

# **Configuration Manual**

MSc Research Project MSc Cyber Security

Ankit Kantilal Patel Student ID: X21101850

School of Computing National College of Ireland

Supervisor: Prof. Jawad Salahuddin

#### National College of Ireland



#### **MSc Project Submission Sheet**

	School of Computing	
	Ankit Kantilal Patel	
Student Name:		
	x21101850	
Student ID:		
	MSc Cyber Security	2022
Programme:	MSc Research Project Year:	······
Module:	-	
	Jawad Salahuddin	
Lecturer: Submission Due Date:	15th December 2022	
	Detecting Malicious URL using Extreme Learning Ma	achine Algorithm
<b>Project Title:</b>		
-	678 7	
Word Count:	Page Count:	

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.

<u>ALL</u> 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	Ankit Kantilal Patel
Signature.	14th December 2022
Date:	

#### 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**

#### Ankit Kantilal Patel Student ID: x21101850

### **1** Introduction

The manual documents about all the necessary tools and the technologies required to implement the research model. The manual is further divided into various sections. Section 2 mentions about the environment setup required, Section 3 mentions about the tools and software's used, Section 4 mentions about implementing the project, and in the last references are mentioned.

#### 2 Environment Setup

Below mentioned configuration was used to implement the model.

- Processor: Intel i7
- Memory: 16GB RAM
- Programming language: Python3
- Python Environment: Jupyter Notebook, Anaconda Navigator

### **3** Tools and Software Used

Below mentioned software were used to implement the model.

• For programming Language, we have used python 3.11.0.



**Figure 1. Python Version** 

• For dataset processing and performing operations on it, we have used a python open source project called jupyter. It uses python kernel for processing and performing operations on the datasets.



**Figure 2. Python Jupyter** 

### 4 Implementation of the Model

**Step 1 :-** We have used anaconda navigator, to use python jupyter notebook. First, we need to install an anaconda navigator. Anaconda Navigator [1] can be installed from the website mentioned in the reference. We launch the navigator, and we can see jupyter in the main dashboard then we click on launch to launch jupyter notebook as shown in below Figure 3.



Figure 3. Anaconda Navigator

**Step 2 :-** Open the file "Malicious URL Detection Using ELM Model.ipynb" in the jupyter notebook.

Malicious URL D	etection Using E x +		
$\leftrightarrow$ $\rightarrow$ G (C	localhost 8888/notebooks/Malicious%20URL%20Detection%20Using%20ELM%20Model.ipynb		🖻 🛧
	Jupyter Malicious URL Detection Using ELM Model Last Checkpoint: 20 hours ago (autosaved)		P Logout
	File Edit View Insert Cell Kernel Widgets Help	Not Trusted	Python 3 (ipykernel) O
	E) + x 2 1 1 + v + w + Run ■ C >> Markdown ∨		

Figure 4. Ipynb file launched in Jupyter

**Step 3 :-** From here we start the implementation of the model, we start by importing all the necessary libraries for the model.

Addictions URL Detection	tsing 🗈 🗙 🕂				~	-	σ	×
$\leftrightarrow \rightarrow \mathbf{C}$ (0 locally	st:8888/notebooks/Malicious%20URL%20Detection%20Using%20ELM%20ModeLipynb				\$		• (8	:
	C JUPYTET Malicious URL Detection Using ELM Model Last Checkpoint Yesterday at 5.32 PM (autosaved)			Logout				î
	File Edit View Insert Cell Kernel Widgets Help	Not Trusted	Python 3 (ip	ykernel) O				-1
	E + ೫ 2 1 1 + ↓ ► Run ■ C > Markdown ∨ EB							. 1
	Loading all the important libraries In [164]: import pandas as pd import increation of the import as plt import matplotlib.pyplot as plt import of the import sealor as as a import sealorn as as as import sealorn as insolved the import train_test_split from willib.parse import tableEncoder from sklearn.preprocessing import tableEncoder from sklearn.preprocessing import tableEncoder from sklearn.preprocessing import tableEncoder from sklearn.metrics import classification_report,confusion_matrix, accuracy_score, plot_roc_curve from tild import get_tid import os.path	e, plot_confus	ion_matrix					

**Figure 5. Libraries Imported for the Model** 

**Step 4 :-** We load the dataset [2] using pandas libraries.

Malicious URL Detection Using E ×	+						~	8 °C		0	×
← → C O localhost88888/not				Model.ipynb		₫ \$			• 🗖	۲	
💭 јир	yter	Malic	ous URL Detection Using ELM M	odel Last Ch	eckpoint: Yesterday at 5:32 PM (autosaved)	at					Î
File	Edit \	/iew	Insert Cell Kernel Widgets He	lp	Not Trusted Python 3 (ipykernel)	0					
<b>B</b> +	> ह	в	↑ ↓ ► Run ■ C → Markdown	~ 📼							
											*
Ir	n [2]: (	lf=pd.r	<pre>read_csv('malicious_phish.csv')</pre>								ч
			The Dataset Loaded								
Ir	n [3]: 0	lf									
OL	ut[3]:		uri	type							
		0	br-icloud.com.br	phishing							
		1	mp3raid.com/music/krizz_kaliko.html	benign							
		2	bopsecrets org/rexroth/cr/1.htm	benign							
		3	http://www.garage-pirenne.be/index.php?option=	defacement							
		4	http://adventure-nicaragua.net/index.php?optio	defacement							
		651186	xbox360.ign.com/objects/850/850402.html	phishing							
		651187	games.teamxbox.com/xbox-360/1860/Dead-Space/	phishing							
		651188	www.gamespot.com/xbox360/action/deadspace/	phishing							
		651189	en.wikipedia.org/wiki/Dead_Space_(video_game)	phishing							
		651190	www.angelfire.com/goth/devilmaycrytonite/	phishing							
		651191	rows × 2 columns								

Figure 6. Loading the Dataset in the Model



**Step 5 :-** Features extracted from the dataset.

**Figure 7. Dataset Features Extracted** 

Step 6 :- Enter URLs in the urls array to be detected as malicious and the model detects it.

	URLs given to detect them
In [171]:	<pre>urls = ['titaniumcorporate.co.za','en.wikipedia.org/wiki/North_Dakota']</pre>
In [172]:	<pre>for url in urls: print(get_prediction_from_url(url))</pre>
	MALWARE SAFE

**Figure 8. Final Detection of URL** 

**Step 7 :-** Code in Jupyter can be executed by using Run button.

S Malcious UBL Detection Using 1 x +	- 0	×
🔄 🔶 C 🕐 localhost 8888/notebooks/Maliciour%20UKing%20ELM%20Model izynb 😰 🖈 🔳	* 🗆 🤅	e) :
💭 jupyter Malicious URL Detection Using ELM Model Last Checkpoint Yesterday at 5.32 PM (autosaved) 🐣 Logost		Â
File Edit View Insert Cell Kernel Widgets Help Nat Trusted Python 3 (pykernel) O		
Loading all the important libraries		Î
<pre>In [164]: import many as pd import many as np import itertools import itertools import itertools import os import os import os import os import itertools from sklearn.model_selection import train_test_split from onlib.parse import uniparse from sklearn.preprocessing import LabelEncoder from sklearn.preprocessing import LabelEncoder from tid import settid import os.path import import im from numpy.innaig import in from numpy.innaig import in from tid import tosification_report,confusion_matrix, accuracy_score, plot_roc_curve, plot_confusion_matrix from tid import os.path import os.path</pre>	La	
In [2]: df=pd.read_csv('malicious_phish.csv')		

**Figure 9. Code Execution in Jupyter** 

#### References

[1] "Anaconda | Anaconda Distribution," Anaconda. https://www.anaconda.com/products/distribution (accessed Dec. 13, 2022).
Feng, G. and Buyya, R. (2016). Maximum revenue-oriented resource allocation in cloud, *IJGUC* 7(1): 12–21.

[2] "Malicious URLs dataset." https://www.kaggle.com/datasets/sid321axn/malicious-urls-dataset (accessed Dec. 06, 2022).