

# How to run my ARE code?

Muhammad Haseeb

2020-10-192

## File Structure:

The submitted folder “Project” contains three folders:

- Analysis
- mod\_1\_web\_crawler\_and\_contexter
- mod\_2\_corpus\_and\_rule\_based\_der

The last two folders are of ARE implementation.

**mod\_1\_web\_crawler\_and\_contexter** contains:

- web\_crawler.py
- output.txt (may or may not be present)

**mod\_2\_corpus\_and\_rule\_based\_der** contains:

- device\_types
- vendors
- index.js
- find\_device\_info.py
- ind.json (may or may not be present)
- Output/ (a folder which may or may not have a output.txt file within it)

## Running the Code:

### Step 1:

First of all, one needs to run the web crawler (the contexter is integrated in this crawler so there is no need to separately run the code for contexter).

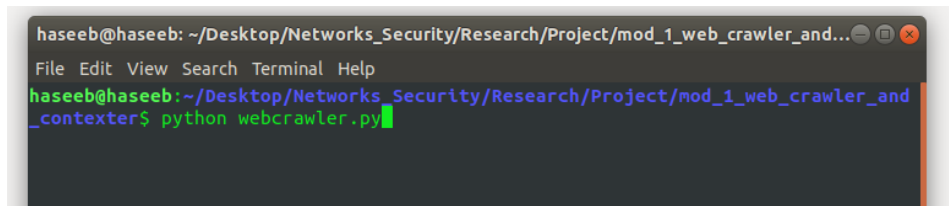
For this, go to the first directory of ARE **mod\_1\_web\_crawler\_and\_contexter** and give the input by changing the seventh line. Example is shown in screenshot:

```

3 import time
4 from urllib.request import Request, urlopen
5
6
7 query = "220 Welcome to ASUS RT-AC58U FTP service.\r\n"
8
9 # query = '220 ZXR10 ftp service ready for new user. '
10 #This is the input to my program.

```

Then run the following command:



```

haseeb@haseeb: ~/Desktop/Networks_Security/Research/Project/mod_1_web_crawler_and...
File Edit View Search Terminal Help
haseeb@haseeb:~/Desktop/Networks_Security/Research/Project/mod_1_web_crawler_and...
_contexter$ python webcrawler.py

```

(python webcrawler.py)

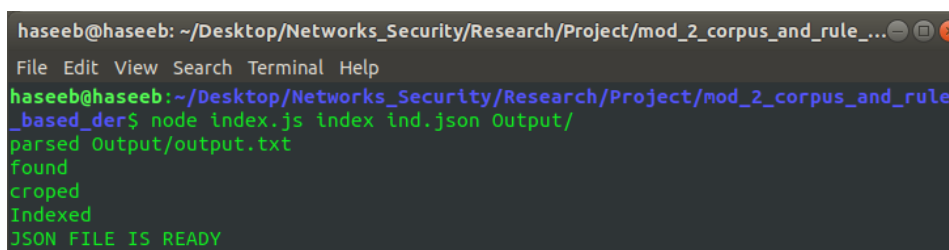
This will produce a file output.txt. This is the file from which DER will take input and predict labels. Copy this file as we will paste it in another folder.

### Step 2:

Now, go to second folder of ARE **mod\_2\_corpus\_and\_rule\_based\_der**. Go further into the folder Output and paste the copied file output.txt here. Now get out of this Output folder. Now you are back into folder **mod\_2\_corpus\_and\_rule\_based\_der**.

### Step 3:

Now we are going to index the output we have pasted in Output folder. For this run the following command:



```

haseeb@haseeb: ~/Desktop/Networks_Security/Research/Project/mod_2_corpus_and_rule...
File Edit View Search Terminal Help
haseeb@haseeb:~/Desktop/Networks_Security/Research/Project/mod_2_corpus_and_rule...
_based_der$ node index.js index ind.json Output/
parsed Output/output.txt
found
cropped
Indexed
JSON FILE IS READY

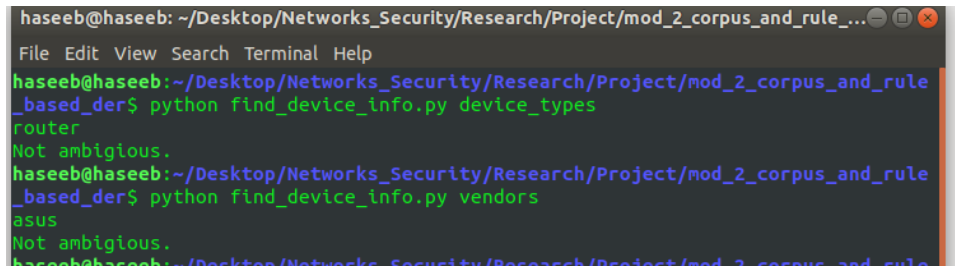
```

(node index.js index ind.json Output/)

This will create a file ind.json or overwrite it if already present.

#### **Step 4:**

Now run the following commands and see the predicted labels:



```
haseeb@haseeb: ~/Desktop/Networks_Security/Research/Project/mod_2_corpus_and_rule_...
File Edit View Search Terminal Help
haseeb@haseeb:~/Desktop/Networks_Security/Research/Project/mod_2_corpus_and_rule_...
based_der$ python find_device_info.py device_types
router
Not ambiguous.
haseeb@haseeb:~/Desktop/Networks_Security/Research/Project/mod_2_corpus_and_rule_...
based_der$ python find_device_info.py vendors
asus
Not ambiguous.
haseeb@haseeb:~/Desktop/Networks_Security/Research/Project/mod_2_corpus_and_rule_...
```

(python find\_device\_info.py device\_types)

(python find\_device\_info.py vendors)

The predicted labels can be seen on the terminal (as shown in attached screenshot, the predictions are “router” and “asus”).

### **Important points**

- Python3 is used for this code.
- vendors file can be edited and added more vendors for better efficiency.
- device\_types file can be edited and added more device types for better efficiency