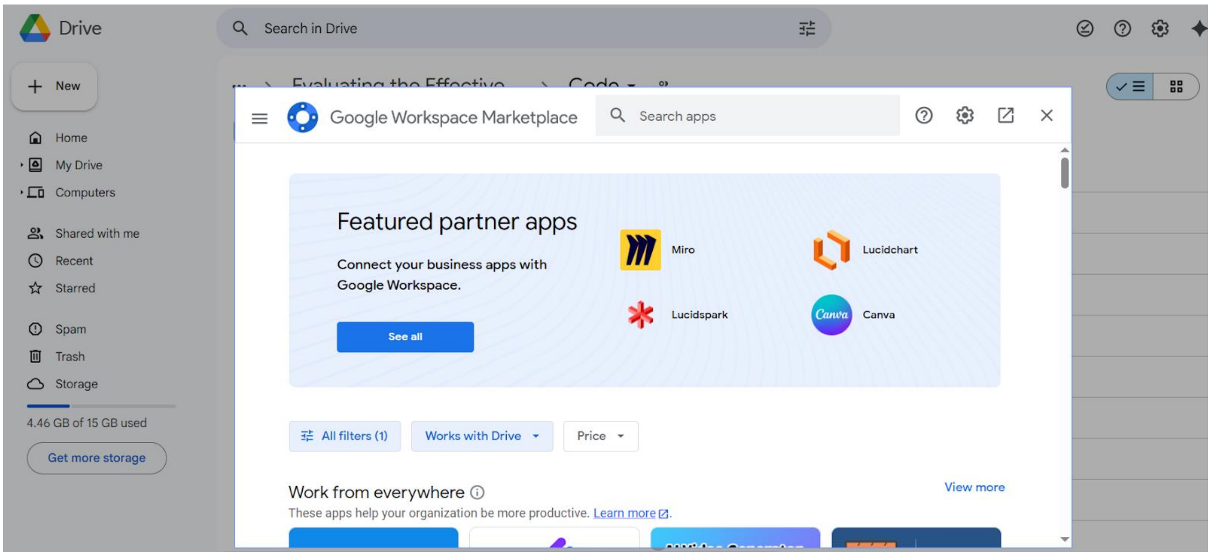
Store and share files online

AI-powered cloud storage for seamless file sharing and enhanced collaboration.

Step 3: Install google Collaboratory

To write and run Python code in the cloud:

- In Google Drive, click **New > More > Connect more apps**.
- Search for **Colaboratory** and install it.
- Once installed, open. ipynb notebook files directly from Google Drive using Colab.

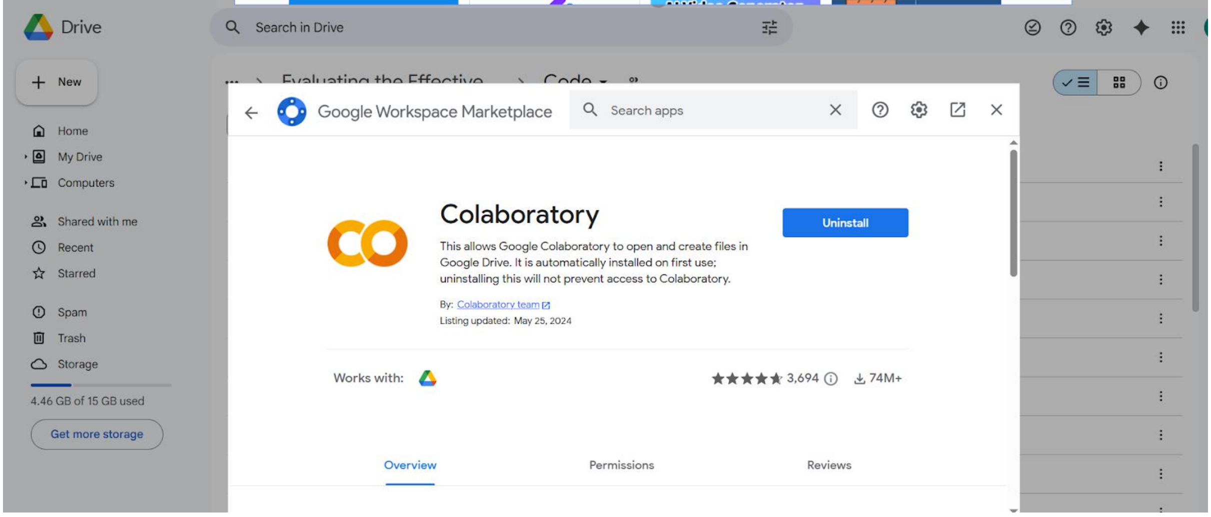


Featured partner apps

Connect your business apps with Google Workspace.

Work from everywhere

These apps help your organization be more productive. Learn more.



Colaboratory

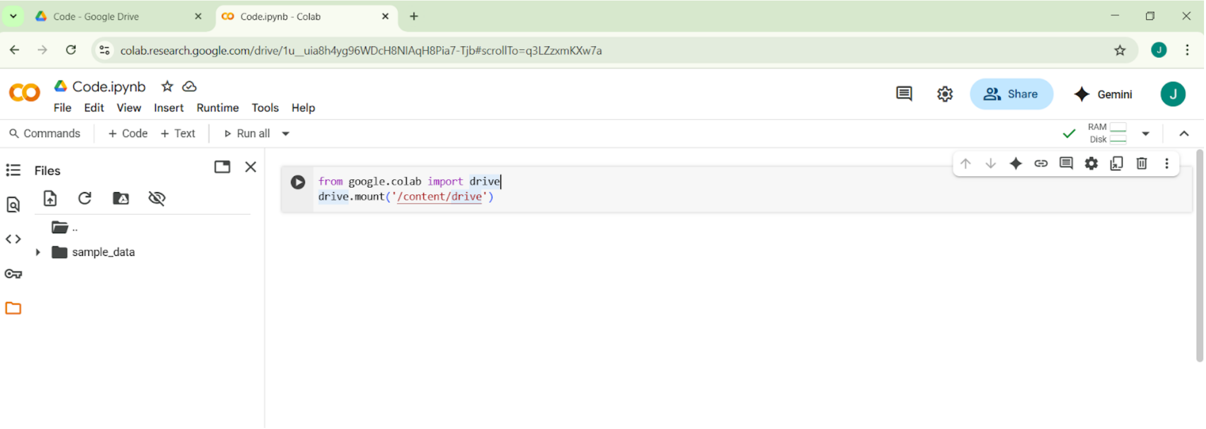
This allows Google Colaboratory to open and create files in Google Drive. It is automatically installed on first use; uninstalling this will not prevent access to Colaboratory.

By: Colaboratory team

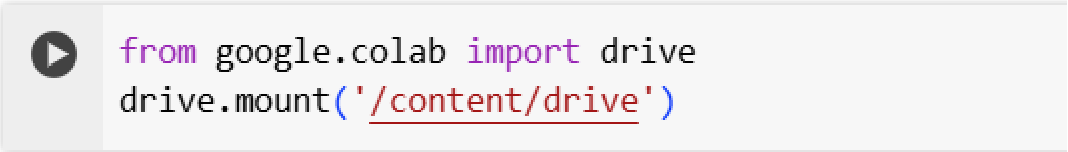
Listing updated: May 25, 2024

Step 4: Create new Notebook & Mount a Drive

- Click New→More → Google Colaboratory
- And mount with google drive

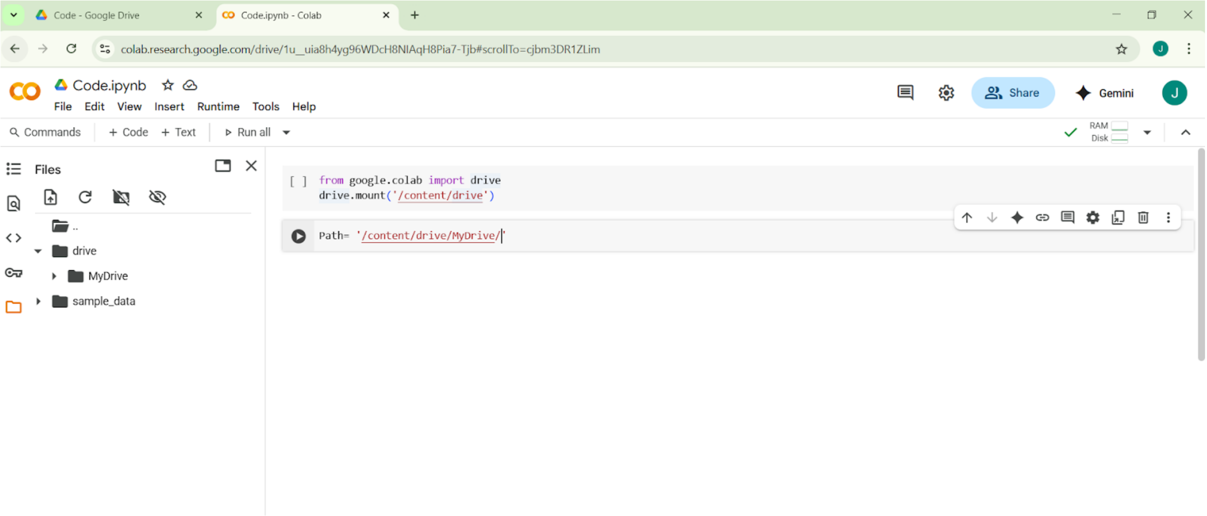


```
from google.colab import drive
drive.mount('/content/drive')
```



```
from google.colab import drive
drive.mount('/content/drive')
```

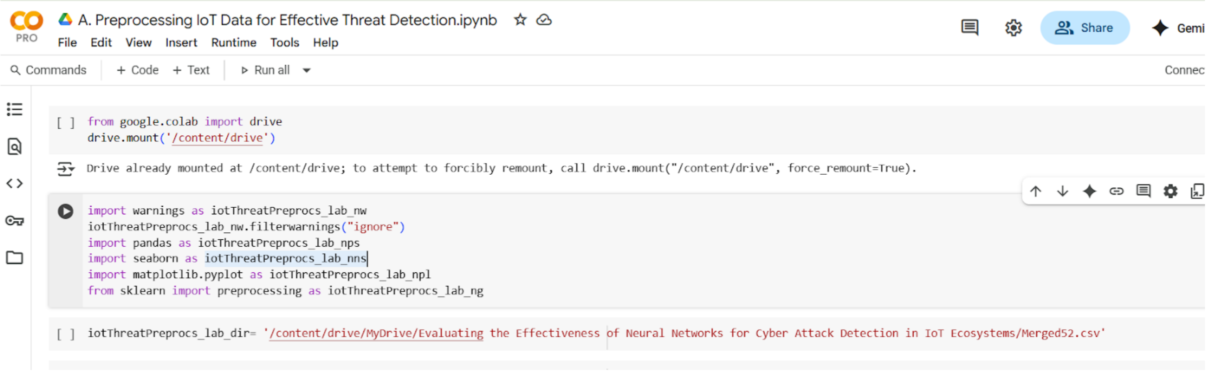
Step 5: Set a path and Adding, Running Code or Text Cells



Code - Google Drive
Code.ipynb - Colab
colab.research.google.com/drive/1u_uia8h4yg96WDcH8NIAqH8Pia7-Tjb#scrollTo=cjbm3DR1ZLim
Code.ipynb
File Edit View Insert Runtime Tools Help
Share
Gemini
Commands
+ Code + Text
Run all
RAM
Disk
Files
..
drive
MyDrive
sample_data

```
[ ] from google.colab import drive
    drive.mount('/content/drive')

Path= '/content/drive/MyDrive/'
```



A. Preprocessing IoT Data for Effective Threat Detection.ipynb
PRO
File Edit View Insert Runtime Tools Help
Share
Gemi
Commands
+ Code + Text
Run all
Connec

```
[ ] from google.colab import drive
    drive.mount('/content/drive')

Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_remount=True).

import warnings as iotThreatPreprocs_lab_nw
iotThreatPreprocs_lab_nw.filterwarnings("ignore")
import pandas as iotThreatPreprocs_lab_nps
import seaborn as iotThreatPreprocs_lab_nns
import matplotlib.pyplot as iotThreatPreprocs_lab_npl
from sklearn import preprocessing as iotThreatPreprocs_lab_ng

[ ] iotThreatPreprocs_lab_dir= '/content/drive/MyDrive/Evaluating the Effectiveness of Neural Networks for Cyber Attack Detection in IoT Ecosystems/Merged52.csv'
```

Step 6: Adding and Running Code or Text Cells

Use the **+ Code** button to insert new cells for Python code, and **+ Text** for documentation and explanations.

- Keep code organized by using headings and comments.
- Divide notebook into sections like: *Data Preprocessing*, *Model Building*, *Training and Evaluation*.

Use TensorFlow/Keras to design and train model: Save trained model for future use. You can also export graphs or processed data to Google Drive.

Code Execution Process

The code execution workflow in Google Colab integrates seamlessly with Google Drive for persistent storage. Initially, Drive is mounted using the google.colab.drive module, and the dataset path is specified. Pandas is used to load data into memory, followed by sequential execution of preprocessing, training, and evaluation blocks. Documentation is maintained in text cells to ensure clarity and reproducibility. Model results, visualisations, and processed data are saved back to Drive, ensuring no progress is lost due to Colab's session resets. This structure supports collaboration by enabling shared access to all project files in the Drive folder.

References

Pandas, n.d. *Pandas: Python Data Analysis Library*. Version 2.2.2. Available at: https://pandas.pydata.org/

NumPy, n.d. *NumPy*. Version 2.3.2. Available at: https://numpy.org/

Matplotlib, n.d. *Matplotlib: Visualization with Python*. Version 3.9.2. Available at: https://matplotlib.org/stable/

Seaborn, n.d. *Seaborn: Statistical Data Visualization*. Version 0.13.2. Available at: https://seaborn.pydata.org/

Scikit-learn, n.d. *Machine Learning in Python*. Version 1.7.1. Available at: https://scikit-learn.org/stable/

TensorFlow, n.d. *TensorFlow*. Version 2.17.0. Available at: https://www.tensorflow.org/

Keras, n.d. *Keras API reference*. Version 3.3.3. Available at: https://keras.io/