Configuration Manual

MSc Research Project
Programme Name

Musa Aboki
Student ID: X20218061

School of Computing
National College of Ireland

Supervisor: Ross Spelman

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name: Musa Idisere Aboki
………

Student ID: X20218061
………

Programme: Master of Science in Cyber Security ………………………………………………………………… **Year:** 2021-2022 ……………………………..

Module: ………

Lecturer: Ross Spelman
………

Submission Due Date: ………

Project Title: Towards improved phishing detection from URLs, using supervised machine learning
………

Word Count: 734 ……………………………………… **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: Musa Idisere Aboki
……

Date: 1st /02/2023
……

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

Musa Aboki
Student ID: X20218061

1 Introduction to Configuration Manual

The manual outlines the Software and tools used to implement the Project. The manual contains the steps and instructions used to install the required software to implement the project and the systems used to get the required results.

2 Hardware Specification Details

Due to the high resource need and requirements for the Machine Learning algorithm and process. The Hardware specification is outlined below-:

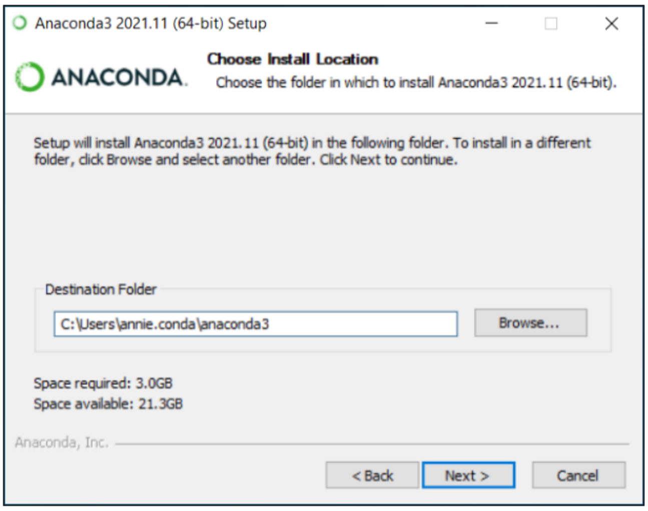
Hardware

Technical Environment Hardware

Physical Device	
Device Model	Dell Inspiron Build Machine
Processor	Intel(R) Core (TM) i5-3570 CPU @ 3.40GHz 3.40 GHz
Installed RAM	32.0 GB RAM
System Type	Windows Operating System
HardDrive	500GB SSD Hard drive

Software Specification

Installed Anaconda for Windows using the below steps. Anaconda was the preferred method as it came added with the jupyter Notebook (*Troubleshooting — Anaconda documentation*, no date)



Anaconda3 2021.11 (64-bit) Setup

Choose Install Location

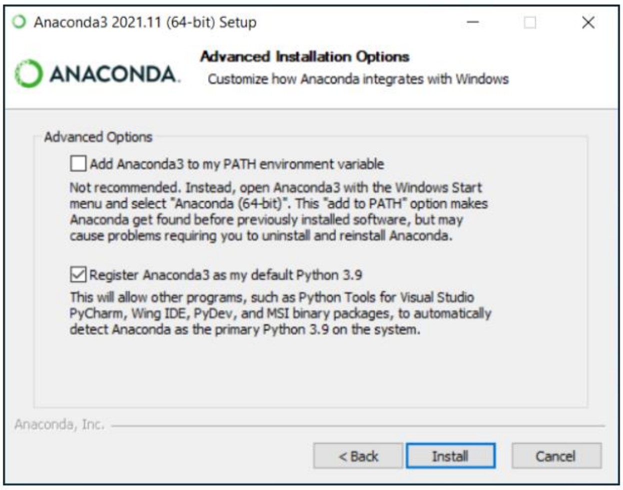
Choose the folder in which to install Anaconda3 2021.11 (64-bit).

Setup will install Anaconda3 2021.11 (64-bit) in the following folder. To install in a different folder, click Browse and select another folder. Click Next to continue.

Destination Folder: C:\Users\annie.conda\anaconda3

Space required: 3.0GB

Space available: 21.3GB



Advanced Installation Options

Customize how Anaconda integrates with Windows

Advanced Options

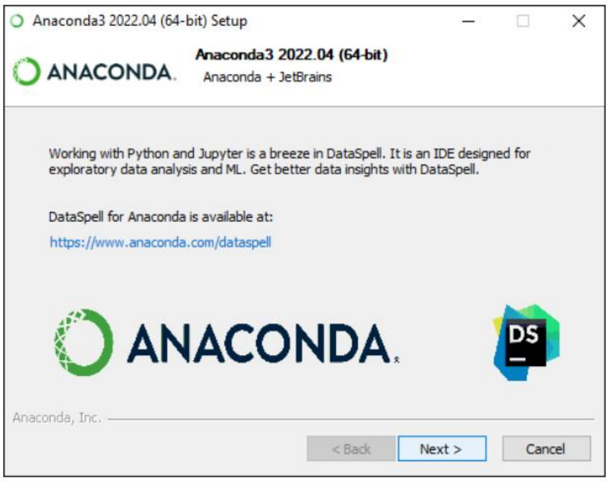
□ Add Anaconda3 to my PATH environment variable

Not recommended. Instead, open Anaconda3 with the Windows Start menu and select "Anaconda (64-bit)". This "add to PATH" option makes Anaconda get found before previously installed software, but may cause problems requiring you to uninstall and reinstall Anaconda.

☑ Register Anaconda3 as my default Python 3.9

This will allow other programs, such as Python Tools for Visual Studio PyCharm, Wing IDE, PyDev, and MSI binary packages, to automatically detect Anaconda as the primary Python 3.9 on the system.

Anaconda, Inc.



Anaconda3 2022.04 (64-bit)

Anaconda + JetBrains

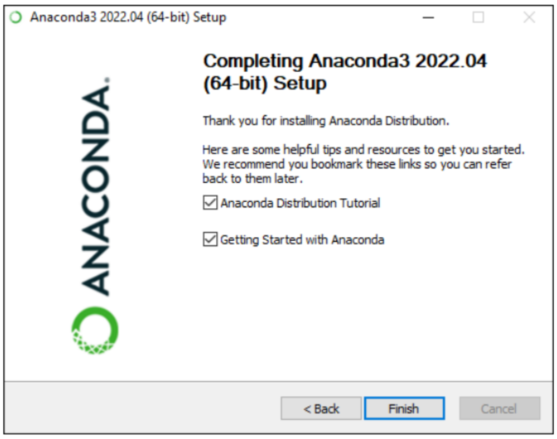
Working with Python and Jupyter is a breeze in DataSpell. It is an IDE designed for exploratory data analysis and ML. Get better data insights with DataSpell.

DataSpell for Anaconda is available at:

https://www.anaconda.com/dataspell

Anaconda, Inc.

12. After a successful installation you will see the "Thanks for installing Anaconda" dialog box:



Completing Anaconda3 2022.04 (64-bit) Setup

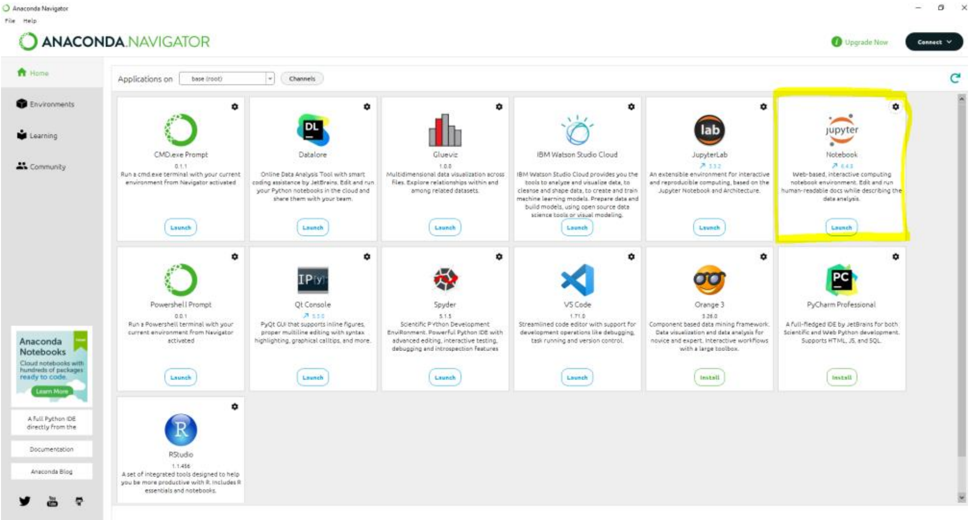
Thank you for installing Anaconda Distribution.

Here are some helpful tips and resources to get you started. We recommend you bookmark these links so you can refer back to them later.

☑ Anaconda Distribution Tutorial

☑ Getting Started with Anaconda

13. If you wish to read more about Anaconda.org and how to get started with Anaconda, check the boxes "Anaconda Distribution Tutorial" and "Learn more about Anaconda". Click the **Finish** button.



Run a cmd.exe terminal with your current environment from Navigator activated

Online Data Analysis Tool with smart coding assistance by JetBrains. Edit and run your Python notebooks in the cloud and share them with your team.

Multidimensional data visualization across files. Explore relationships within and among related datasets.

IBM Watson Studio Cloud provides you the tools to analyze and visualize data, to cleanse and shape data, to create and train machine learning models. Prepare data and build models, using open source data science tools or visual modeling.

An extensible environment for interactive and reproducible computing, based on the Jupyter Notebook and Architecture.

Web-based, interactive computing notebook environment. Edit and run human-readable docs while describing the data analysis.

Run a Powershell terminal with your current environment from Navigator activated

PyQt GUI that supports inline figures, proper multiline editing with syntax highlighting, graphical calltips, and more.

Scientific PYthon Development EnviRonment. Powerful Python IDE with advanced editing, interactive testing, debugging and introspection features

Streamlined code editor with support for development operations like debugging, task running and version control.

Component based data mining framework. Data visualization and data analysis for novice and expert. Interactive workflows with a large toolbox.

A full-fledged IDE by JetBrains for both Scientific and Web Python development. Supports HTML, JS, and SQL.

A set of integrated tools designed to help you be more productive with R. Includes R essentials and notebooks.

3 Setup of the Machine learning jupyter Notebook version 6.4.8

The project is aimed at using python programming language due to the availability of mature and well tested data management and machine learning libraries such as scikit, pandas, NumPy to help analyze and train the dataset. Python has been the preferred programming language due to the availability of these libraries and the ease of using the language. The main object would be to tag either the websites belonging to phishing or legitimate using the machine learning framework. After this, the results would be analyzed in relation to precision, recall, and accuracy. The result would also show the details of the program and how effectively it identifies the websites

3.1 Feature Extraction Procedures

Feature extraction was needed because the original raw data cannot be used for the machine learning model. Most of the useful information is captured as a result of the extracted features. This creates a smaller set of features captured from the raw data. We would extract useful information from the .csv data for the purpose of the machine learning classifiers to work from.

Features Extracted as Outlined Below

Address Bar-based Features

Domain of URL
IP Address in URL
"@" Symbol in URL
getDomainEntropy
Length of URL
Depth of URL
Redirection "//" in URL
"http/https" in Domain name
Using URL Shortening Services “TinyURL”
Prefix or Suffix "-" in Domain Each of these features are explained and the coded below:

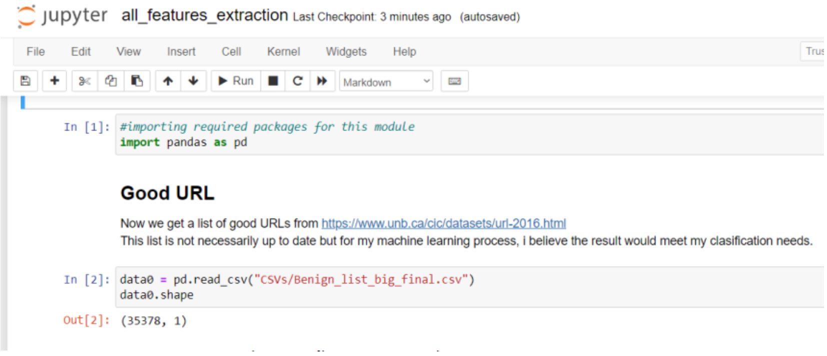
Domain based Features

DNS Record
Website Traffic
Age of Domain
End Period of Domain
Each of these features are explained and the coded below:

HTML & Javascript based Features

IFrame Redirection
Status Bar Customization
Disabling Right Click
Website Forwarding

3.1.1 Good URLs Procedures

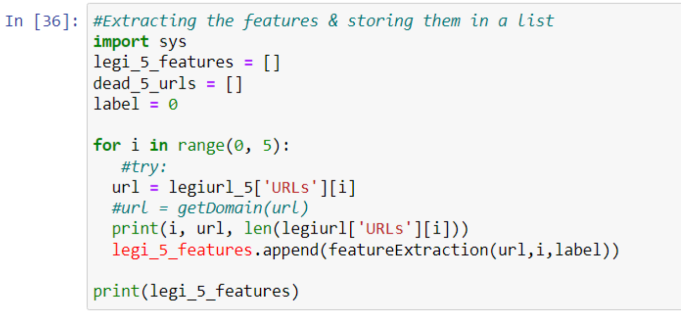

jupyter all_features_extraction Last Checkpoint: 3 minutes ago (autosaved)
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Trus
Run
Markdown

```
In [1]: #importing required packages for this module
        import pandas as pd
```

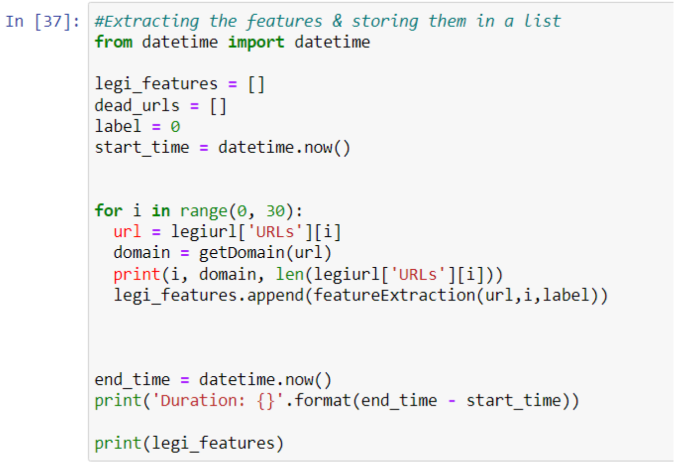
Good URL

Now we get a list of good URLs from https://www.unb.ca/cic/datasets/url-2016.html
This list is not necessarily up to date but for my machine learning process, i believe the result would meet my clasification needs.

```
In [2]: data0 = pd.read_csv("CSVs/Benign_list_big_final.csv")
        data0.shape

Out[2]: (35378, 1)
```

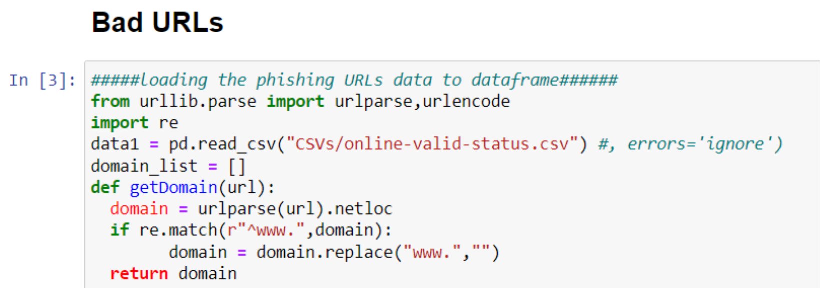


```
In [36]: #Extracting the features & storing them in a list
         import sys
         legi_5_features = []
         dead_5_urls = []
         label = 0

         for i in range(0, 5):
           #try:
          url = legiurl_5['URLs'][i]
          #url = getDomain(url)
          print(i, url, len(legiurl['URLs'][i]))
          legi_5_features.append(featureExtraction(url,i,label))

         print(legi_5_features)
```



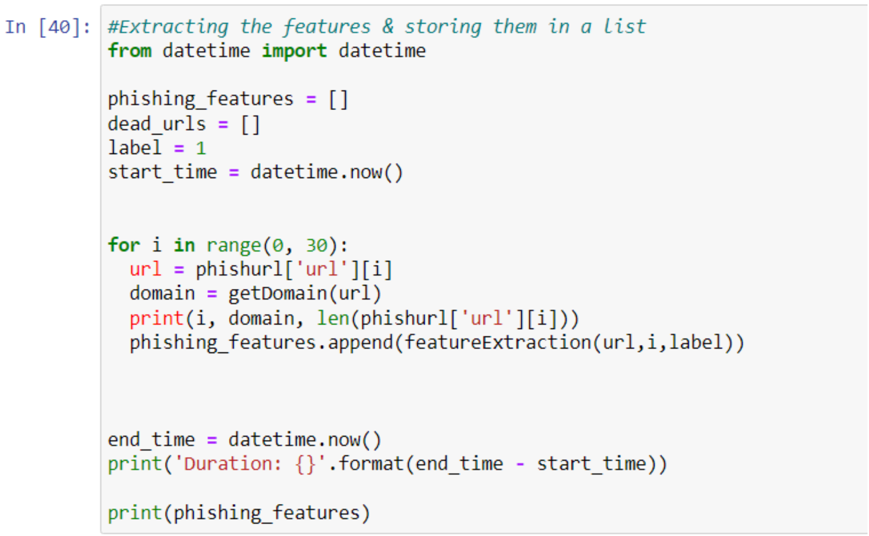
```
In [37]: #Extracting the features & storing them in a list
         from datetime import datetime

         legi_features = []
         dead_urls = []
         label = 0
         start_time = datetime.now()


         for i in range(0, 30):
          url = legiurl['URLs'][i]
          domain = getDomain(url)
          print(i, domain, len(legiurl['URLs'][i]))
          legi_features.append(featureExtraction(url,i,label))



         end_time = datetime.now()
         print('Duration: {}'.format(end_time - start_time))

         print(legi_features)
```

3.1.2 Bad URLs Procedures



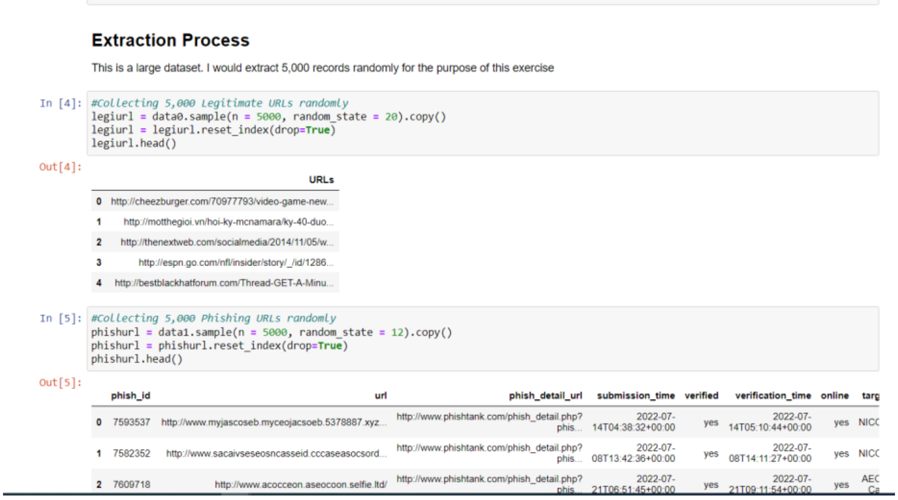
Bad URLs

```
In [3]: #####loading the phishing URLs data to dataframe######
        from urllib.parse import urlparse,urlencode
        import re
        data1 = pd.read_csv("CSVs/online-valid-status.csv") #, errors='ignore')
        domain_list = []
        def getDomain(url):
         domain = urlparse(url).netloc
         if re.match(r"^www.",domain):
            domain = domain.replace("www.","")
         return domain
```



```
In [40]: #Extracting the features & storing them in a list
         from datetime import datetime

         phishing_features = []
         dead_urls = []
         label = 1
         start_time = datetime.now()


         for i in range(0, 30):
           url = phishurl['url'][i]
           domain = getDomain(url)
           print(i, domain, len(phishurl['url'][i]))
           phishing_features.append(featureExtraction(url,i,label))



         end_time = datetime.now()
         print('Duration: {}'.format(end_time - start_time))

         print(phishing_features)
```

3.1.3 *Extracted 5000 records Randomly from both the Good URLs and Bad URLs*



Extraction Process

This is a large dataset. I would extract 5,000 records randomly for the purpose of this exercise

```
In [4]: #Collecting 5,000 Legitimate URLs randomly
        legiurl = data0.sample(n = 5000, random_state = 20).copy()
        legiurl = legiurl.reset_index(drop=True)
        legiurl.head()
```

Out[4]:

	URLs
0	http://cheezburger.com/70977793/video-game-new...
1	http://motthegioi.vn/hoi-ky-mcnamara/ky-40-duo...
2	http://thenextweb.com/socialmedia/2014/11/05/w...
3	http://espn.go.com/nfl/insider/story/_/id/1286...
4	http://bestblackhatforum.com/Thread-GET-A-Minu...

```
In [5]: #Collecting 5,000 Phishing URLs randomly
        phishurl = data1.sample(n = 5000, random_state = 12).copy()
        phishurl = phishurl.reset_index(drop=True)
        phishurl.head()
```

Out[5]:

	phish_id	url	phish_detail_url	submission_time	verified	verification_time	online	targ
0	7593537	http://www.myjascoseb.myceojacsoeb.5378887.xyz...	http://www.phishtank.com/phish_detail.php?phis...	2022-07-14T04:38:32+00:00	yes	2022-07-14T05:10:44+00:00	yes	NICC
1	7582352	http://www.sacaivseseosncasseid.cccaseasocsord...	http://www.phishtank.com/phish_detail.php?phis...	2022-07-08T13:42:36+00:00	yes	2022-07-08T14:11:27+00:00	yes	NICC
2	7609718	http://www.acocceon.aseocoon.selfie.ltd/	http://www.phishtank.com/phish_detail.php?phis...	2022-07-21T06:51:45+00:00	yes	2022-07-21T09:11:54+00:00	yes	AEC Ca

References

Troubleshooting — Anaconda documentation (no date). Available at: https://docs.anaconda.com/anaconda/user-guide/troubleshooting/ (Accessed: 30 October 2022).