شناسایی داده های غیرنرمال جهت پیداکردن CRAWLER و INTRUSION ها از روی LOG سرور با استفاده از الگوریتم های یادگیری ماشین

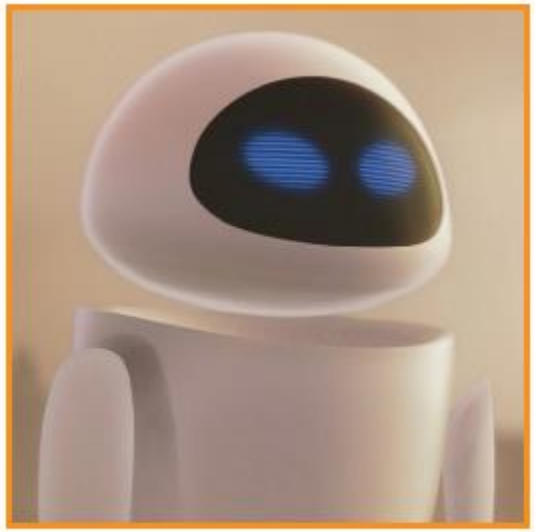
ارائه دهندگان : **حسین بیدکی ، پریسا مجرائی مقدم**

استاد راهنما: **جناب مهندس احمد اعتضادی**

تیر۱۴۰۱

OUTLINE

Data Exploration



Feature Engineering

Baseline Models

OUTLINE

Data Exploration

Feature Engineering

Baseline Models

DATA EXPLORATION

	ip	time	method	url	requested_file_type	status_code	response_length	user_agent	response_time	label
0	207.213.193.143	2021-5-12T5:6:0.0+0430	Get	/cdn/profiles/1026106239	NaN	304	0	Googlebot-Image/1.0	32	1
1	207.213.193.143	2021-5-12T5:6:0.0+0430	Get	images/badge.png	png	304	0	Googlebot-Image/1.0	4	1
2	35.110.222.153	2021-5-12T5:6:0.0+0430	Get	/pages/630180847	NaN	200	52567	Mozilla/5.0 (Linux; Android 6.0.1; SAMSUNG SM-...	32	0
3	35.108.208.99	2021-5-12T5:6:0.0+0430	Get	images/fav_icon2.ico	ico	200	23531	Mozilla/5.0 (Linux; Android 6.0; CAM-L21) Appl...	20	0

```
df.shape
```

```
(1241945, 10)
```

DATA EXPLORATION

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1241945 entries, 0 to 1241944
Data columns (total 10 columns):
 #   Column                Non-Null Count    Dtype
---  ------                --------------    -----
 0   ip                    1241945 non-null  object
 1   time                  1241945 non-null  object
 2   method                1241945 non-null  object
 3   url                   1241945 non-null  object
 4   requested_file_type   608061 non-null   object
 5   status_code           1241945 non-null  int64
 6   response_length       1241945 non-null  object
 7   user_agent            1241382 non-null  object
 8   response_time         1241945 non-null  object
 9   label                 1241945 non-null  int64
dtypes: int64(2), object(8)
memory usage: 94.8+ MB
```

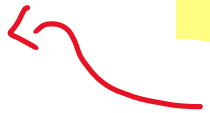
ویژگی	جنس	توضیحات
ip	string	به عنوان نمونه: 207.213.193.143
time	string	به عنوان نمونه: 2021-5-12T5:6:0.0+0430
method	string	Get/Put/Post/Options/Head
status_code	integer	وضعیت یک درخواست را نشان میدهد
url	string	به عنوان نمونه:images/fav_icon2.ico
response_length	integer	حجم اطلاعات موجود در آن درخواست.
user_agent	string	به عنوان نمونه : Googlebot-Image/۱.۰
response_time	float	زمان پاسخ سرور به کلاینت

جدول ۱ ویژگیهای دیتاست استفاده شده

CHECK MISSING VALUE

	requested_file_type	user_agent
0	NaN	Googlebot-Image/1.0
1	png	Googlebot-Image/1.0
2	NaN	Mozilla/5.0 (Linux; Android 6.0.1; SAMSUNG SM-...
3	ico	Mozilla/5.0 (Linux; Android 6.0; CAM-L21) Appl...
4	png	Mozilla/5.0 (Linux; Android 6.0.1; SAMSUNG SM-...
...	...	...
1241940	png	Mozilla/5.(Linux; Android 9; Redmi 7A) AppleWe...
1241941	woff	Mozilla/5.(Linux; Android 9; Redmi 7A) AppleWe...
1241942	NaN	okhttp/3.12.1
1241943	NaN	Googlebot-Image/1.0
1241944	NaN	Googlebot-Image/1.0

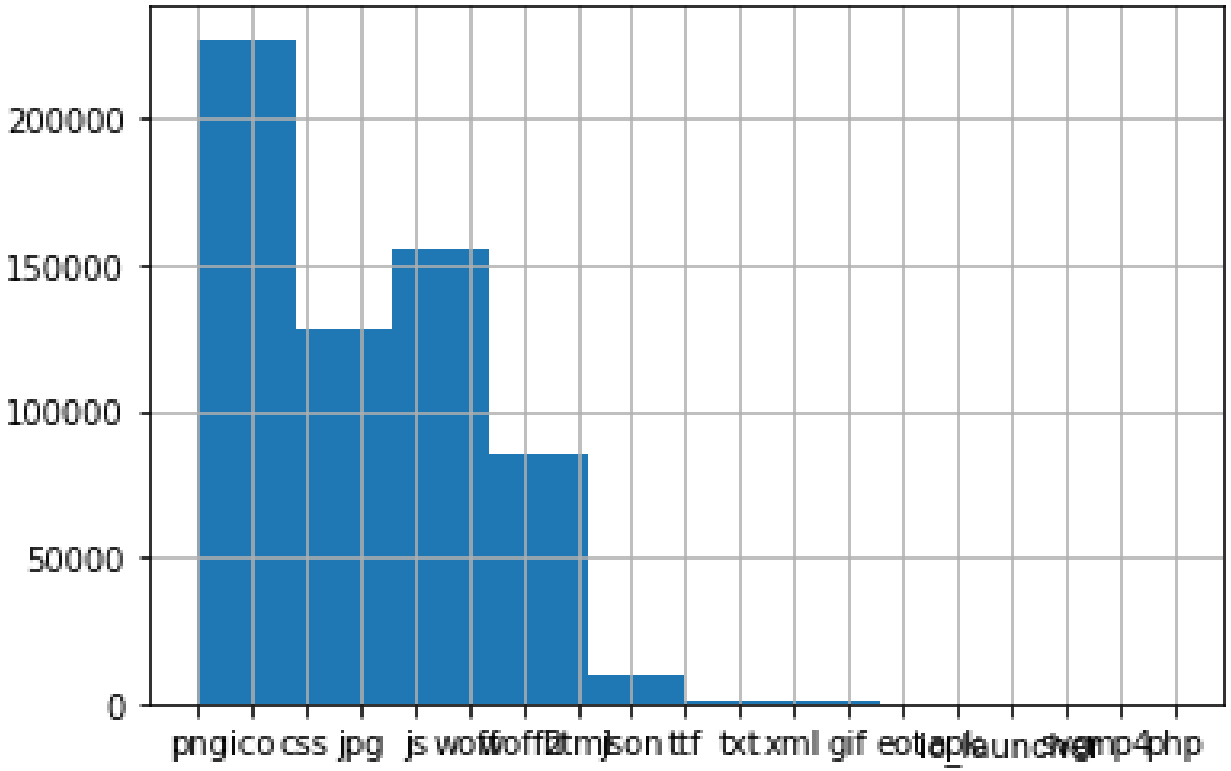
1241945 rows × 2 columns



```
df.isna().sum()
```

```
ip                        0
time                      0
method                    0
url                       0
requested_file_type    633884
status_code               0
response_length           0
user_agent              563
response_time             0
label                     0
dtype: int64
```

CHECK REQUESTED_FILE_TYPE FEATURE



از آنجایی که هدف ما شناسایی خزنده ها است، نوع فایل درخواستی اهمیت چندانی ندارد و از آنجایی که تعداد مقادیر از دست رفته بسیار زیاد است، حذف آنها منطقی نیست .بنابراین، مقادیر از دست رفته را برابر با بیشترین مقدار که PNG است پر می کنیم.

CHECK USER AGENT FEATURE

	ip	time	method	url	requested_file_type	status_code	response_length	user_agent	response_time
999	35.124.160.15	2021-5-12T5:6:47.0+0430	Get	images/appSlider/third.jpg	jpg	200	73[[Mozilla/5.0	NaN	(Linux; Android 8.0.0; SM-G570F) AppleWebKit/5...
14643	180.16.46.235	2021-5-12T5:17:21.0+0430	Get	images/appSlider/third.jpg	jpg	200	73[[Mozilla/5.0	NaN	(Linux; Android 6.0; ALE-L21) AppleWebKit/537....
15525	35.109.119.97	2021-5-12T5:18:8.0+0430	Get	images/appSlider/third.jpg	jpg	200	73[[Mozilla/5.0	NaN	(Linux; Android 11; SM-A505F) AppleWebKit/537....
26870	67.38.88.127	2021-5-12T5:28:26.0+0430	Get	/cdn/gadgets/754515939	png	200	127[[Mozilla/5.0	NaN	(Linux; Android 9; SAMSUNG SM-J701F) AppleWebK...
51074	14.62.137.148	2021-5-12T5:47:12.0+0430	Get	images/appSlider/third.jpg	jpg	200	73[[Mozilla/5.0	NaN	(Linux; Android 8.1.0; SM-J410F) AppleWebKit/5...
...	...	...	...	...	...	...	...	...	...
1227904	92.130.218.56	2021-5-12T15:2:46.0+0430	Get	images/appSlider/third.jpg	jpg	200	73[[Mozilla/5.0	NaN	(Linux; Android 7.0; SM-G920F) AppleWebKit/537...
1228372	4.0.51.74	2021-5-12T15:2:59.0+0430	Get	/cdn/profiles/392547969	png	200	2[[Mozilla/5.0	NaN	(Linux; Android 5.1; HUAWEI LUA-U22 Build/HUAW...
									(Linux; Android 10;

HANDLE MISSING VALUE

```
df.drop(df[df['user_agent'].isna()].index, inplace = True)
```

از آنجایی که تعداد این missing value ها در مقیاس با کل مقادیر کم است و همچنین از نظر منطقی، پر کردن آن می تواند برای تشخیص خزنده بودن بسیار مهم باشد، همانطور که در اسلاید قبل دیدیم ، فیچر های مشابه بسیار زیادی در فیلد های ویژگی های دیگر غیر خالی آنان دیده می شود آن ها را خزنده در نظر میگیریم و از دیتاست حذف می کنیم.

CHECK OTHER MISSING VALUE

```
df[df['response_time'] == '-'].size
```

```
17170
```

از آنجایی که response time، زمانی است که برای ارسال پاسخ سرور به مشتری نیاز است، تشخیص خزنده وب خیلی مهم نیست، بنابراین آن را با مقدار 0 پر می کنیم.

```
df[df["url"] == ""].size
```

```
0
```

برای سایر ویژگی ها نیز چک میکنیم ، مثلا فیلد url می تواند مقداری نداشته باشد که بررسی شد.

HANDLE MISSING VALUE

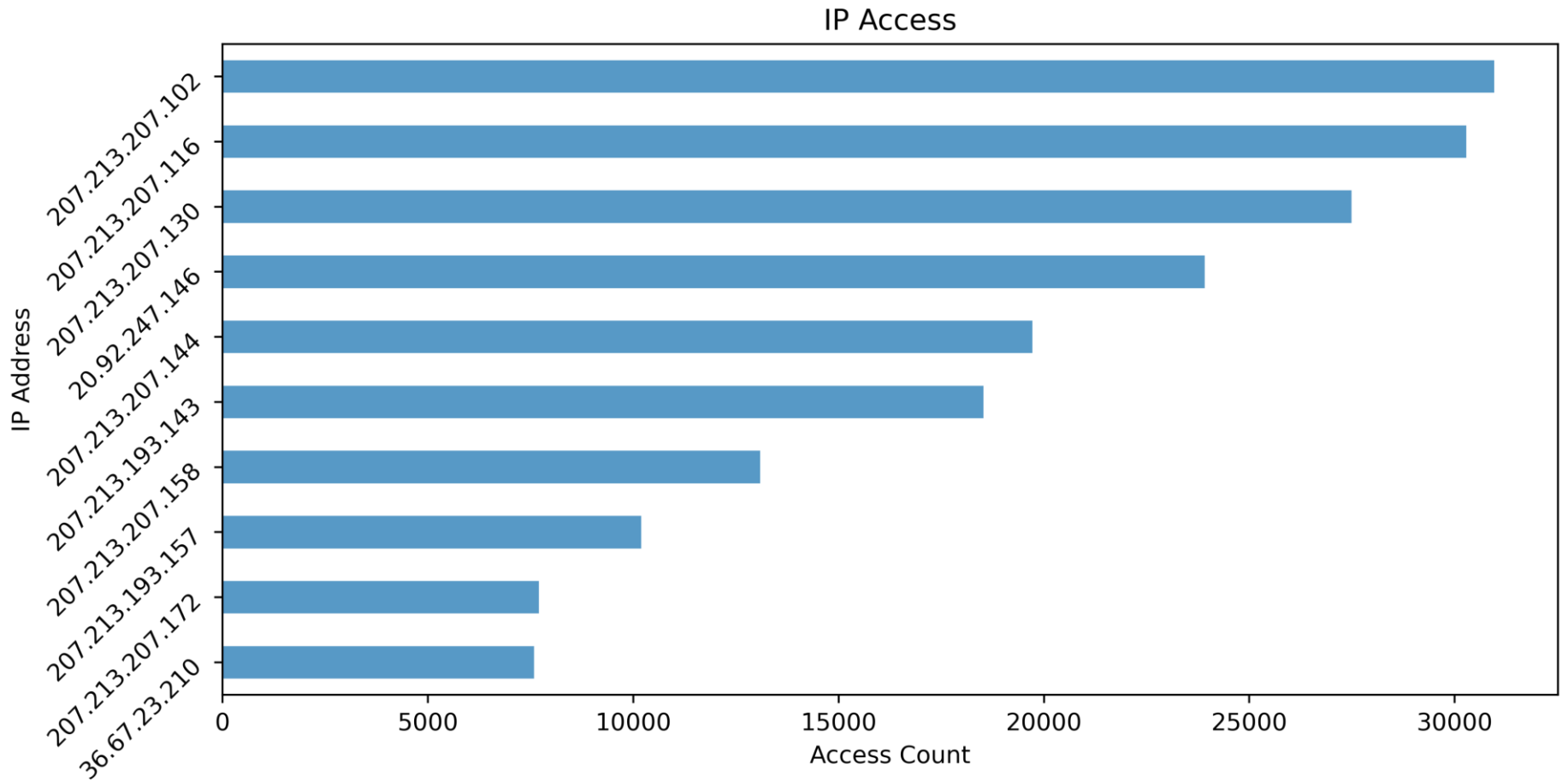
```
df.isna().sum()
```

```
ip                    0
time                  0
method                0
url                   0
requested_file_type   0
status_code           0
response_length       0
user_agent            0
response_time         0
label                 0
dtype: int64
```

در نهایت تمامی missing value ها مدیریت شد

- The Most Commons

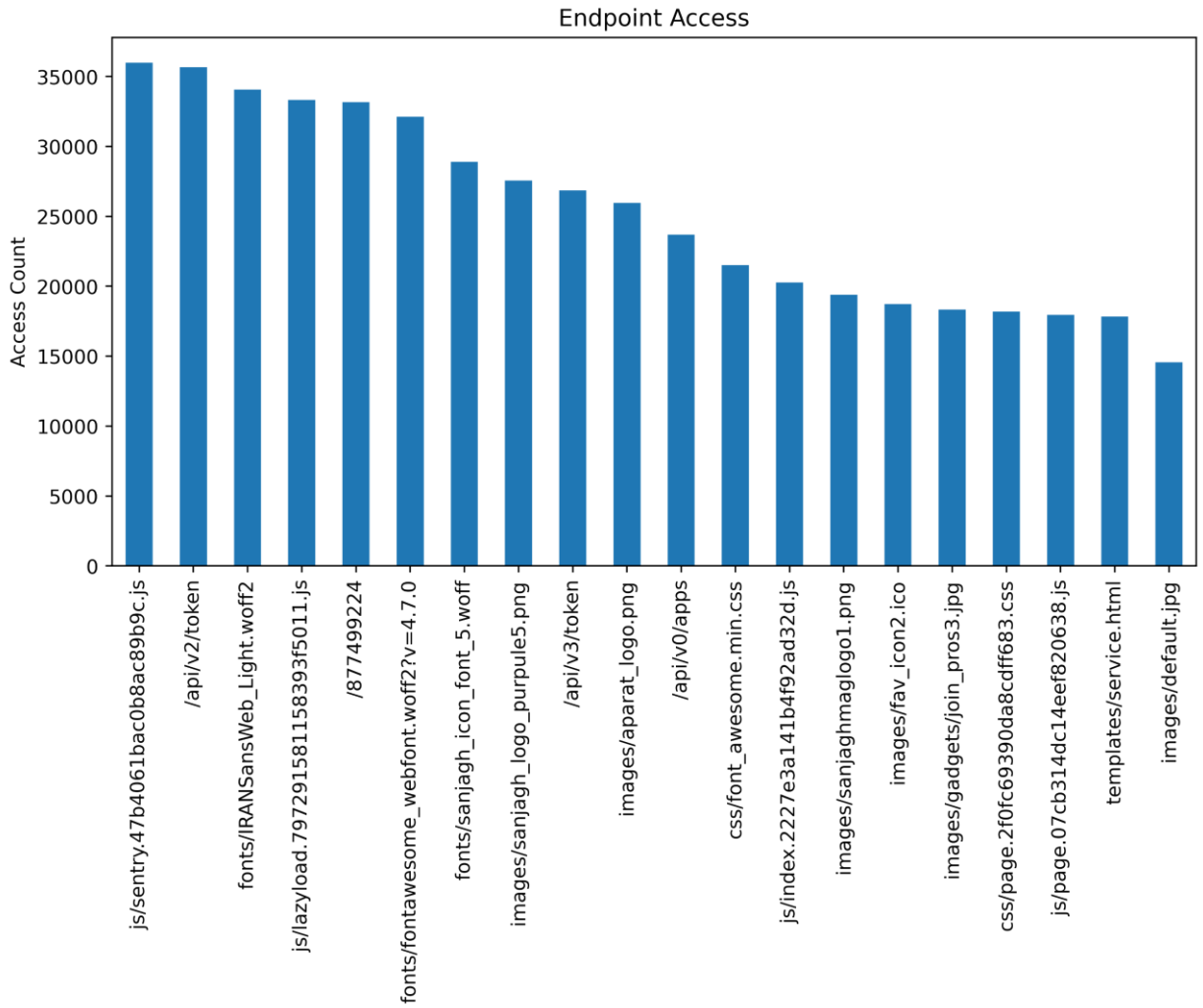
THE MOST VISITED IP ADDRESS



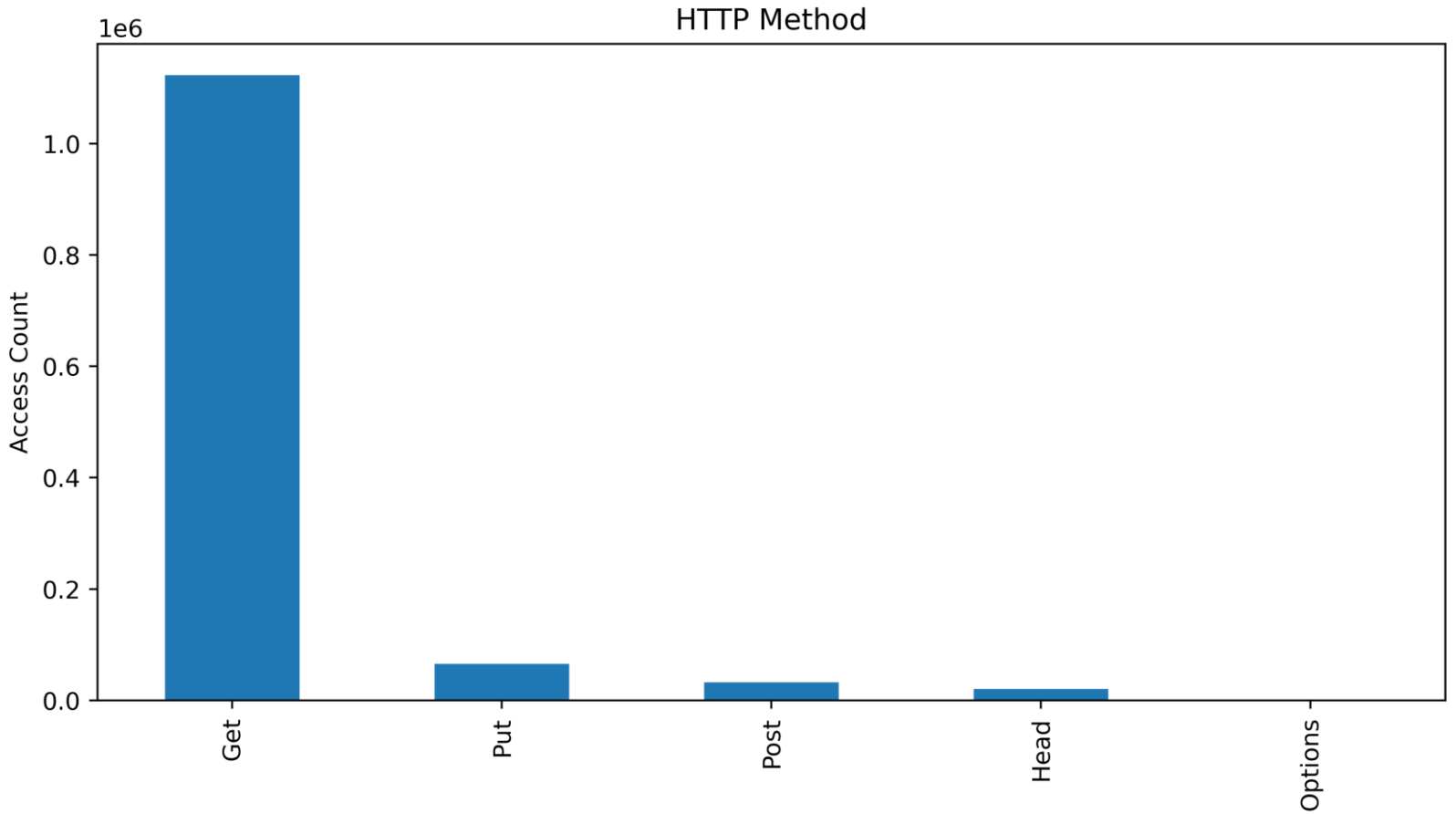
THE MOST VISITED IP ADDRESS

ip	time	method	url	requested_file_type	status_code	response_length	user_agent	response_time	lat
207.213.207.102	2021-5-12T6:25:56.0+0430	Get	/cdn/articles/1148001967	png	304	0	Googlebot-Image/1.0	16	
207.213.207.102	2021-5-12T7:53:23.0+0430	Get	/cdn/pro_photo_gallery/1647737278	png	304	0	Googlebot-Image/1.0	28	
207.213.207.102	2021-5-12T7:54:1.0+0430	Get	/cdn/articles/1258441802	png	304	0	Googlebot-Image/1.0	20	
207.213.207.102	2021-5-12T8:11:24.0+0430	Get	/cdn/pro_photo_gallery/2005389343	png	304	0	Googlebot-Image/1.0	16	
207.213.207.102	2021-5-12T9:23:9.0+0430	Get	/cdn/articles/1663054446	png	304	0	Googlebot-Image/1.0	20	
...	...	...	...	...	...	...	...	...	
207.213.207.102	2021-5-12T15:8:56.0+0430	Get	/best_pros/1331986151	png	200	60429	Mozilla/5.0 (Linux; Android 6.0.1; Nexus 5X Bu...	152	
207.213.207.102	2021-5-12T15:8:56.0+0430	Get	/amp/order/670781382/1990297374	png	200	103370	Mozilla/5.0 (Linux; Android 6.0.1; Nexus 5X Bu...	116	
207.213.207.102	2021-5-12T15:8:57.0+0430	Get	/cdn/profiles/1087372774	png	304	0	Googlebot-Image/1.0	12	
207.213.207.102	2021-5-12T15:8:58.0+0430	Get	/cdn/pro_photo_gallery/1818326365	png	304	0	Googlebot-Image/1.0	36	

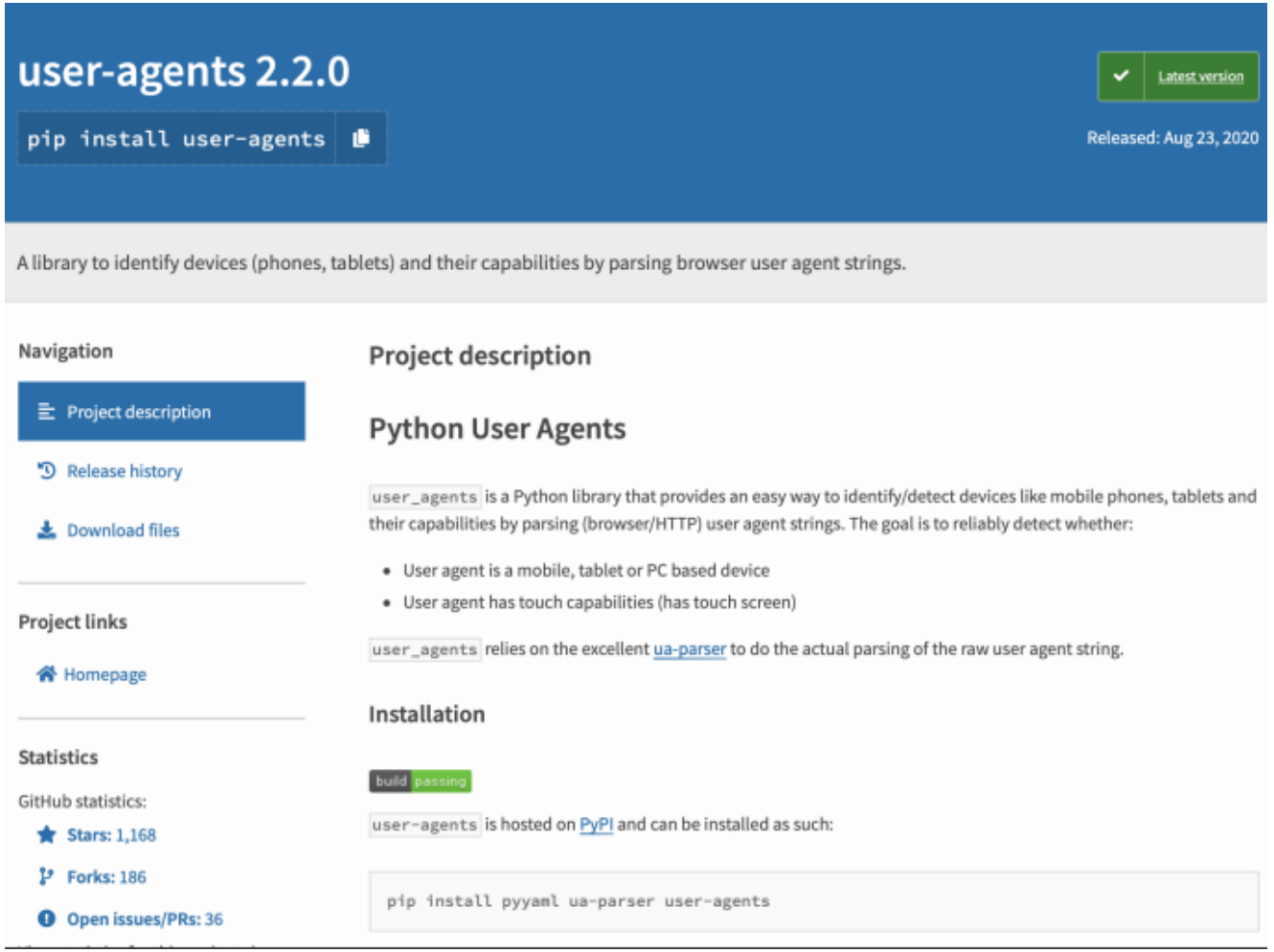
THE MOST REQUESTED ENDPOINTS



HTTP METHOD WITH THE MOST REQUESTS



THE MOST COMMON USER AGENTS



user-agents 2.2.0

pip install user-agents

Latest version

Released: Aug 23, 2020

A library to identify devices (phones, tablets) and their capabilities by parsing browser user agent strings.

Navigation

Project description

Release history

Download files

Project links

Homepage

Statistics

GitHub statistics:

Stars: 1,168

Forks: 186

Open issues/PRs: 36

Project description

Python User Agents

`user_agents` is a Python library that provides an easy way to identify/detect devices like mobile phones, tablets and their capabilities by parsing (browser/HTTP) user agent strings. The goal is to reliably detect whether:

- User agent is a mobile, tablet or PC based device
- User agent has touch capabilities (has touch screen)

`user_agents` relies on the excellent ua-parser to do the actual parsing of the raw user agent string.

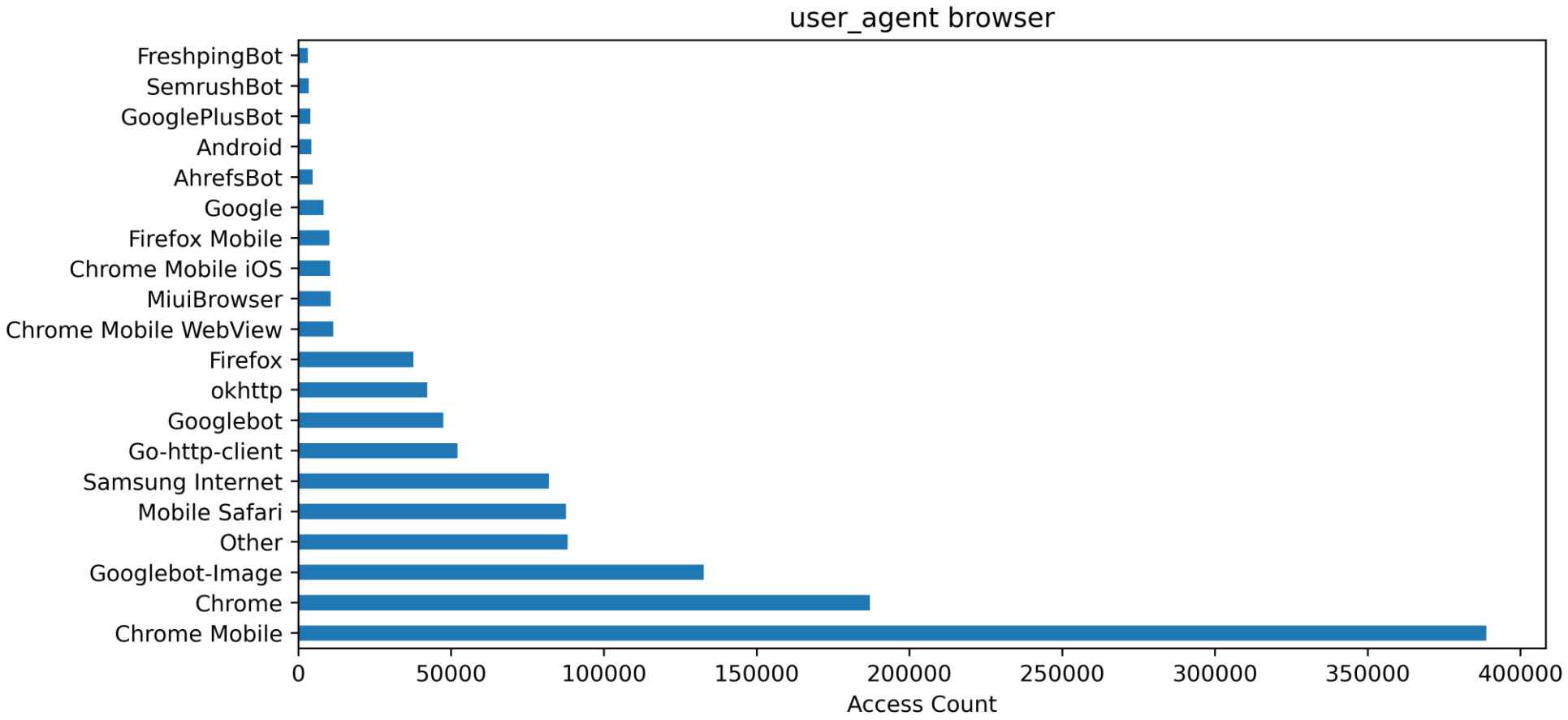
Installation

build passing

`user-agents` is hosted on PyPI and can be installed as such:

```
pip install pyyaml ua-parser user-agents
```

THE MOST COMMON USER AGENTS



Data Exploration

Feature Engineering

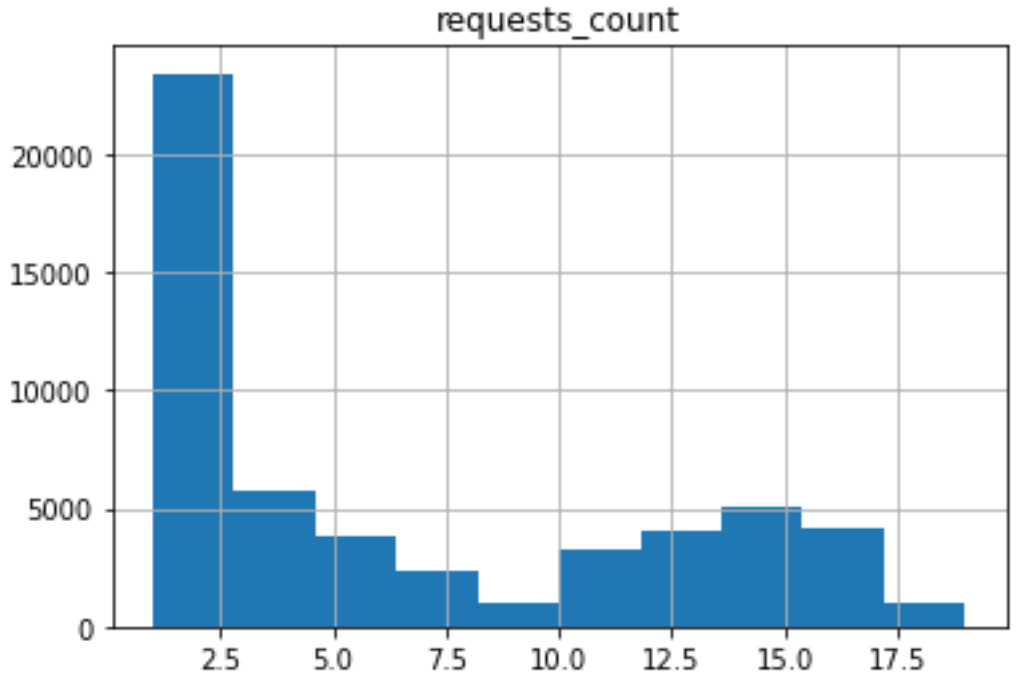
Baseline Models

- با توجه به دشـوار بودن تشـخیص anomaly بودن یا نبودن از روی تک درخواسـت، تصـمیم بر این گرفتیم تا دیتاسـت را براسـاس IP و User agent گروهبندی کنیم. با این کار گویی داریم یک نشست تعریف میکنیم. پس در این مرحله سـعی بر تعریف و مهندسـیسـازی ویژگی برای هر نشـسـت داریم. تعریف ویژگی ها از یک مقاله پژوهشـی الهام گرفته شـده اسـت

https://link.springer.com/chapter/10.1007/978-3-642-21916-0_52

- **Features per user**

NUMBER OF REQUESTS PER USER



زیادبودن تعداد درخواست در یک نشست موجب افزایش احتمال خزنده یا بات بودن آن کاربر میشود.

ip	user_agent	requests_count
20.92.247.146	sentry/21.4.1 (https://sentry.io)	23912
207.213.207.102	Googlebot-Image/1.0	23627
207.213.207.116	Googlebot-Image/1.0	23380
207.213.207.130	Googlebot-Image/1.0	21494
207.213.207.144	Googlebot-Image/1.0	15362
...	...	...
35.195.33.229	Mozilla/5.0 (compatible; heritrix/3.4.0-20200304 +https://zarebin.ir/)	1
35.195.33.187	Mozilla/5.0 (compatible; heritrix/3.4.0-20200304 +https://zarebin.ir/)	1
35.195.31.184	okhttp/3.12.1	1
35.195.31.168	Mozilla/5.0 (Linux; Android 11; SM-A515F Build/RP1A.200720.012; wv) AppleWebKit/537.36 (KHTML, like Gecko) Version/4.0 Chrome/89.0.4389.105 Mobile Safari/537.36 GSA/12.10.7.23.arm64	1
99.96.90.10	Mozilla/5.0 (iPhone; CPU iPhone OS 14_6 like Mac OS X) AppleWebKit/605.1.15 (KHTML, like Gecko) Version/14.1.1 Mobile/15E148 Safari/604.1	1

66372 rows × 1 columns

STD OF PATH LENGTH PER USER انحراف معیار عمق درخواستها

ip	user_agent	requests_count	url_length_std
102.24.134.34	Mozilla/5.0 (X11; Linux x86_64) app_version: 735	6	0.000000
102.86.13.63	Mozilla/5.0 (X11; Linux x86_64) app_version: 581	6	0.000000
102.86.6.9	Mozilla/5.(Linux; Android 10; SAMSUNG SM-A217F) AppleWebKit/537.36 (KHTML, like Gecko) SamsungBrowser/11.Chrome/75.0.3770.143 Mobile Safari/537.36	6	0.000000
102.93.72.182	Mozilla/5.(Linux; Android 8.1.0; DUB-LX1) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/85.0.4183.127 Mobile Safari/537.36	6	0.000000
11.127.239.60	Mozilla/5.0 (X11; Ubuntu; Linux i686; rv:24.0) Gecko/20100101 Firefox/24.0	6	0.000000
...	...	...	...
35.54.12.42	Mozilla/5.0 (Linux; Android 10; MAR-LX1M) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/83.0.4103.106 Mobile Safari/537.36	6	1.722401
35.124.166.122	Mozilla/5.0 (Linux; Android 8.1.0; SAMSUNG SM-J730F) AppleWebKit/537.36 (KHTML, like Gecko) SamsungBrowser/10.2 Chrome/71.0.3578.99 Mobile Safari/537.36	8	1.752549
60.92.118.166	Mozilla/5.0 (Linux; Android 5.1.1; SM-G531H Build/LMY48B) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/46.0.2490.76 Mobile Safari/537.36	11	1.793929
145.0.206.48	Mozilla/5.0 (Android 10; Mobile; rv:80.0) Gecko/80.0 Firefox/80.0	18	1.855041
14.226.145.71	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.101 Safari/537.36	13	2.594373

35011 rows × 2 columns

کاربرهای انسان، درخواست هایی که معمولا در یک نشست میزنند دارای عمق path با طول های متفاوتتر نسبت به خزندهها و باتها است.

PERCENTAGE OF 4XX RESPONSE CODES PER USER

ip	user_agent	requests_count	url_length_std	4xx_percentage(%)
35.124.193.182	Dalvik/2.1.0 (Linux; U; Android 10; SM-A115F Build/QP1A.190711.020)	104	0.000000	100.0
35.132.17.132	Mozilla/5.(Linux; Android 6.0.1; SAMSUNG SM-J700F) AppleWebKit/537.36 (KHTML, like Gecko) SamsungBrowser/14.Chrome/87.0.4280.141 Mobile Safari/537.36	45	0.000000	100.0
153.126.209.239	Go-http-client/1.1	35	0.000000	100.0
35.202.101.118	Mozilla/5.(Linux; Android 5.1.1; SAMSUNG SM-J111F) AppleWebKit/537.36 (KHTML, like Gecko) SamsungBrowser/13.2 Chrome/83.0.4103.106 Mobile Safari/537.36	22	0.000000	100.0
35.244.120.44	Go-http-client/1.1	18	0.000000	100.0
...	...	...	...	...
92.239.237.42	Mozilla/5.0 (X11; Linux x86_64) app_version: 581	6	0.836660	0.0
92.239.251.224	Mozilla/5.(Linux; Android 10; Redmi Note 9 Pro) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.101 Mobile Safari/537.36	6	0.000000	0.0
92.51.185.100	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.77 Safari/537.36	6	0.000000	0.0
99.239.174.175	Mozilla/5.0 (iPhone; CPU iPhone OS 14_5 like Mac OS X) AppleWebKit/605.1.15 (KHTML, like Gecko) Version/14.1 Mobile/15E148 Safari/604.1	6	0.516398	0.0
99.239.247.155	Mozilla/5.(Linux; Android 10; SM-A307FN) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.88 Mobile Safari/537.36	6	0.000000	0.0

35011 rows × 3 columns

معمولا خزندهها به پاسخهایی از سمت سرور با خطایی از خانواده ۴۰۰ بیشتر مواجه میشوند.

PERCENTAGE OF 3XX RESPONSE CODES PER USER

ip	user_agent	requests_count	url_length_std	4xx_percentage(%)	3xx_percentage(%)
35.26.221.84	Mozilla/5.(Windows NT 6.1; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.101 Safari/537.36	4043	0.453993	0.0	100.0
35.125.194.34	Mozilla/5.(Linux; Android 10; Redmi Note 9 Pro Max Build/QKQ1.191215.002) AppleWebKit/537.36 (KHTML, like Gecko) Soul/4.Chrome/91.0.4472.88 Mobile Safari/537.36	530	0.455319	0.0	100.0
207.213.193.143	Mozilla/5.(Linux; Android 6.0.1; Nexus 5X Build/MMB29P) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.9Mobile Safari/537.36 (compatible; Googlebot/2.1; +http://www.google.com/bot.html)	524	0.329976	0.0	100.0
35.117.109.145	Mozilla/5.(Linux; Android 11; SM-A505F) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.101 Mobile Safari/537.36	424	0.292646	0.0	100.0
86.151.172.46	Mozilla/5.(Linux; U; Android 7.0; en-US; TRT-L21A Build/HUAWEITRT-L21A) AppleWebKit/537.36 (KHTML, like Gecko) Version/4.Chrome/78.0.3904.108 UCBrowser/13.3.8.1305 Mobile Safari/537.36	399	0.490717	0.0	100.0
...	...	...	...	...	...
92.239.17.78	Mozilla/5.0 (X11; Linux x86_64) app_version: 581	6	0.836660	0.0	0.0
92.239.216.149	Mozilla/5.0 (Linux; Android 9; SM-J701F) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/77.0.3865.92 Mobile Safari/537.36	6	0.516398	0.0	0.0
92.239.237.42	Mozilla/5.0 (X11; Linux x86_64) app_version: 581	6	0.836660	0.0	0.0
92.51.185.100	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.77 Safari/537.36	6	0.000000	0.0	0.0

خانواده ۳۰۰ به معنای redirect شدن به یک صفحه دیگر است. خزندهها و باتها معمولا بیشتر با این پاسخ روبرو میشوند.

PERCENTAGE OF HTTP HEAD REQUESTS, PER USER

ip	user_agent	requests_count	url_length_std	4xx_percentage(%)	3xx_percentage(%)	HEAD_count(%)
20.163.161.41	Mozilla/5.(iPhone; CPU iPhone OS 7_like Mac OS X; en-us) AppleWebKit/537.51.1 (KHTML, like Gecko) Version/7.Mobile/11A465 Safari/9537.53	27	0.891556	0.000000	0.000000	100.000000
36.67.23.210	Go-http-client/2.0	7582	0.000000	1.609074	9.773147	45.805856
60.148.0.167	Go-http-client/2.0	7351	0.000000	1.714053	10.053054	43.980411
20.92.247.170	Go-http-client/2.0	7273	0.000000	0.000000	10.325863	42.334662
76.212.164.3	Go-http-client/2.0	6549	0.000000	1.878149	9.406016	42.113300
...	...	...	...	...	...	...
92.239.237.42	Mozilla/5.0 (X11; Linux x86_64) app_version: 581	6	0.836660	0.000000	0.000000	0.000000
92.239.251.224	Mozilla/5.(Linux; Android 10; Redmi Note 9 Pro) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.101 Mobile Safari/537.36	6	0.000000	0.000000	100.000000	0.000000
92.51.185.100	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.77 Safari/537.36	6	0.000000	0.000000	0.000000	0.000000

در خزندهها یا بات ها بیشتر است زیرا احتمال مواجهه با صفحات پاکشده
یا تاریخ گذشته در این نوع از کاربران زیادتر است.

PERCENTAGE OF IMAGE REQUESTS PER USER

ip	user_agent	requests_count	url_length_std	4xx_percentage(%)	3xx_percentage(%)	HEAD_count(%)	image_count(%)
102.15.174.47	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.Safari/537.36	21	0.000000	0.0	0.0	0.0	0.0
102.29.29.19	Mozilla/5.0 (X11; Linux x86_64) app_version: 581	21	0.000000	0.0	0.0	0.0	0.0
113.11.38.30	Mozilla/5.0 (X11; Linux x86_64) app_version: 581	21	0.749603	0.0	0.0	0.0	0.0
113.111.195.219	Mozilla/5.0 (X11; Linux x86_64) app_version: 581	21	0.000000	0.0	0.0	0.0	0.0
113.118.175.102	Mozilla/5.0 (X11; Linux x86_64) app_version: 581	21	0.810643	0.0	0.0	0.0	0.0
113.118.45.106	okhttp/3.12.1	21	0.436436	0.0	0.0	0.0	0.0
113.118.90.28	Mozilla/5.0 (X11; Linux x86_64) app_version: 581	21	0.810643	0.0	0.0	0.0	0.0
113.118.96.94	okhttp/3.12.1	21	0.358569	0.0	0.0	0.0	0.0

خزندهها و باتها معمولا به عکس ها درخواست نمیدهند.

AVERAGE AND SUM OF THE RESPONSE LENGTH AND RESPONSE TIME PER USER

]:

count(%)	total_response_length	mean_response_length	total_response_time	mean_response_time
0.000000	29827298272982729827298272982729827	4.971216e+28	008488	1.414667e+03
5.000000	298272982729827235312982729827298272353 1	3.728412e+38	00044448	5.556000e+03
2.857143	284001038864477160332023235313989 0	4.057158e+32	00481644	6.880629e+04
6.666667	000000	0.000000e+00	444004	7.400067e+04

بدلیل اینکه کاربرهای انسان از مرورگر برای دسترسی به صفحات وب استفاده میکنند، وقتی به یک صفحه درخواست میزنند، نشست مجبور به دریافت منابع و عکسهای متفاوتی است و همین باعث زیاد شدن زمان پاسخ و حجم درخواست میشود.

SET THE BROWSER FOR EACH USER AGENT

مرورگر–سیستمعامل–بات بودن یا نبودن–از یک pc بودن یا نبودن.

mean_response_time	avg_url_count_norm	browser	os	is_bot	is_pc
2.800000e+01	0.000161	Mobile Safari	iOS	False	False
2.800000e+01	0.002900	Mobile Safari	iOS	False	False
6.366633e+12	0.002440	Mobile Safari	iOS	False	False
5.731473e+44	0.726120	Samsung Internet	Android	False	False

NUMBER OF "ROBOTS.TXT" REQUESTS

ip	user_agent	requests_count	robots_txt_reqs
67.149.194.62	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/92.0.4515.51 Safari/537.36	47	9.0
79.130.18.121	Mozilla/5.0 (compatible; bingbot/2.0; +http://www.bing.com/bingbot.htm)	237	8.0
207.213.207.116	Mozilla/5.0 (compatible; Googlebot/2.1; +http://www.google.com/bot.html)	45	4.0
207.213.207.3	Mozilla/5.0 (compatible; Googlebot/2.1; +http://www.google.com/bot.html)	4	4.0
118.151.92.167	Mozilla/5.0 (compatible; MJ12bot/v1.4.8; http://mj12bot.com/)	8	4.0
...	...	...	...
			0.0

خزنده ها برای اینکه متوجه شوند سرور سایت امکان خزش به چه صفحاتی را داده است، به robots.txtدرخواست میزنند. اینگونه اطلاعات در آنجا نوشته شده است.

PERCENTAGE OF CONSECUTIVE REPEATED HTTP REQUESTS PER SESSION

ip	user_agent	requests_count
20.92.247.146	sentry/21.4.1 (https://sentry.io)	23912
36.67.23.210	Go-http-client/2.0	7582
60.148.0.167	Go-http-client/2.0	7351
20.92.247.170	Go-http-client/2.0	7273
76.212.164.3	Go-http-client/2.0	6549
20.116.215.189	Go-http-client/2.0	6401
20.117.146.75	Go-http-client/2.0	6067

browser	os	is_bot	is_pc	robots_txt_reqs	consq_rep_url_count
Other	Other	False	False	0.0	99.983272
Go-http-client	Other	False	False	0.0	99.973622
Go-http-client	Other	False	False	0.0	99.972793
Go-http-client	Other	False	False	0.0	99.972501
Go-http-client	Other	False	False	0.0	99.969461
Go-http-client	Other	False	False	0.0	99.968755
Go-http-client	Other	False	False	0.0	99.967035

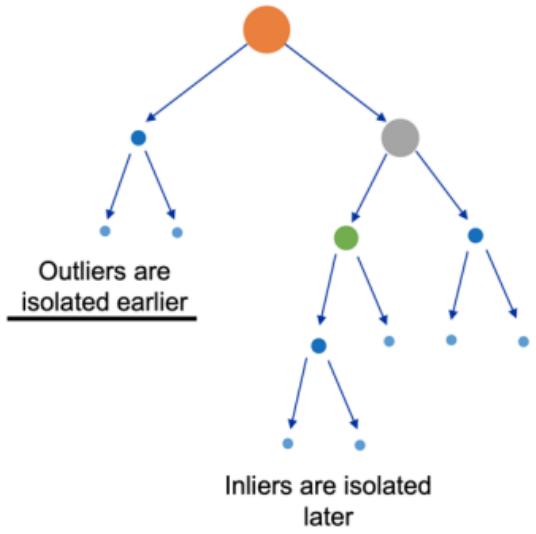
یک ویژگی عددی محاسبه شده به عنوان تعداد درخواست های مکرر ارسال شده به ترتیب متعلق به همان فهرست وب ارسال شده توسط کاربر در طول یک جلسه

Modeling

مدل های غیرنظارتی

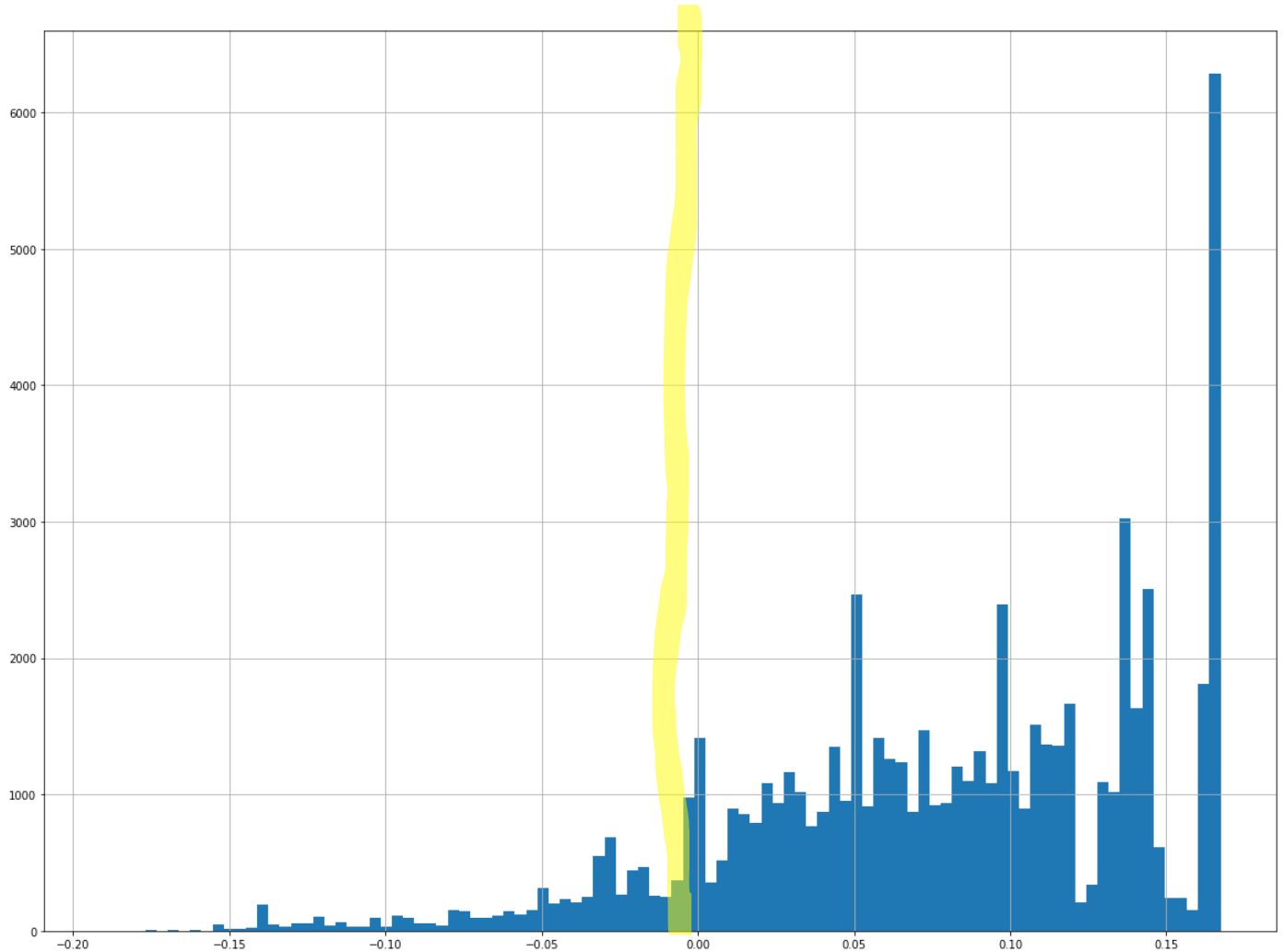
- به دلیل نداشتن برچسب برای دادگان، باید از مدل های غیرنظارتی بهره ببردیم. مدل های غیرنظارتی زیادی در علم یادگیری ماشین برای تشخیص داده پرت تا به الان شناخته شده اند. از معروفترین آنها میتوان به موارد زیر اشاره نمود:
  - Isolation Forest
  - Local Outlier Factor
  - One-class SVM
  - Robust covariance

ISOLATIONFOREST

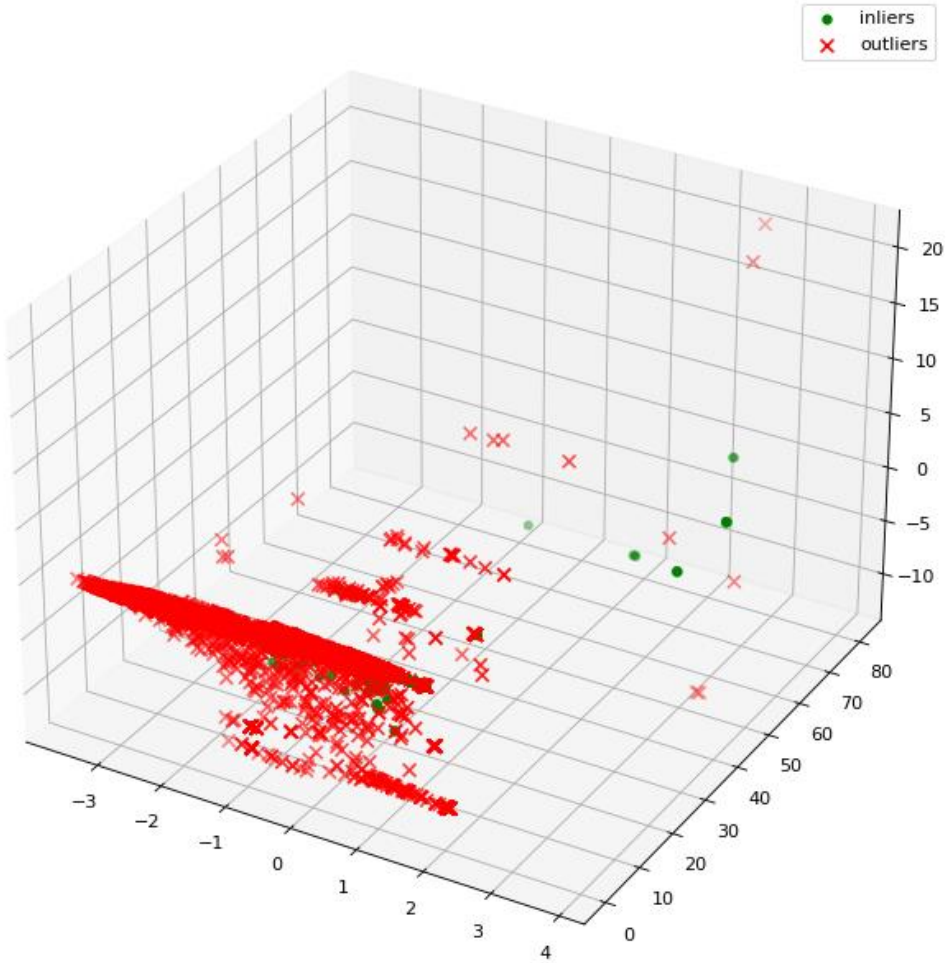


- این روش، تکنیکی برای تشـــخیص دادههای پرت بـه روش غیرنظارتی اســـت.
- در این روش مشاهدهها، با انتخاب ویژگیها به صـورت تصـادفی و جداسـازی مقدار أن به بیشـترین و کمترین مقادیر ویژگی انتخابی، ایزوله میشـوند.
- به دلیل خاصـیت بازگشـتی بودن این روش، این روش با یک سـاختار درختی قابل نمایش اســـت.
- بدلیل اینکه مقادیر ویژگی های دادههای پرت به طرز قابل توجهی با بقیه دادگان تفاوت دارد، دادههای پرت زودتر در درخت تصـمیم ایزوله میشـوند.

تفکیک داده‌های پرت و غیرپرت با آستانه گذاری بصورت تجربی



مصورسازی دادگان پرت و نرمال به کمک PCA



```
print(X['anomaly'].values_counts())
```

```
 1    27756
-1     3785
Name: anomaly, dtype: int64
```

1. Isolation Forest – Results

Crawler

	ip	time	method	status_code	path	response_length	user_agent	response_time	path_count_norr
296503	233.46.142.110	2021-05-12 09:32:04+04:30	Put	200	api/v2/token	289	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_13_6...	8.0	2.
297902	233.46.142.110	2021-05-12 09:32:50+04:30	Put	200	api/v2/token	289	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_13_6...	8.0	2.
302321	233.46.142.110	2021-05-12 09:35:31+04:30	Get	101	api/v2/connect/1396318207	99	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_13_6...	160053.0	0.
302369	233.46.142.110	2021-05-12 09:35:33+04:30	Put	200	api/v2/token	289	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_13_6...	12.0	2.
303761	233.46.142.110	2021-05-12 09:36:14+04:30	Get	101	api/v2/connect/1944714213	465235	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_13_6...	40604.0	0.
304029	233.46.142.110	2021-05-12 09:36:18+04:30	Put	200	api/v2/token	289	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_13_6...	8.0	2.
328160	233.46.142.110	2021-05-12 09:48:56+04:30	Put	200	api/v2/token	289	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_13_6...	12.0	2.

1. Isolation Forest – Results

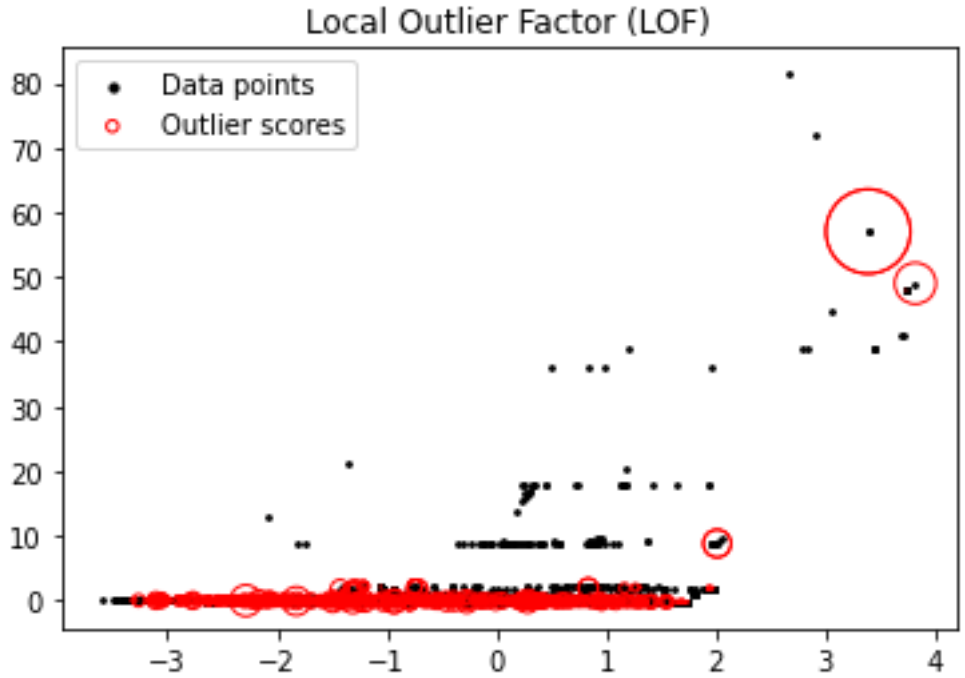
Normal

	ip	time	method	status_code	path	response_length	user_agent	response_time	path_count_norm
631560	4.138.32.12	2021-05-12 11:58:09+04:30	Get	200	pages/630180842	50797	Mozilla/5.0 (Linux; Android 10; Redmi Note 8) ...	16.0	0.0
631958	4.138.32.12	2021-05-12 11:58:19+04:30	Get	200	css/font_awesome.min.css	30891	Mozilla/5.0 (Linux; Android 10; Redmi Note 8) ...	8.0	1.7
631959	4.138.32.12	2021-05-12 11:58:19+04:30	Get	200	css/page.2f0fc69390da8cdff683.css	50880	Mozilla/5.0 (Linux; Android 10; Redmi Note 8) ...	8.0	1.4
634810	4.138.32.12	2021-05-12 11:59:31+04:30	Get	200	js/page.07cb314dc14eef820638.js	332023	Mozilla/5.0 (Linux; Android 10; Redmi Note 8) ...	28.0	1.4
634808	4.138.32.12	2021-05-12 11:59:31+04:30	Get	200	images/gadgets/join_pros3.jpg	34053	Mozilla/5.0 (Linux; Android 10; Redmi Note 8) ...	8.0	1.4
634807	4.138.32.12	2021-05-12 11:59:31+04:30	Get	200	images/sanjagh_logo_purpule5.png	4680	Mozilla/5.0 (Linux; Android 10; Redmi Note 8) ...	4.0	2.2
634809	4.138.32.12	2021-05-12 11:59:31+04:30	Get	200	images/default.jpg	20993	Mozilla/5.0 (Linux; Android 10; Redmi Note 8) ...	8.0	1.1

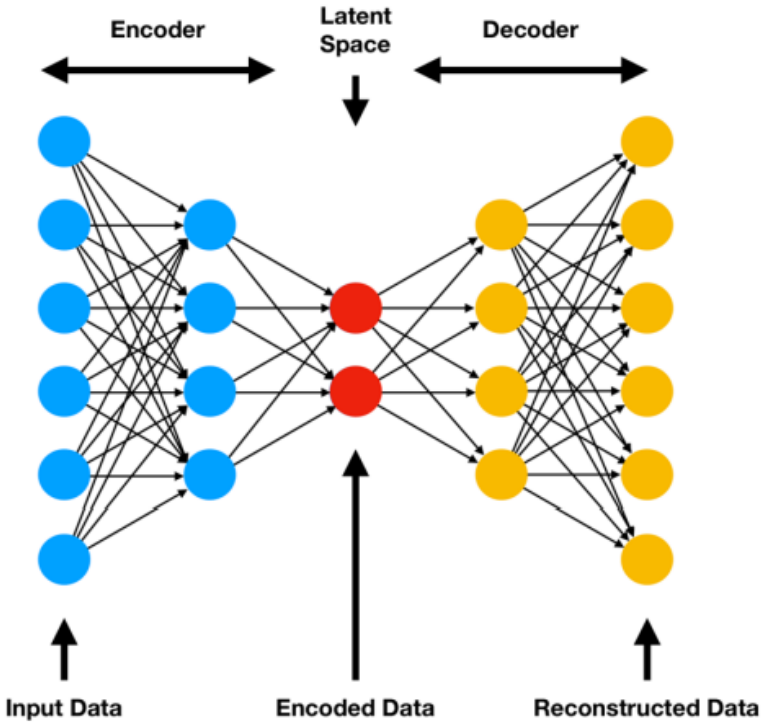
LOCALOUTLIERFACTOR

- در روش LOF، انحراف محلی چگالی هر نمونه نسبت به همسایگان خود محاسبه میشود. پرت بودن یک داده به میزان ایزوله و تنها بودن آن نمونه نسبت به همسایگان خود سنجیده میشود.
- همچنین چگالی محلی به کمک متریک فاصلهای که در روش KNN محاسبه میشود صـ ورت میگیرد.

اما این روش نتایج خوبی برای دیتاستی که ما داشتیم به همراه نداشت زیرا حتی محتمل ترین نمونه هایی که به عنوان داده پرت درنظر گرفته بود گاها داده نرمال بودند. به همین دلیل از این روش در ادامه استفاده ای نشد.



AUTO-ENCODER



اتوانکودرها یک نوع خاص ی از شبکه های عصبی هستند که مقادیر نورونهای ورودی را در خروجی کپی میکنند.

در فرآیند آموزش این نوع شبکه، دیگر نیازی به برچسب دادهها وجود ندارد برای همین میتوان اتوانکودر ها را جزو الگوریتمهای غیر نظارتی دانست.

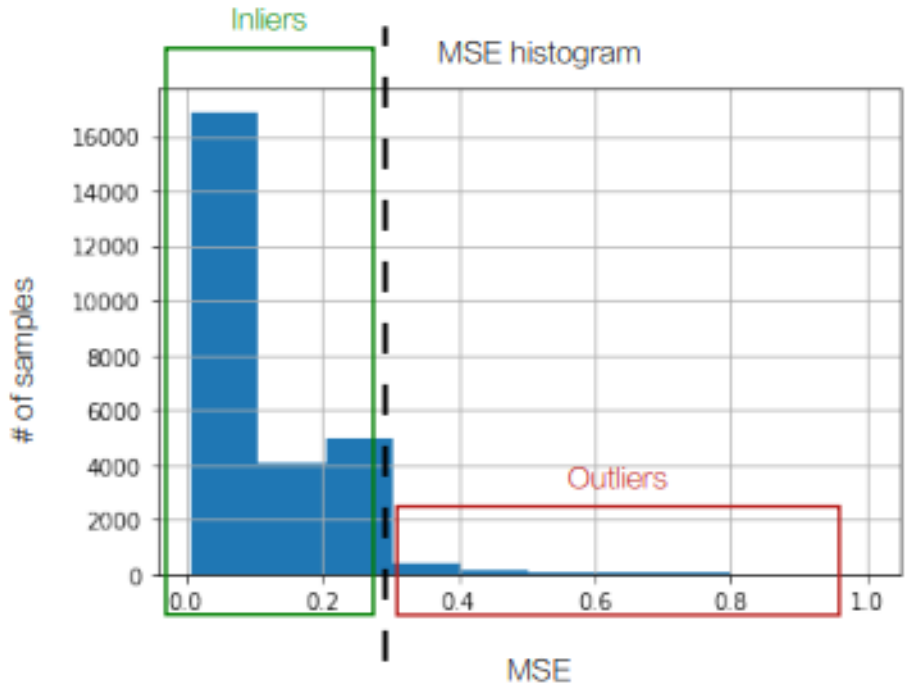
به طور معمول لایه های مخفی در این نوع شبکه ها، دارای تعداد نورون کمتری نسبت به نورونهای ورودی و خروجی دارند. به همین دلیل، لایه های مخفی اطلاعات ضروری را در خود ذخیره میکنند و از نویز ها صرف نظر میکنند.

در فرآیند آموزش این نوع شبکه دو قسمت مهم encode و decode وجود دارد. در مرحلهای encoding، مقادیر ورودی را فشرده سازی میکند و به فضای برداری لایه مخفی میبرد. در مرحله decoding، اطلاعات فشردهشده، بازسازی میشوند.

- اتوانکودرها کاربردهای وسیعی در زمینه پردازش تصویر و بینایی ماشین دارند همچنین نتایج درخشانی در زمینه تشخیص ناهنجاری نیز داشته اند.
- در فرآیند decoding، هنگامی که عمل بازسازی صورت میگیرد میتوان خطای یکسان نبودن داده خروجی با داده ورودی اولیه را حساب کرد. به عبارت دیگر، لایه مخفی سعی بر یادگیری یک embedding از داده های ورودی است و میخواهد داده های ورودی را تنها با داشتن ویژگی های موجود در لایه مخفی بازسازی کند. به طبع اگر داده پرتی که اختلاف زیادی از نظر مقادیر ویژگیها با دادههای دیگر وجود داشته باشد، خطایبازسازی بسیار زیاد و متفاوت است.
- پس با این رویکرد میتوان از اتوانکودر ها برای تشخیص ناهنجاری نیز استفاده نمود.
- در فرآیند آموزش از بهینه ساز Adam و تابع خطای MSE برای محاسبه خطا استفاده کردیم. همچنین از ReLu به عنوان تابع غیرخطی ساز استفاده کردهایم. دیتاست با توزیع ۸۰-۲۰ به دو داده آموزش و تست تقسیم شد و در جدول مقادیر خطا پس از آموزش برای شبکه های با معماری هایی متفاوت قابل مشاهده است.

تعداد نورونها	خطای داده آموزش	خطای داده تست
[15, 7, 15]	۰.۴۲	۰.۴۸
[15, 3, 15]	۰.۲۸	۰.۳۹
[15, 7, 3, 7, 15]	۰.۲۹	۰.۴۳
[15, 7, 7, 7, 15]	۰.۳۱	۰.۴۲

- پس از آموزش مدل، برای فاز پیشبینی ناهنجاری‌بودن یا نبودن، مشـــابه الگوریتمهای قبلی، باید یک حد آســتانه برای خطای MSE نیز تعریف کرد.
- با توجه به دانش قبلی و بررســی بات بودن یا نبودن نشـست های داخل دیتافریم براساس user agent شـان، حدود ۵ درصـد از دیتاسـت به واضـح بات بودند، بنابراین آسـتانه خطای MSE به طوری انتخاب شـد که ۵ درصـد از نمونه ها به عنوان ناهنجاری تشخیص داده شوند.



One fact:

Approximately, **5%** of the dataset contains common bots, crawlers.

MSE threshold = 0.30

- پایان