

Application Building

Build the python Flask app

```
import ipaddress
import re
import urllib.request from bs4 import BeautifulSoup
import socket import requests
from googlesearch import search
import whois
from datetime import datetime
import time
from dateutil.parser import parse as date_parse
#Calculates number of months

def diff month(d1, d2): return (d1.year - d2.year)

#Generate data set by extracting the features from the URL
def generate_data_set(url):
    data_set = []

    # Converts the given URL into standard format if not re.match(r"\"https?\", url):

    url = "http://" + url
```

Stores the response of the given URL

try:

response requests.get(url) soup BeautifulSoup (response.text, 'html.parser')

except:

response

soup-999

* Extracts domain from the given URL

domain- re.findall(e"://([^/]+)/?", url)[0] if re.match(".", domain):

domain- domain.replace("Mr.", "")

Requests all the information about the domain whois_response whois.whois
(domain)

rank_checker response

requests.post("https://www.checkpagerank.net/index.php", { "name":
domain})

#Extracts global rank of the website

try:

global_rank = int(re.findall(r"Global Rank: ([0-9]+)",
rank_checkerresponse.text)[0])

ipaddress.ip_address(url)

```
data_set.append(-1)
```

```
except:
```

```
data set.append(1)
```

2.URL Length

```
if len(url) < 54:
```

```
    data set.append(1)
```

```
elif len(url)
```

```
    54 and len(url) < 75:
```

```
data set.append(0)
```

```
else:
```

```
data set.append(-1)
```

3.Shortining Service

```
matchere.search("bit.ly/goo.gl/shortel.st/go21\ink|x\,colou\ly/t.co/tinyurl|tr  
\.in/is.gd/cli\pyfrog.com/igre\self\in/tiny.cc/ur14\eu/twit\ac/sul.pr|tuur1.  
nl/snipurl.com/  
short\to/BudURL.com/ping\fm/post\ly/Just\es/bkite\com/snipr\com/fic\k  
r|loopt\usdoiop\com/short\ie/kl\|up\ae/rubyurl.com/on\ly/tal.ly/bit.do/t.  
co/lnkd\in]db\tt/gr\ae/adf.ly/goo.gl/bitly.com/cur\lv/tinyurl.com/ow.ly/bit.l  
y|ity\im|generate dataset)
```

```
    elif
```

```
    lentre.findall())...gs/is.gd/po.st/bc.vc/twitthis\com/ul.to/j.mp/buzurl\c  
om/cutt\us/u\bblyourls\org/
```

```
    x.co/prettylinkpro\com/scrnch\ne|filoops\info/vzturl\com/qr\net|1u  
rl.com/tweez\me/v\gd/tr\in|Link\zip\net", url)
```

```
if match:
```

```
    data_set.append(-1)
```

else:

data_set.append(1)

#4. having At Symbol if re.findall("@", url):

data_set.append(-1)

else:

data set.append(1)

#5.double slosh_redirecting

list [x.start(0) for x in re.finditer('//', url)]

if list[len(list)-1]>6: data_set.append(-1)

else:

data_set.append(1)

6. Prefix Suffix

if re.findall(r"https?://[^\-]+[^\-]+/", url): data set.append(-1)

else:

data_set.append(1)

#7, having Sub Domain

if len(re.findall("\.", url)) = 1: data set.append(1) elif len(re.findall(".", url)) - 21

data set.append(e)

```
else:
```

```
data set.append(-1)
```

```
# 8final State
```

```
try:
```

```
if response.text:
```

```
    data set.append(1)
```

```
except:
```

```
    data set.append(-1)
```

```
#9.Domain_registration Length
```

```
expiration_date whois_response.expiration_date registration_length - 8
```

```
try:
```

```
    expiration date- min(expiration_date)
```

```
    today time.strptime("%Y-%m-%d')
```

```
    today datetime.strptime(today, "Y-%m-%d') registration_length
```

```
    abs((expiration_datetoday).days)
```

```
if registration_length / 365 <- 1:
```

```
data set.append(-1)
```

```
else:
```

```
    data set.append(1)
```

except:

data set.append(-1)

10. Favicon

if soup -999: data set.append(-1)

else:

try:

for head in soup.find_all("head"):

for head.link in soup.find_all('link', href=True):

dots [x.start(e) for x in re.finditer("\.",head.link['href'])]

if url in head.link['href'] or len(dots) 1 or domain in head. link["href"]:

data set.append(1)

raise StopIteration

else:

data set.append(-1)

raise StopIteration

except :

StopIteration: pass

#11. Port

try:

port domain.split(":")[1]

if port:

data_set.append(-1)

```
else:
data set.append(1)
except:
    data set.append(1)
```

#12. HTTPS_token

```
if re.findall(r"\"https://\", url):
data set.append(1)
else:
    data set.append(-1)
```

#13. Request URL

```
success
if soup-999:
    data set.append(-1)
else:
    for ing in soup.find_all('Ing', srce True): dots- [x.start(8) for x in
re.finditer("\.,ing['src'])]
if url in ing['src'] or domain in ['src'] or len(dots)==1:
success - success + 1
    for audio in soup.find_all("audio", srce True): dets [x.start(0) for x in
re.finditer(".", audio["src"])]
if url in audio['src'] or domain in audie['src'] or len(dots)-l
    success success + 1
```

```

        for embed in soup.find_all("embed", srce True): dots=[x.start() for x in
        re.finditer(".", embed["src"])]

if url in embed["src"] or domain in embed['src'] or len(dots)==11

    success success + 2

    i=i+1


for iframe in soup.find_all('iframe', src= True):


dots [x.start(0) for x in re.finditer('\.', iframe['src'])] if url in iframe['src'] or
domain in iframe['src'] or len(dots)==1:

    success success + 1

    i=i+1


try:
percentage = success/float(i)
if percentage < 22.0 :
    dataset.append(1)
elif((percentage >= 22.0) and (percentage < 61.0)) :
else:
    data_set.append(e)
    data_set.append(-1)
except:
    data_set.append(1)

```


#14. URL of Anchor

```
percentage=0
```

```
i=0
```

```
unsafe=0
```

```
if soup-999:
```

```
    data_set.append(-1)
```

```
else:
```

```
    for a in soup.find_all('a', href=True):
```

2nd condition was "JavaScript ::void(0)" but we put JavaScript because the space between javascript and might not be there in the actual of 'href']

```
if "a" in a["href"] or "javascript" in a["href"].lower() or "mailto" in  
a["href"].lower() or not (url in a["href"] or domain in a["href"]):
```

```
    i=i+1
```

```
try:
```

```
    percentage = unsafe / float(i) * 100
```

```
    phishing_detection.py
```

```
    forest.py
```

```
except:
```

```
    data_set.append(1)
```

```
if percentage < 31.0:
```

```
    data_set.append(1)
```

```
elif ((percentage > 31.0) and (percentage < 67.0)):
```

```
    data_set.append(0)
```

```
else:
```

```
    data set.append(-1)
```

```
#13. Request URL
```

```
i = 0
```

```
success
```

```
if soup-999:
```

```
    data_set.append(-1)
```

```
else:
```

```
    for ing in soup.find_all('img src= True):
```

```
        dots [x.start(e) for x in re. finditer(\., img['src'])] if url in img['src'] or  
        domain in img['src'] or len (dots)==1: success = success + 1
```

```
    i=i+1
```

```
for audio in soup.find_all('audio, src- True):
```

```
    dots [x.start(e) for x in re.finditer('\., audio['src'])] if url in audio['src'] or  
    domain in audio['src'] or len(dots)==1: i=i+1
```

```
success = success + 1
```

```
for embed in soup.find_all('embed', src= True):
```

```
    dots=[x.start(0) for x in re.finditer('\.', embed['src'])]
```

```
    if url in embed['src'] or domain in embed['src'] or len(dots)==1:
```

```
        success success + 1-1+1
```

```

for iframe in soup.find_all('iframe', srce True):
    dots=[x.start(0) for x in re.finditer("\.", iframe['src'])]
    if url in iframe['src'] or domain in iframe['src'] or len (dots)--1:
        success success + 1
    i-i+1
try:
    percentage = success/float(i) ● 100
if percentage < 22.0:
    dataset.append(1)
elif((percentage >= 22.0) and (percentage < 61.0)) :
    data_set.append(0)
else:
    data_set.append(-1)
except:
    data set.append(1)

```

#14. URL of Anchor

```

percentage = 0
i = 0
unsafe=0
if soup == -999:
    data set.append(-1)

```

else:

for a in soup.find_all('a', href=True):

if "a" in al "href"] or "javascript" in at 'href'].lower() or "mailto" in a['href'],
lower() or not (url in s['href']) or domain in all unsafe- unsafe 1

try:

percentage unsafe / float(i) = 100)

except:

data_set.append(1)

if percentage < 31.0:

data_set.append(1)

elif ((percentage >= 31.8) and (percentage < 67.8)):

data_set.append(e)

else:

data set.append(-1)

#15. Links in togs

i = 0

success -0

if soup-999:

data set.append(-1)

else:

for link in soup.find_all('link', href= True): dots [x.start(e) for x in re.finditer("\",
link['href'])]

```
if url in link['href'] or domain in link['href'] or len(dots)==1: success success +1  
i-i+1
```

```
for script in soup.find_all('script, srce True):  
dots=[x.start(8) for x in re.finditer('\.,script['src']))]
```

getInput.html

```
if url in script['src'] or domain in script['src'] or len(dots)==1:
```

```
success success +1
```

```
i-i+1
```

```
try:
```

```
percentage success float(i)- 100
```

```
except:
```

```
data set.append(1)
```

```
if percentage < 17.0:
```

```
data set.append(1)
```

```
elif ((percentage >= 17.8) and (percentage < 81.0)) :
```

```
data_set.append(8) else:
```

```
data_set.append(-1)
```

#16, SFH

```
for form in soup.find_all('form', action= True):
```

```
if form["action"] or form['action'] = "about:blank":  
    data_set.append(-1)  
break  
  
elif url not in form['action'] and domain not in form['action']:  
    data_set.append(e)  
break  
else:  
    data_set.append(1)  
    break
```

#17. Submitting to email

```
if response == "":  
    data_set.append(-1)  
else:  
    if re.findall("[mail\\(\\) [mailto:?}", response.text):  
        data_set.append(1)  
else:  
    data_set.append(-1)
```

#18. Abnormal_URL

```
if response == "":  
    data_set.append(-1)
```

```
else:
if response.text == "":
    data_set.append(1)
else:
    data set.append(-1)
```

#19. Redirect

```
if response == "":
    data_set.append(-1)
else:
if len(response.history) <= 1:
    data_set.append(-1)
elif
len(response.history) <= 4;
    data set.append(e)
else:
    data set.append(1)
```

#20. on mouseover

```
if response = "" :
    data_set.append(-1)
```

```
else:
    if re.findall("<script>. data_set.append(1)+onmouseover.+</script>",
response.text):
else:
    data_set.append(-1)
```

#21. RightClick

```
if response == "":
    data_set.append(-1)
else:
    if re.findall(r"event.button?== ?2", response.text):
        data_set.append(1)
    else:
        data set.append(-1)
```

#22. popUplvidnow

```
if response == "":
    data_set.append(-1)
else:
    if re.findall(r"alert \(", response.text):
        data set.append(1)
    else:
        data set.append(-1)
```


#23. Iframe

```
if response:
    data set.append(-1)
else:
    if re.findall(r[<iframe><frameBorder>]", response.text):
        data set.append(1)
    else:
        data set.append(-1)

if response":
    data set.append(-1)
else:
    try:
        registration_date= re.findall(r'Registration Date: c/divdiv class="df-
value">([^]+)</es_response.text][[] if diff month(date.today(), date
parse(registration_date)) > 6:

        data set.append(-1)
    else:
        data set.append(1)
except:
    data set.append(1)
```

#25. DNSRecord

```
dns = 1
try:
    d = whois.whois (domain)
except:
    dns=-1
if dns == -1:
    data_set.append(-1)
else:
    if registration_length / 365 <= 1:
        data_set.append(-1)
    else:
        data_set.append(1)
```

#26. web traffic

```
try:
    rank = BeautifulSoup
    (urllib.request.urlopen("http://data.alexa.com/data/cli=108dat-s&url=" +
    url).read(), "nl").find("REACH")['RANK'] rank= int(rank)

    if (rankcleeeee):
        data set.append(1)
```

```
else:  
    data_set.append(e)  
except TypeError:  
    data set.append(-1)
```

#27. Page Rank

```
try:  
    if global_ranke and global_rank < 100000:  
        data_set.append(-1)  
else:  
    data set.append(1)  
except:  
    data set.append(1)
```

#28. Google Index

```
site search(url, 5)  
if site:  
    data_set.append(1)  
else:  
    data set.append(-1)
```

#29. Links pointing to page

```
if response == "":
    data_set.append(-1)
else:
    number_of_links = len(re.findall (r"<a href=", response.text))
if number_of_links == 0:
    data_set.append(1)
elif number_of_links <= 2:
    data_set.append(e)
else:
    data_set.append(-1)
```

#30. Statistical_report

try:

```
url_match=re.search('at\.ualusa\.cc/baltazarpresentes\.com\.br/pe\.hu/
esy\.es/hol\.es/sweddy\.com/myjino\.ru/96\. It ow\.ly, url)
```

ip_address-socket.gethostbyname (domain)

ip_match-

```
re.search(146\.112\.61\.108/213\.174\.157.151|1211.50\.168\.88/192\.1851.  
217\.116/78\.461.211\.158 181\.174\.165.13/46\.242\.1  
107\.151.148\.44|107\.151\.148\.107|641.701.19\.203|199\.184\.144\.27|10  
7\.151.148\.108 107\.151.1481.109/1191.281.  
118\.184\.251.861671.2081.74\.71|231.253\.1261.58/1041.2391.157\.210/17  
51.126.1231.219/141\.81.2241.221/101.10\.101.10 216\.218\.185\.162  
541.225\.104\.146/103\.243\.241.98/1991.59\.243\.120/31\.170\.160\.61/21  
31.191.1281.77162\.1131....
```

except:

```
print ('Connection problem. Please check your internet connection!)
```

```
print (data_set)
```

```
return data set
```