REVA UNIVERSITY
Bengaluru, India

Established as per the section 2(f) of the UGC Act, 1956,
Approved by AICTE, New Delhi

REVA Academy for Corporate Excellence (RACE)

Building a Multi-class prediction App for Malicious URLs

Paper ID : 371
ANTIC-2022

Presenter : Vijayaraj S

MTech in Cybersecurity
Vijayraj.cs04@race.reva.edu.in

2nd & 3rd Authors:

Dr. Shinu Abhi & Dr. Rashmi Agrawal

Date: 24-Dec-2022

race.reva.edu.in

Introduction

Malicious landing URLs are those that give users the chance to click on harmful links.

Simple click may misuse user's computing resources, steal confidential data, or carry out other forms of assaults.

Employees receive an average of 14 phishing emails per year.

Users clicking on bad URL reported as second most expensive cause of data breaches.

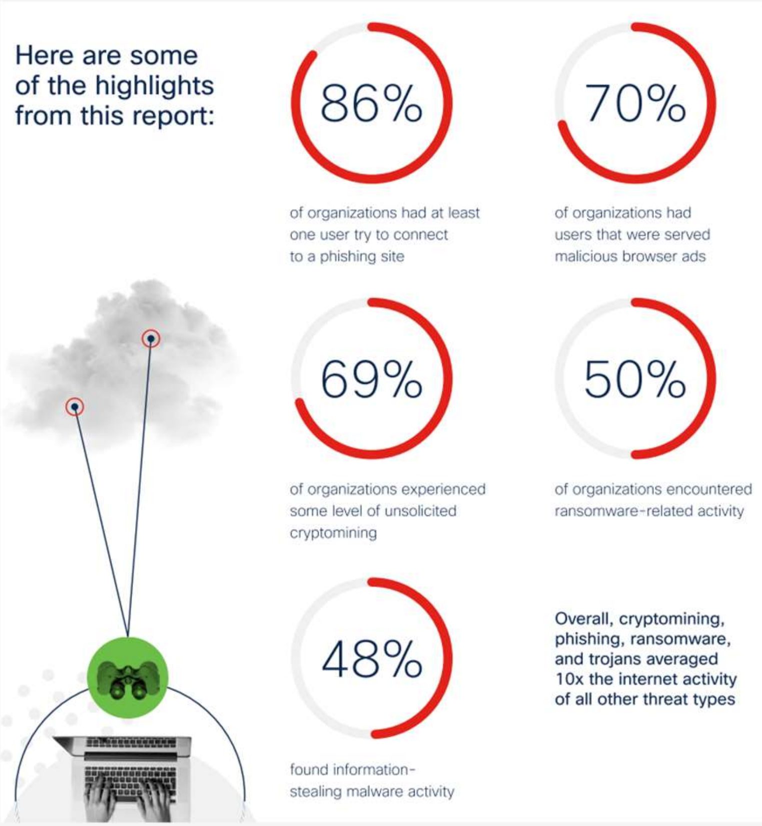
It is imperative to detect and block these threats effectively.

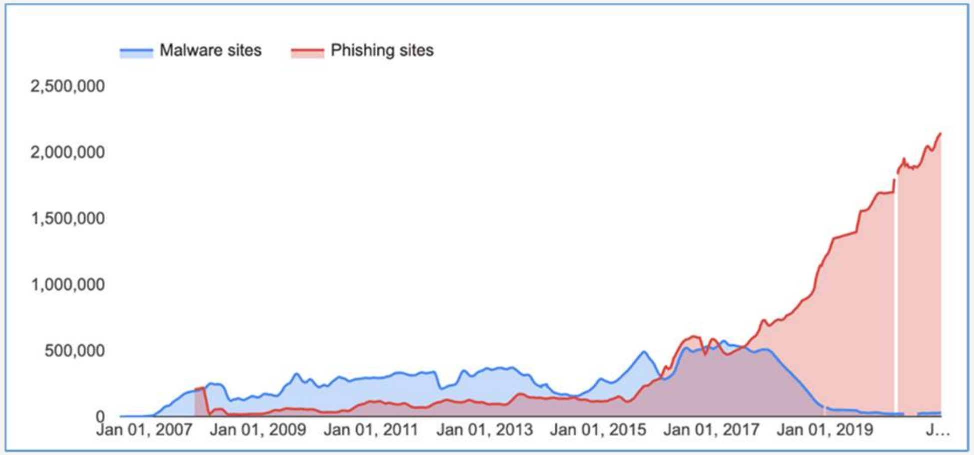
Google Safe Browsing data depicts sharp rise in Phishing sites between Jan 2016 – Jan 2021

Introduction

Background | Current status | Why this topic

2021 Malware Phishing Trojans Increase Trends





[1] Courtesy: Cisco : 2021 Cyber Security Threat trends

Problem Statement

Technical | Functional

- **Methods widely used** : Sandboxing, secure email gateways, and installing plugins.

- **ML** Detection Techniques (**Binary class**) in **literatures** : URL based, Host based, content based.

- **Needs supplementation**
  - Host combinations are less explored.
  - Parametric and nonparametric algorithms less explored for **multi classification** or URLs.

- Need of simple powerful solution bringing **detection** and **protection** for end users.

REVA UNIVERSITY
Bengaluru, India
Established as per the section 2(f) of the UGC Act, 1956, Approved by AICTE, New Delhi

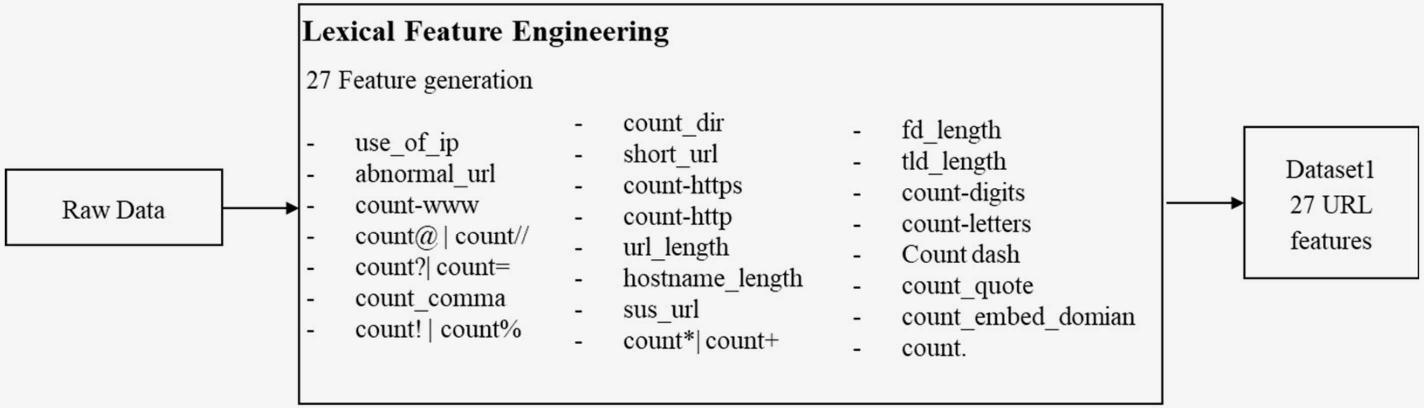
Work Objectives

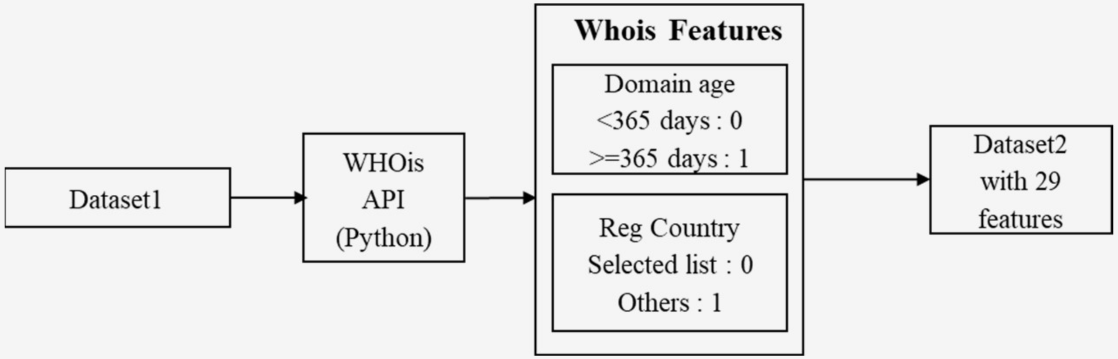
- Validate application of Parametric & non-parametric (**ANN & Ensemble)** classifiers for multi categorization of URLs
- Use lexical and host features (combination) for building best **Multi classification** model
- Evaluate performance of discriminative features
- Explore the **influence of feature variation** on detection result of parametric and parametric models.
- Build a **sustainable** (low code) Web **Solution** for **URL prediction**
- **Integrate third party tool** (Virus Total) for parallel validation.

Project Methodology

Data Preparation and Feature Engineering Phase

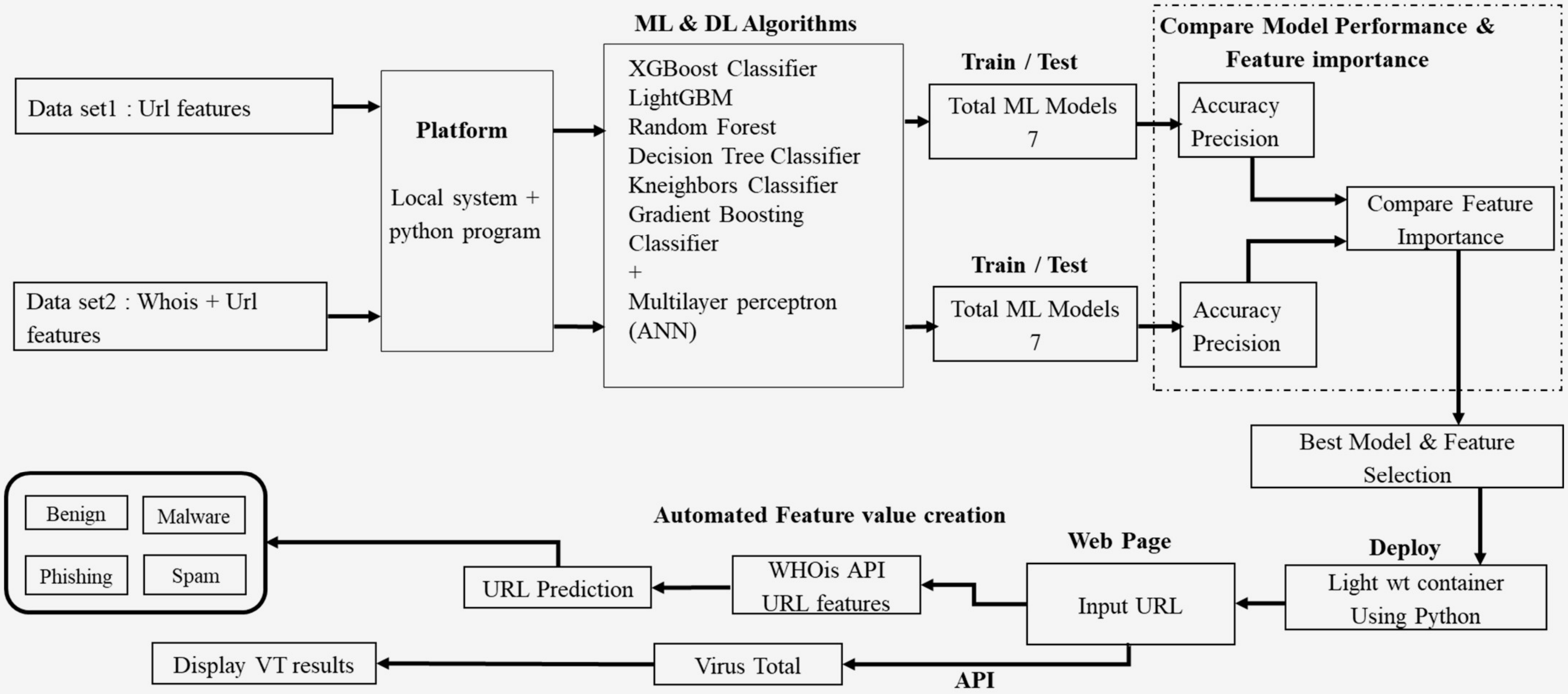
Datasets creation: 2





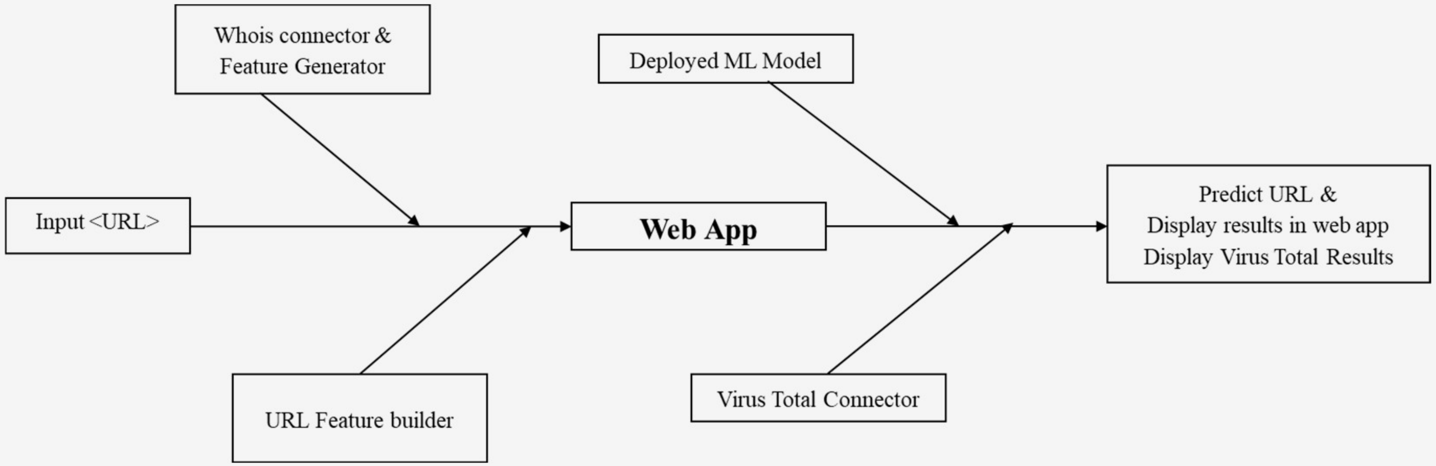
Project Methodology

High level Workflow



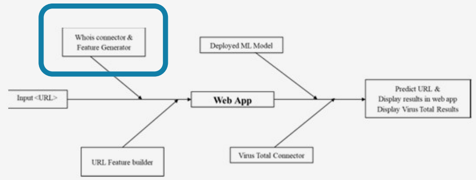
Web App Design

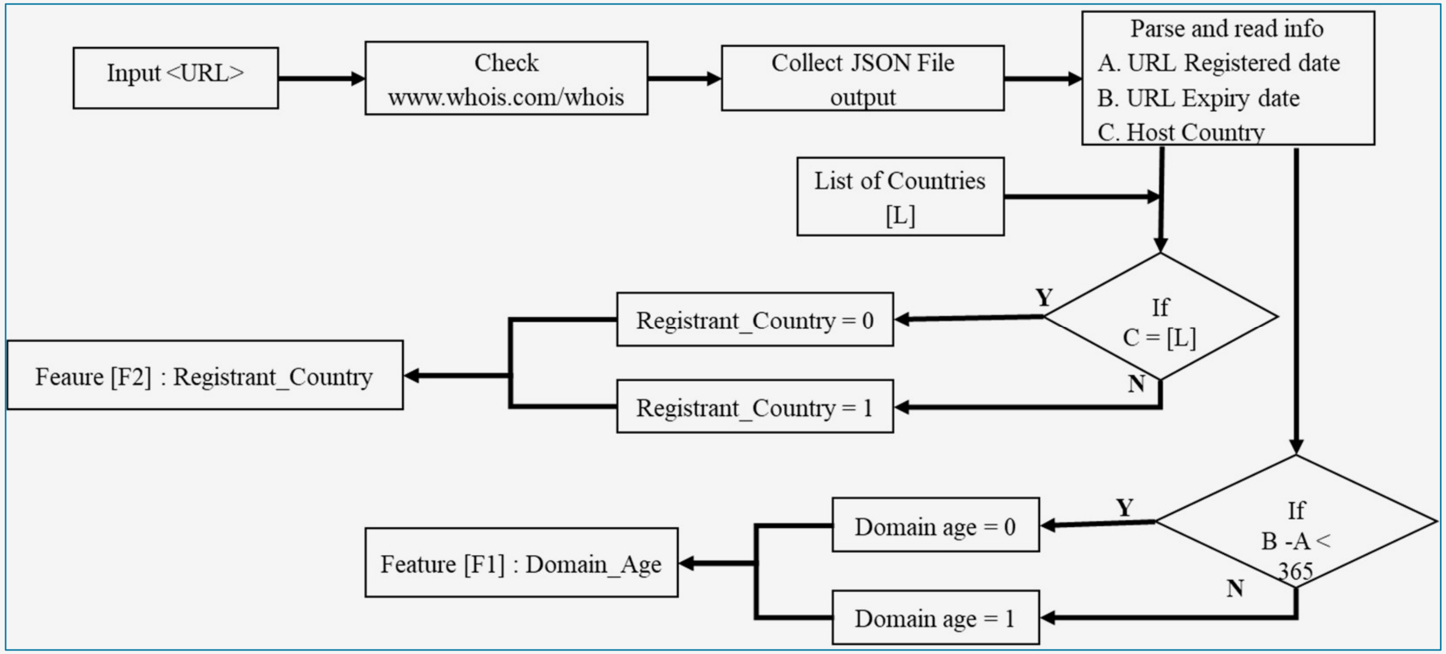
Integration of modules



Module 1 Workflow

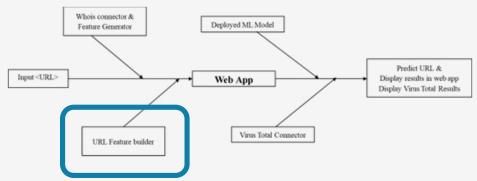
Whois Feature Generator





Module 2 Workflow

Lexical Feature Builder

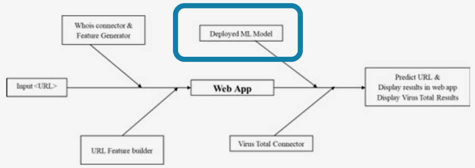


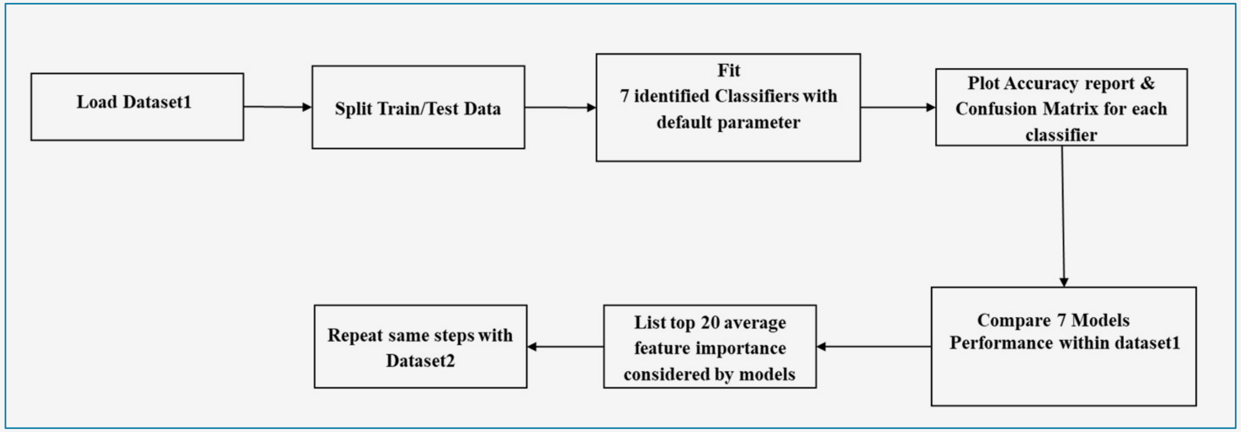
4	use_of_ip	106230	non-null	int64
5	abnormal_url	106230	non-null	int64
6	count.	106230	non-null	int64
7	count-www	106230	non-null	int64
8	count@	106230	non-null	int64
9	count//	106230	non-null	int64
10	count?	106230	non-null	int64
11	count=	106230	non-null	int64
12	count_comma	106230	non-null	int64
13	count*	106230	non-null	int64
14	count+	106230	non-null	int64
15	count!	106230	non-null	int64
16	count%	106230	non-null	int64
17	count-	106230	non-null	int64
18	count_quote	106230	non-null	int64
19	count_dir	106230	non-null	int64
20	count_embed_domian	106230	non-null	int64
21	short_url	106230	non-null	int64
22	count-https	106230	non-null	int64
23	count-http	106230	non-null	int64
24	url_length	106230	non-null	int64
25	hostname_length	106230	non-null	int64
26	sus_url	106230	non-null	int64
27	fd_length	106230	non-null	int64
28	tld_length	106230	non-null	int64
29	count-digits	106230	non-null	int64
30	count-letters	106230	non-null	int64

REVA UNIVERSITY
Bengaluru, India
Established as per the section 2(f) of the UGC Act, 1956,
Approved by AICTE, New Delhi

Module 3 – Model Building & Deployment

Model Performance validation and Best model selection

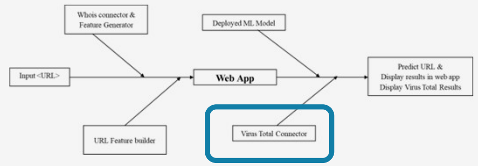


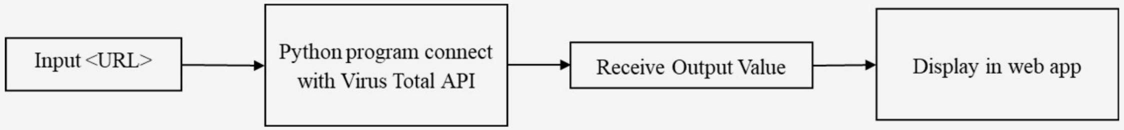


REVA UNIVERSITY
Bengaluru, India
Established as per the section 2(f) of the UGC Act, 1956, Approved by AICTE, New Delhi

Module 4 – Virus total Validation

Integration Workflow





REVA UNIVERSITY
Bengaluru, India
Established as per the section 2(f) of the UGC Act, 1956,
Approved by AICTE, New Delhi

Project Methodology

Algorithms Selection

1. Parametric Classifier

- Multilayer perceptron (MLP) – ANN

Algo assume data to follow certain distributions

2. Nonparametric Classifiers

- Decision Tree Classifier (DT).
- K-Nearest Neighbor Classifier (KNN).

Ensemble Algorithms :

- Gradient Boosting Classifier.
- Random Forest Classifier (RF).
- Light Gradient Boosting Machine (LGBM).
- Extreme Gradient Boosting Classifier (XGBoost).

Algo doesn't assume data to follow certain distributions

+

use multiple learning algorithms to obtain better predictive performance

Resource Specifications

Software | Hardware | Others

Local Machine Configuration	
System	ACER-NITRO 5 AMD
Operating System	Windows 11 Home Single Language
Processor	AMD Ryzen™ 5 4600H (3.00 GHz base frequency, uptoMax. Boost Clock 4.0GHz, 8 MB cache, 6 cores, 12 Threads)
Memory	32 GB DDR4 3200MHz (2 x 16 GB)
Graphics	AMD Radeon™ Graphics (6 core, 1500 MHz frequency)
Hard Disk	256 GB PCIe® NVMe™ M.2 SSD + 1 TB HDD
Machine Learning Software and Libraries	
Language	Python 3.9.12
Development Environment and GUI	Anaconda Navigator, Jupter, IDLE
Libraries	Pandas, Scikit Learn, confusionmatrix, whois, Numpy, Matplotlib, Seaborn, GridSeaarchCV

REVA UNIVERSITY
Bengaluru, India
Established as per the section 2(f) of the UGC Act, 1956, Approved by AICTE, New Delhi

Resource Specifications

Software | Hardware | Others

Program modules and Web UI	
Virus total Connector	Python
Web application	Python, Streamlit
Deployment	Lightweight container using Python libraries

REVA UNIVERSITY
Bengaluru, India
Established as per the section 2(f) of the UGC Act, 1956,
Approved by AICTE, New Delhi

Data & Feature Vizualisation

Exploratory Data Analysis

Multiclass Dataset distribution

Label	No. of URLs
Benign	26144
Malware	27424
Phishing	26773
Spam	25889
Total URLs: 1,06,230	

Out[7]:

	url	type
0	http://www.keve-kiserdo.hu/portal/index.php?op...	spam
1	www.agdealer.com/keystone	phishing
2	sbcspa.tripod.com/sbcs/	phishing
3	hockeydraft.ca/players/profile.aspx?id=4132&na...	benign
4	210.37.11.238/jm32/includes/site/config.bin	malware

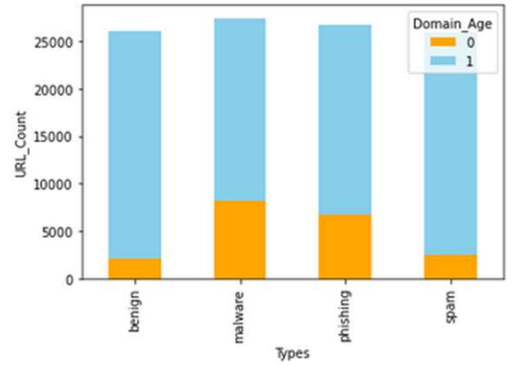
~ Equal distribution

Features List

```
df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 106230 entries, 0 to 106229
Data columns (total 32 columns):
 #   Column             Non-Null Count   Dtype
---  ------             --------------   -----
 0   url                106230 non-null  object
 1   type               106230 non-null  object
 2   Domain_Age         106230 non-null  int64
 3   Registrant_Country 106230 non-null  int64
 4   use_of_ip          106230 non-null  int64
 5   abnormal_url       106230 non-null  int64
 6   count.             106230 non-null  int64
 7   count-www          106230 non-null  int64
 8   count@             106230 non-null  int64
 9   count//            106230 non-null  int64
 10  count?             106230 non-null  int64
 11  count=             106230 non-null  int64
 12  count_comma        106230 non-null  int64
 13  count*             106230 non-null  int64
 14  count+             106230 non-null  int64
 15  count!             106230 non-null  int64
 16  count%             106230 non-null  int64
 17  count-             106230 non-null  int64
 18  count_quote        106230 non-null  int64
 19  count_dir          106230 non-null  int64
 20  count_embed_domian 106230 non-null  int64
 21  short_url          106230 non-null  int64
 22  count-https        106230 non-null  int64
 23  count-http         106230 non-null  int64
 24  url_length         106230 non-null  int64
 25  hostname_length    106230 non-null  int64
 26  sus_url            106230 non-null  int64
 27  fd_length          106230 non-null  int64
 28  tld_length         106230 non-null  int64
 29  count-digits       106230 non-null  int64
 30  count-letters      106230 non-null  int64
 31  type_code          106230 non-null  int32
dtypes: int32(1), int64(29), object(2)
memory usage: 25.5+ MB
```

Domain Age vs Type Label

```
#Plotting Domain age vs type lable. domain age zero is <365 days
cross_tab_prop.plot(kind='bar', stacked=True, color=['orange','skyblue'])

plt.xlabel('Types')
plt.ylabel('URL_Count')
plt.show()
```



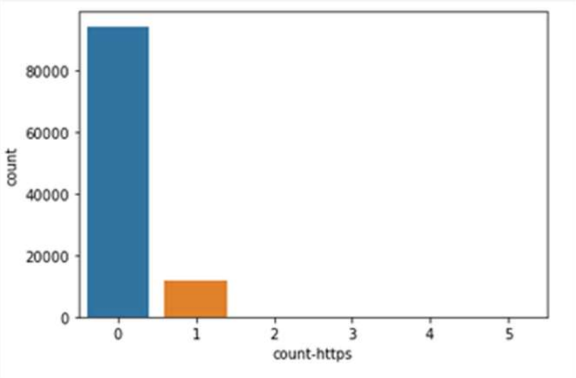
Young urls contribute more to Malware and Phishing attacks

Feature Analysis

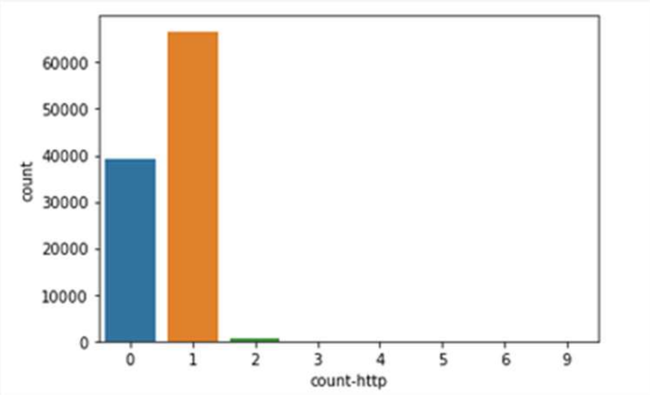
Exploratory Data Analysis

REVA UNIVERSITY
Bengaluru, India
Established as per the section 2(f) of the UGC Act, 1956,
Approved by AICTE, New Delhi

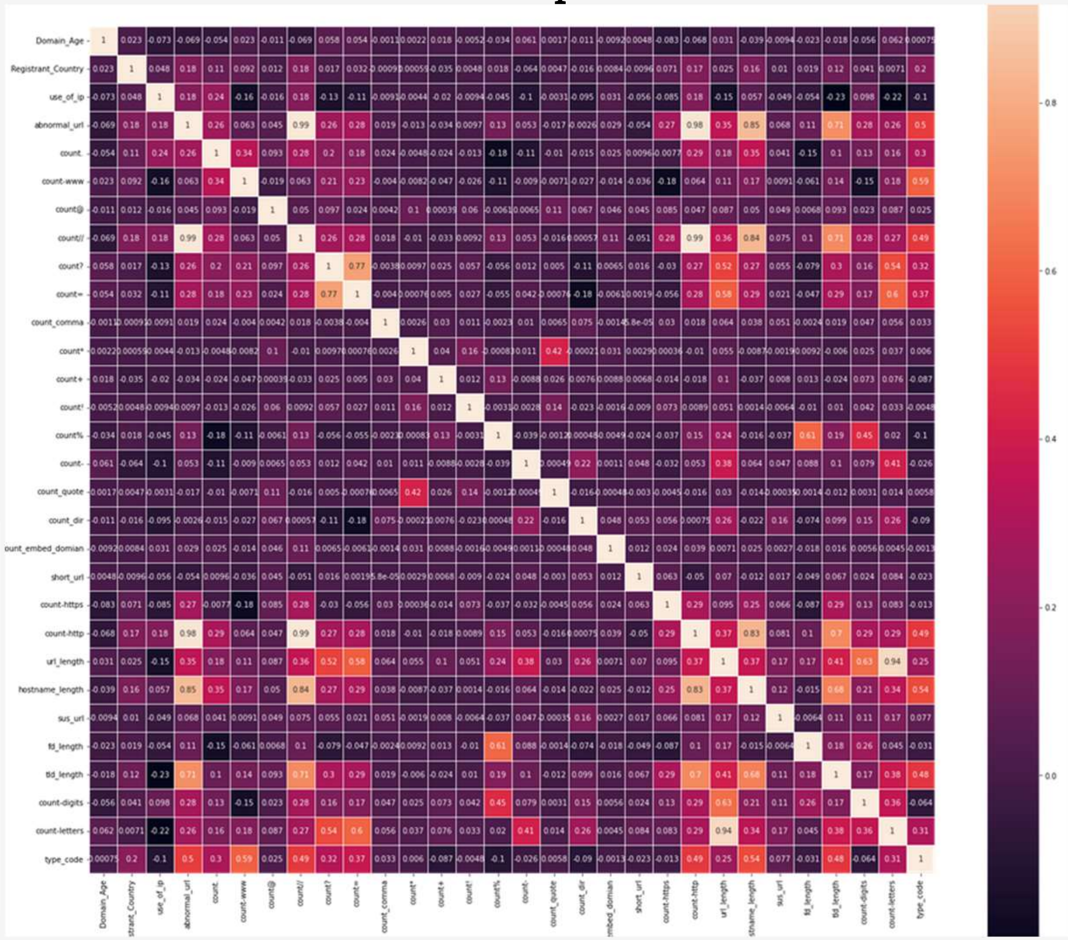
Bening urls use https



Malware urls use http



Heatmap



Few correlated features

Lot of these make sense, for example longer URLs will have more letter counts.

Feature dependency Analysis

REVA UNIVERSITY
Bengaluru, India

Established as per the section 2(f) of the UGC Act, 1956,
Approved by AICTE, New Delhi

Exploratory Data Analysis

```
# looking at the correlations of the features.
corr = df.corr()
corr.style.background_gradient(cmap='coolwarm')
```

	Domain_Age	Registrant_Country	use_of_ip	abnormal_url	count.	count-www	count@	count//	count?	count=	count_c
Domain_Age	1.000000	0.023014	-0.073401	-0.068769	-0.054421	0.022927	-0.010675	-0.069397	0.058018	0.054067	-0.
Registrant_Country	0.023014	1.000000	0.048292	0.178048	0.107565	0.092166	0.011783	0.176254	0.016851	0.031856	-0.
use_of_ip	-0.073401	0.048292	1.000000	0.184378	0.235070	-0.161901	-0.016122	0.182650	-0.131509	-0.105973	-0.
abnormal_url	-0.068769	0.178048	0.184378	1.000000	0.264189	0.062655	0.044637	0.987431	0.256590	0.281226	0.
count.	-0.054421	0.107565	0.235070	0.264189	1.000000	0.336510	0.092797	0.280689	0.197741	0.176335	0.
count-www	0.022927	0.092166	-0.161901	0.062655	0.336510	1.000000	-0.018764	0.062522	0.207871	0.229186	-0.
count@	-0.010675	0.011783	-0.016122	0.044637	0.092797	-0.018764	1.000000	0.049754	0.096523	0.023840	0.
count//	-0.069397	0.176254	0.182650	0.987431	0.280689	0.062522	0.049754	1.000000	0.263975	0.281807	0.
count?	0.058018	0.016851	-0.131509	0.256590	0.197741	0.207871	0.096523	0.263975	1.000000	0.766861	-0.
count=	0.054067	0.031856	-0.105973	0.281226	0.176335	0.229186	0.023840	0.281807	0.766861	1.000000	-0.
count_comma	-0.001116	-0.000915	-0.009114	0.018521	0.024412	-0.004007	0.004235	0.017912	-0.003809	-0.004024	1.
count*	0.002165	0.000588	-0.004394	-0.012770	-0.004836	-0.008199	0.100517	-0.010123	0.009729	0.000760	0.
count+	0.017654	-0.035301	-0.020303	-0.034351	-0.023789	-0.047315	0.000394	-0.032700	0.024834	0.004961	0.
count!	-0.005228	0.004780	-0.009369	0.009676	-0.013332	-0.025966	0.060015	0.009197	0.056684	0.026989	0.
count%	-0.033765	0.018094	-0.044531	0.133677	-0.183699	-0.111432	-0.006051	0.131002	-0.056374	-0.054606	-0.
count-	0.061249	-0.064133	-0.103820	0.052584	-0.109882	-0.008979	0.006511	0.053414	0.011582	0.042044	0.
count_quote	0.001725	0.004748	-0.003078	-0.016676	-0.010465	-0.007129	0.111983	-0.016470	0.005033	-0.000760	0.
count_dir	-0.010983	-0.016162	-0.095491	-0.002605	-0.014742	-0.027488	0.066675	0.000572	-0.106845	-0.178857	0.
count_embed_domian	-0.009223	0.008363	0.030701	0.028899	0.024585	-0.014188	0.046268	0.108175	0.006542	-0.006095	-0.
short_url	0.004811	-0.009621	-0.056167	-0.054193	0.009587	-0.036374	0.045181	-0.050502	0.015689	0.001855	0.
count-https	-0.083262	0.071039	-0.085025	0.269740	-0.007671	-0.183972	0.085190	0.281477	-0.029702	-0.056115	0.
count-http	-0.068211	0.173655	0.176210	0.979016	0.291526	0.063632	0.046762	0.987062	0.269237	0.284349	0.
url_length	0.030955	0.025444	-0.148018	0.350073	0.181554	0.109536	0.087309	0.358649	0.517528	0.583552	0.
hostname_length	-0.038855	0.155680	0.057293	0.847418	0.350575	0.170016	0.050126	0.836076	0.270153	0.292627	0.
sus_url	-0.009368	0.010184	-0.048611	0.067584	0.041264	0.009071	0.048533	0.074773	0.054857	0.021439	0.
fd_length	-0.022944	0.019434	-0.054034	0.107691	-0.154385	-0.060833	0.006781	0.103502	-0.078981	-0.046535	-0.
tld_length	-0.018145	0.117883	-0.225965	0.714835	0.104452	0.137557	0.092935	0.706669	0.295114	0.286081	0.
count-digits	-0.056062	0.041161	0.098106	0.280604	0.130220	-0.153136	0.022689	0.279927	0.164344	0.171955	0.
count-letters	0.061694	0.007127	-0.219500	0.263701	0.160162	0.183642	0.086505	0.273885	0.537357	0.599672	0.
type_code	0.000748	0.195481	-0.101869	0.502632	0.295175	0.590776	0.025201	0.493702	0.323854	0.372768	0.

Visualize the strength of correlation among features

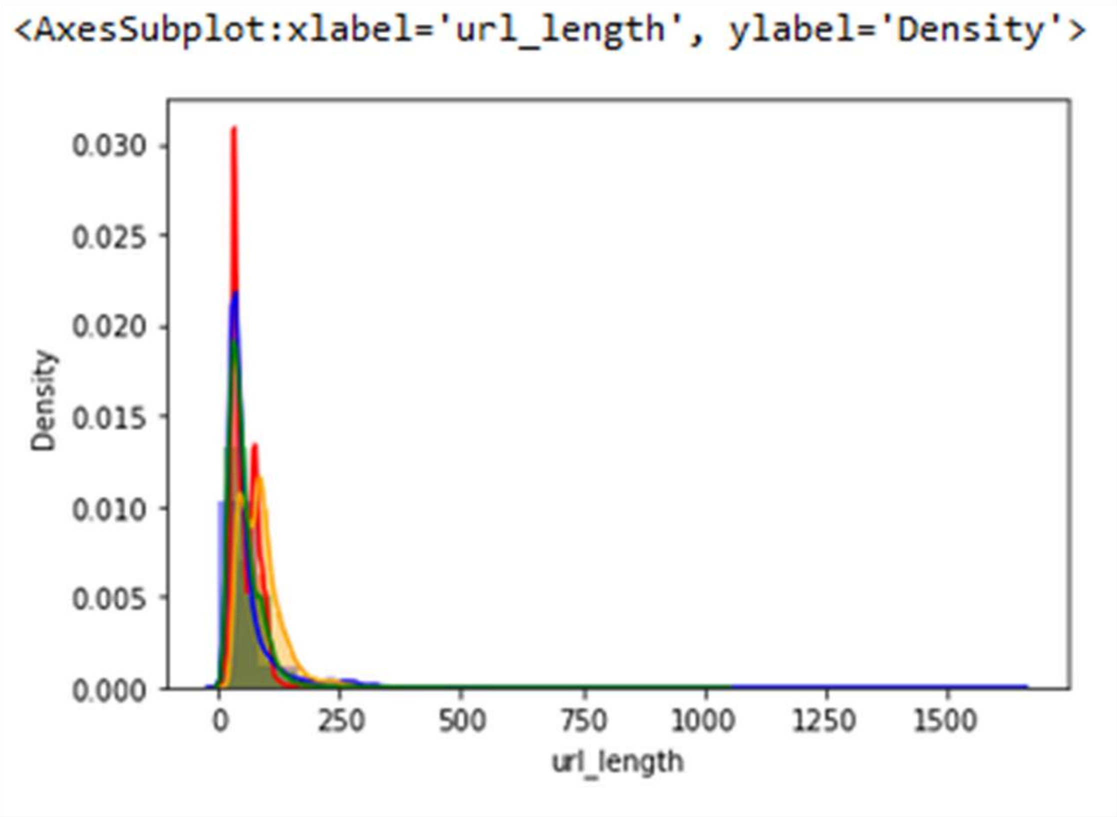
Background Gradient

It is common abnormal URL strongly related to more “//”.

Feature Interpretation

Exploratory Data Analysis

URL Length distribution per Category




```
# malware rep as 1
#phishing rep as 2
#benign rep as 0
#spam rep as 3

sns.distplot(df.loc[df['type_code'] == 1]['url_length'], bins = 20, color='red')
sns.distplot(df.loc[df['type_code'] == 2]['url_length'], bins = 20, color='blue')
sns.distplot(df.loc[df['type_code'] == 3]['url_length'], bins = 20, color='orange')
sns.distplot(df.loc[df['type_code'] == 0]['url_length'], bins = 20, color='green')
```

malware and phishing URLs having common characteristics with particular URL length

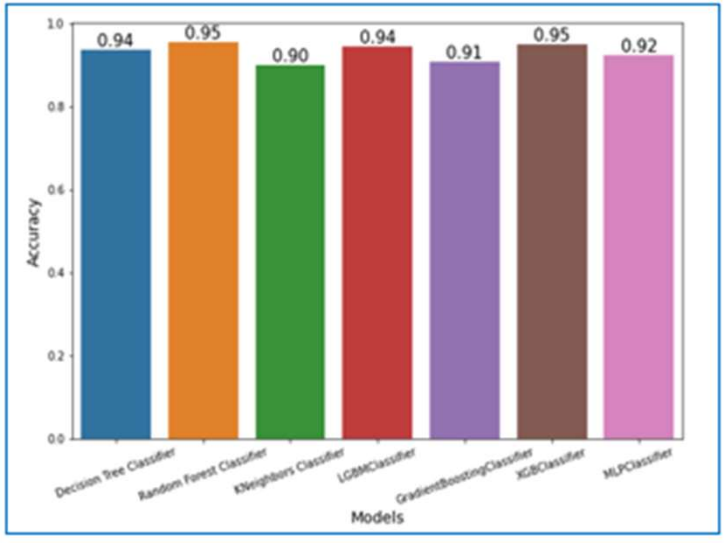
Model Performance

Dataset1 - Experiment Results-Default Parameters

XGBoost classifiers and Random Classifier gave maximum accuracy and precision of 95%

Hyperparameters : Default

Classifier Type	Classifier Name	Accuracy	Precision	Recall	F1
Non-parametric	DT	94%	90%	87%	89%
	RF	95%	93%	90%	91%
	KNN	90%	88%	80%	83%
	LGBM	94%	92%	88%	90%
	GB	91%	89%	81%	85%
	XGBoost	95%	93%	95%	94%
Parametric	MLP	92%	90%	82%	87%



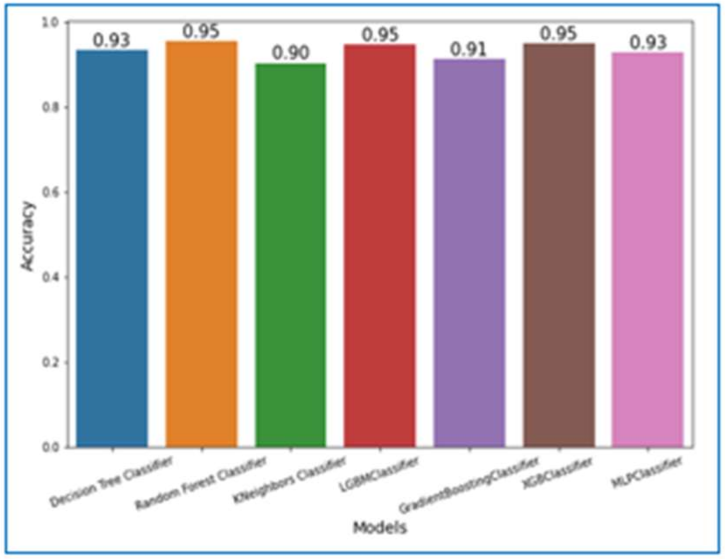
Model Performance

Dataset2 - Experiment Results-Default Parameters

3 models performed with accuracy of 95% which are LGBM classifier, XGB Classifier and RF classifier.

Hyperparameters : Default

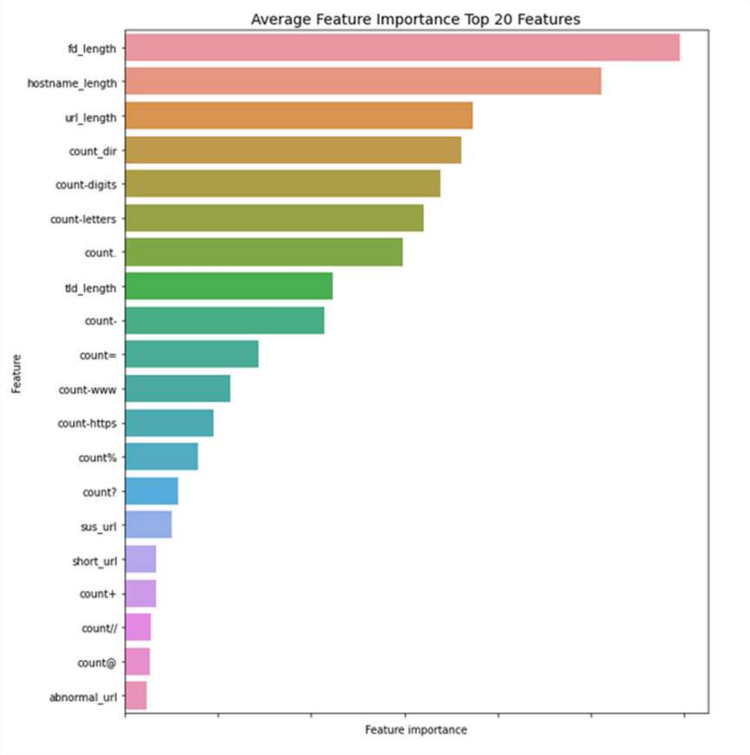
Classifier Type	Classifier Name	Accuracy	Precision	Recall	F1
Non-parametric	DT	93%	93%	87%	88%
	RF	95%	93%	88%	92%
	KNN	90%	88%	80%	84%
	LGBM	95%	96%	88%	91%
	GB	91%	90%	82%	86%
	XGBoost	95%	95%	96%	97%
Parametric	MLP	93%	93%	90%	89%



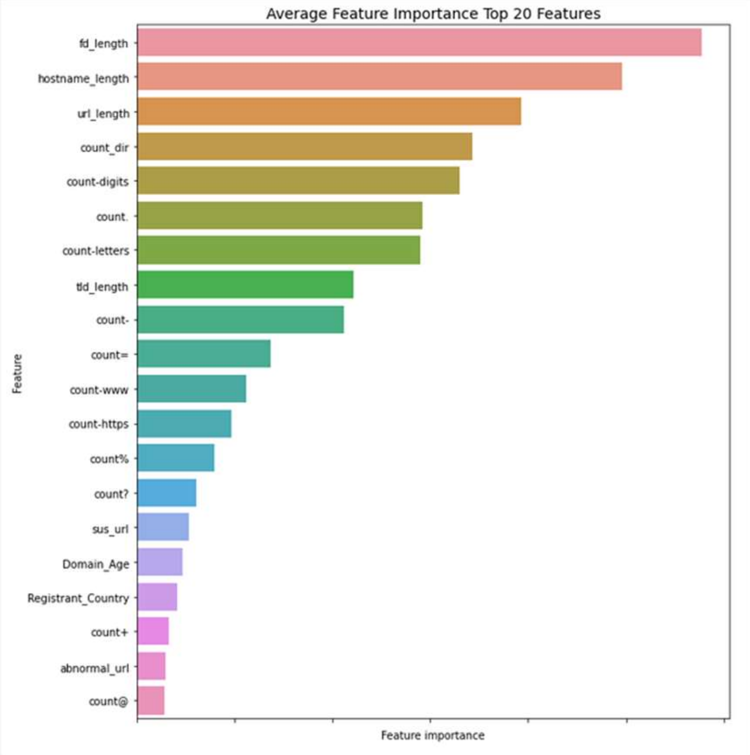
Feature performance validation

Comparison - Feature Importance

Top 20 without Whois Feature : Dataset1



Top 20 with Whois Feature : Dataset2



Implementation

Model comparison across datasets

Summary of Models Performance

Name of Classification Algorithm	Prediction Accuracy: With URL features	Prediction Accuracy: With URL Features + Whois Features	Trend
Decision Tree Classifier	0.94	0.93	↘
Random Forest Classifier	0.95	0.95	→
Kneighbors classifier	0.9	0.9	→
LGBM Classifier	0.94	0.95	↗
Gradient Boosting Classifier	0.91	0.91	→
XGB Classifier	0.95	0.95	→
MLP Classifier	0.92	0.93	↗

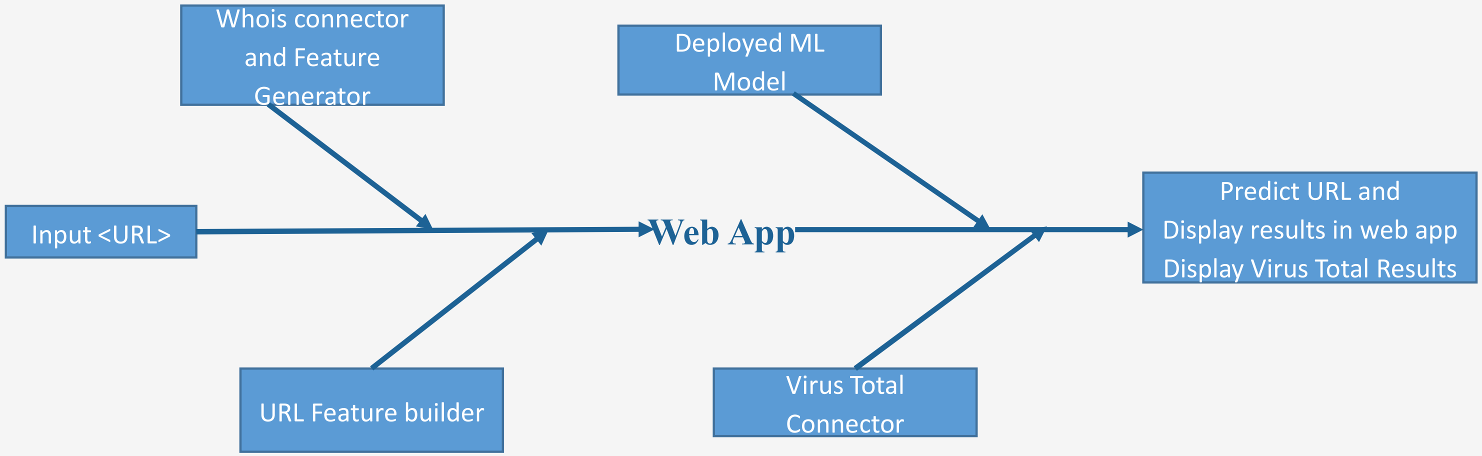
Comparative analysis of performance indicators with previous works

Classifier Name	Accuracy	Precision	Recall	F1
Asela et al. [24]	82.37%	89.78%	58.22%	70.64%
Rohit et al. [23]	89.00%	88.00%	88.00%	92.00%
Chiramdasu et al. [22]	91.00%	62.00%	85.00%	90.00%
Eduardo et al. [9]	93.47%	93.63%	93.28%	93.46%
Chen et al. [8]	85.99%	87.19%	84.66%	85.01%
Zhao et al. [5]	90.31%	90.03%	90.00%	89.00%
Proposed work	**95.02%**	**95.04%**	**96.10%**	**96.99%**

XGBoost model is selected for deployment along with Dataset2 features.

Project Code

Web App Design and Integration of modules



GitHub repository: https://github.com/svijayar1/-Building-a-Deep-Learning-based-Multi-class-prediction-model-for-Malicious-URLs-Proj2-

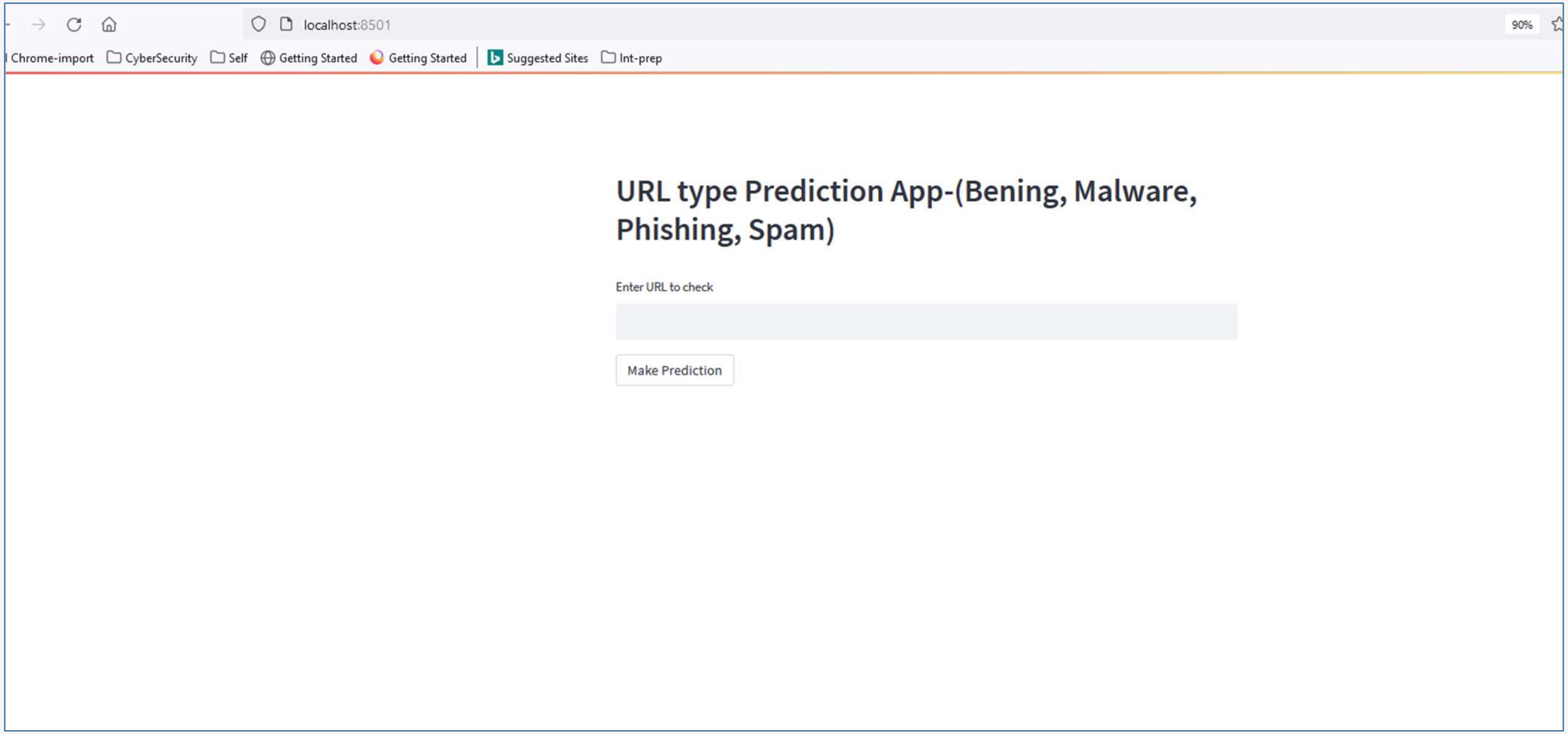
Code

```
import numpy as np

st.header("URL type Prediction App-(Bening, Malware, Phishing, Spam)"

df=pd.read_csv('C:\\Users\\vijay\\Desktop\\Python-programs\\multiclas

#Predictor Variables
X = df[['Domain_Age','Registrant_Country','use_of_ip','abnormal_url',
        'count//', 'count_dir', 'count_embed_domian', 'short_url', 'cc
        'count-http', 'count%', 'count?', 'count-', 'count=', 'url_len
        'hostname_length', 'sus_url', 'fd_length', 'tld_length', 'coun
        'count-letters', 'count_comma', 'count*',
        'count+', 'count!','count_quote']]

#Target Variable
y = df['type_code']

X_train, X_test, y_train, y_test = train_test_split(X, y, stratify=y,

#Load model
model = xgb.XGBClassifier()
model.load_model("xgb_model.json")

#Feature Engineering
#df1 = pd.DataFrame({'url':[st.text_input('Enter URL to check')], 'Dc
#testin line
input_var=st.text_input('Enter URL to check')

df1data = {'url':input_var}


#whois info collection
from openpyxl import load_workbook
from os.path import exists
import sys
import json
import whois
from datetime import datetime

def getDaysDiff(whoi info):
```

Implementation

Deploy Model and Build web application

Streamlit Webpage at launch

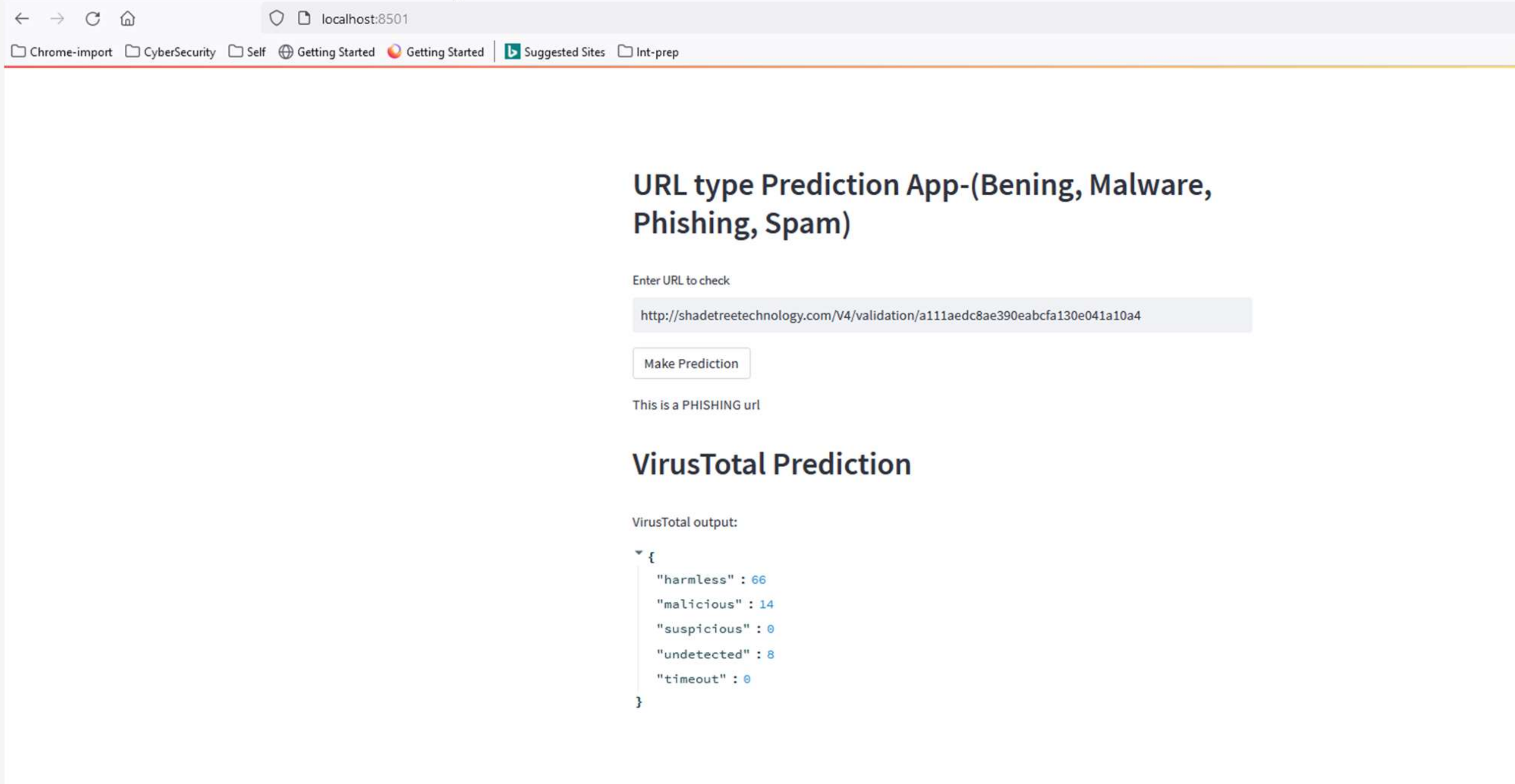


REVA UNIVERSITY
Bengaluru, India
Established as per the section 2(f) of the UGC Act, 1956, Approved by AICTE, New Delhi

Testing and Validation

Checking Input URL and Output

Webpage Test and Results



Analysis and Results

Key Findings | Insights

Nonparametric Ensemble classifier LGBM and Parametric classifier MLP performance improved as we enrich data with more features

Ensemble models like XGBoost and Random Forest demonstrate consistency staying high with prediction accuracy

Non parametric Ensemble models, perform better than parametric (ANN) models for multiclass classification.

Domain features influence both ML and Neural network model performance

Combination of Lexical and Whois features adds value in identifying malicious URL under multiclass set.

First directory, hostname length, URL length, directory and digit counts having maximum impact on model performances.

Suggestions and Conclusion

Insights | Next Step | Future Scope

- Multiclass work can be considered in the perspective of multilabel learning with little finetune in the methodology keeping class as malicious with labels as Malware, Phishing and Spam
- DNS Feature inclusion and further additional feature from whois information can be considered in future work.
- Model efficiency can be increased by fine tunning hyper parameters
- Application of Deep learning algorithms on multi classification of URLs can be considered for future study

REVA UNIVERSITY

Bengaluru, India

Established as per the section 2(f) of the UGC Act, 1956,
Approved by AICTE, New Delhi

Thank you!