URL Classification

678 Big Data Technologies

Prof David Belanger

Business Intelligence and Analytics Spring 2022

Stevens Institute of Technology, Hoboken

Sanket Dahule: Ingestion and Flow of data in components of Model pipeline

Gaurav Ghodke: Preprocessing on unclean and untidy raw data into structured data

Ayush Ashutosh: Custom Feature Extraction and Engineering for Tuning Models

Siddhesh More: Data Privacy and Ethics compliance on model extension and its applications

Section 1: Problem Overview and Motivation

Malicious websites are the primary perpetrators of cyber risks in the current day. They serve as a front for the distribution of viruses, trojans, and other dangerous software that may infect the user, resulting in data theft and money laundering. These URLs are often circulated by e-mail links, pop-up advertising, and embedded downloads.

The significance of URLs in document representation cannot be emphasized. To express a URL's functional purpose or genre, shorthand mnemonics such as "wiki" or "blog" are frequently inserted. Other mnemonics have emerged because of use (for example, a Wordpress particle strongly suggests blogs). Can we use this predictive capacity to infer the genre of a document from the representation of a URL, or perhaps label it as malicious? Experiments utilizing machine learning approaches to test this claim provide promising results, and a novel algorithm for character n-gram decomposition is offered.

One advantage of this strategy is its quickness. The length of a URL is a minuscule fraction of the length of a typical Web page. Because of the lower number of nonzero features, this allows for significantly faster production of feature vectors, as well as faster classification. However, there are times when the content of a Web page is just unavailable. This occurs, for example, during information filtering. An institution may choose to utilize a blacklist of themes (for example, pornography) or a whitelist of topics (for example, science), and only allow its users access to the corresponding pages. Before downloading a page, access should ideally be refused.

On certain sites, the material may also be disguised in graphics. In this case, any classification algorithm has simply the URL to work with. If such a system can forecast the topic of a hyperlink before downloading the page, it can reduce bandwidth waste caused by irrelevant sites.

This skill might be beneficial for improving tailored search results, disambiguating material, crawling the Web effectively in search of relevant pages, and constructing behavioral profiles from clickstream data without processing the whole document.

Section 2. Data

This report uses ISCX-URL-2016 URL dataset and the ODP dataset. The former dataset consists of classes for different malicious urls while the later focuses on the topic represented by the websites content. From the uses ISCX-URL-2016 URL dataset lexical features were extracted from URLs, and 5 URL classes such as benign, defacement, malware, phishing, and spam were labelled in the dataset. The different classes of URLs are briefly introduced below.

Benign URLs: Benign URLs are legitimate URLs that do not lead to any infectious websites and do not try to inject the user's computer with any kind of harmful malware. Benign websites may contain advertisements and adware which are typically harmless to a computer.

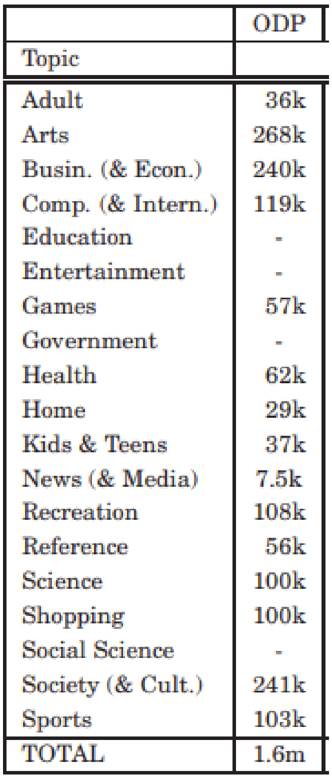
Defacement URLs: Website defacement usually means changing a certain aspect of the website such as its visual appearance and some contents on it. Hacktivists try to deface a website for numerous reasons. This kind of activity is done when some content on the web page needs to be changed without the permission of the original owner of the website which technically means penetrating a website.

Malware URLs: Malware URLs take a user to the malicious website that typically installs some malware on that user's device which can be used for identity theft, corrupt files, and even logging keystrokes. Malware can indirectly be a dangerous software that can harm a computer and steal someone's private information. Some threats such as harmful biological agents, a terrorist cell intent on disrupting operations, etc. can be considered malware. Some examples of malware are ransomware, spyware, and scareware, among others.

Phishing URLs: Phishing URLs conventionally entice a user to visit a fake website and will try to steal as much information they can get from the user. Sometimes a user can easily be led to phishing websites just by having a typo in a URL. Phishing can be defined as the intent of the hacker to steal some private information like credit card number and other digital identity by employing social engineering techniques.

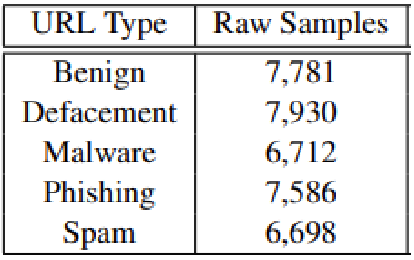
Spam URLs: Spam is a way of sending unsolicited emails to the user with the intent of advertisements or for serious harm to the computer. Spam URLs are usually seen in spam emails. Some spam URLs can be harmful and can infect the computer of the user with spyware and adware.

The Open Directory Project, or ODP for short, is a directory of categorized Web sites that has been human-edited. We utilized Web sites categorized as "Adult," "Arts," "Business," "Computers," "Games," "Health," "Home," "Kids and Teens," "News," "Recreation," "Reference," "Science," "Shopping," "Society," and "Sports." The extra "World" category, which contained solely foreign-language Web sites, was not used, but the "International" subtopic of "Kids and Teens" was retained.



	ODP
Topic	
Adult	36k
Arts	268k
Busin. (& Econ.)	240k
Comp. (& Intern.)	119k
Education	-
Entertainment	-
Games	57k
Government	-
Health	62k
Home	29k
Kids & Teens	37k
News (& Media)	7.5k
Recreation	108k
Reference	56k
Science	100k
Shopping	100k
Social Science	-
Society (& Cult.)	241k
Sports	103k
TOTAL	1.6m

ODP Dataset



URL Type	Raw Samples
Benign	7,781
Defacement	7,930
Malware	6,712
Phishing	7,586
Spam	6,698

ISCX-URL-2016 Dataset

Before building the model, we need to extract features from the URL and convert them to a numeric representation as machine learning models can't work with text data directly. For this we first break down the URL links into its subparts such as the protocol it is using, sub domain, domain name, port, path, query, parameters, and fragment. To extract such a breakdown of the URL we use a parsing library called tld.

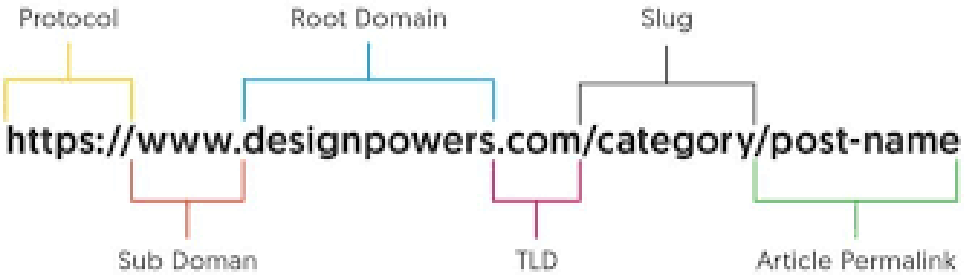


Image Reference: https://www.designpowers.com/blog/url-best-practices

For our model, we are going to extract only the subdomain, domain, top-level domain, and the full level domain which is basically the domain along with its top-level domain. For example, for the URL "en.wikipedia.org" we have the subdomain as "en", the domain as "Wikipedia", the top-level domain as "org" and the full level domain as "Wikipedia.org".

We use the full level domain also as a feature as we are going to vectorize the feature using the term frequency-inverse document frequency and hence the full level domain can give some new information if we vectorize the root domain and top-level domain together as a term.

Another we can go about the feature engineering is to create features based on length of sub part of the URL, having counts of various symbols/characters, binning the row based on which quantile it is in with respect to the URL length and ratio of different counts such as alphas: letters or punctuations: URL length.

Data Preprocessing Modules

Categorical features transformation

Normalization of numerical data

Handling unbalanced data sets(SMOTE, minority)

Feature Selection (SelectKBest)

Section 3. Model Building

We approach this problem with hybrid model approach where we first use a binary classifier to separate the benign websites from the other. After this stage we can use a multiclass classifier to identify what category does the URL falls which makes it a non-benign link.

80% TRAINING DATA & 20% TESTING DATA

FITTING DATA IN MACHINE LEARNING MODEL

FITTING BEST PARAMS IN HYPERTUNNING

For components building of different model in the pipeline we used XgBoost, Random Forest Classifier for multiclass label classification and Logistic Regression, Naives Bayes for binary classification. There are many models that are used for classification and top performing are automatically switched between as per performance upgrade.

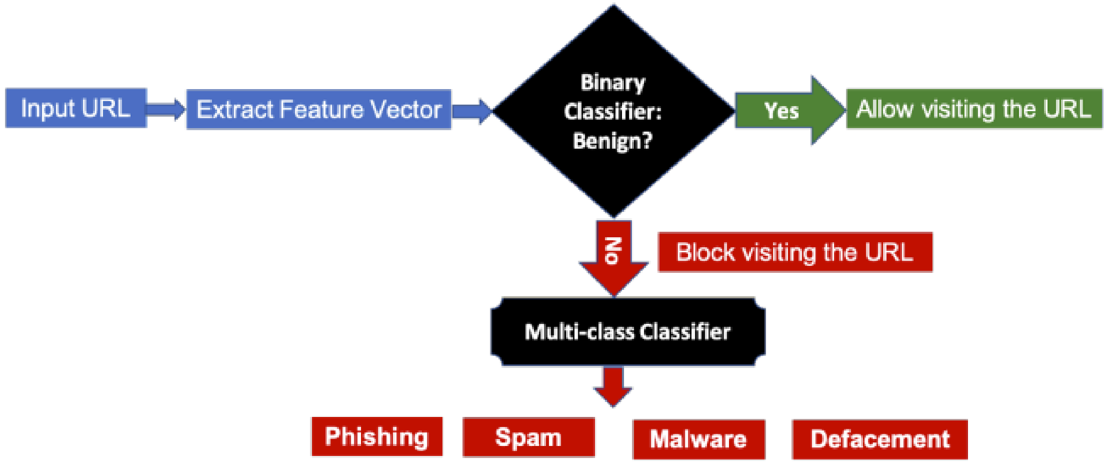
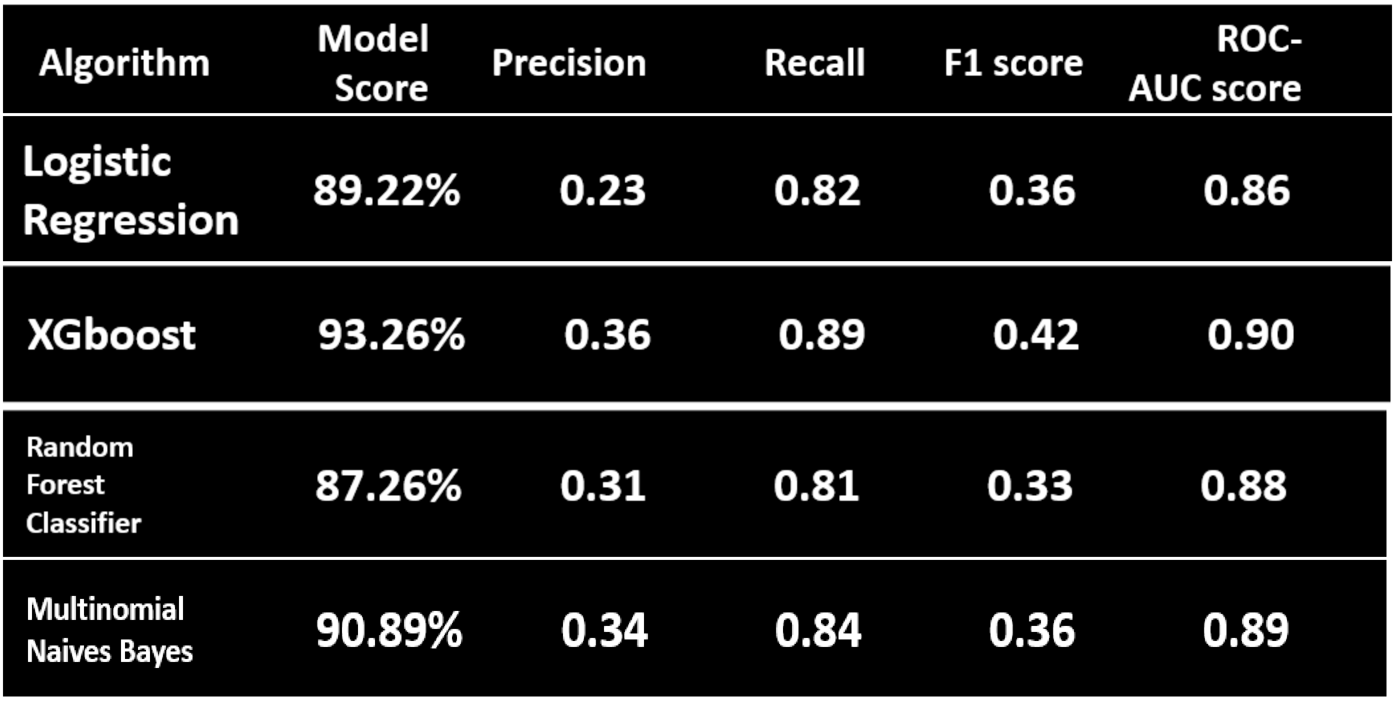


Image Reference: http://isyou.info/jowua/papers/jowua-v11n4-3.pdf

Section 4. Metric and Evaluations



Algorithm	Model Score	Precision	Recall	F1 score	ROC-AUC score
Logistic Regression	89.22%	0.23	0.82	0.36	0.86
XGboost	93.26%	0.36	0.89	0.42	0.90
Random Forest Classifier	87.26%	0.31	0.81	0.33	0.88
Multinomial Naives Bayes	90.89%	0.34	0.84	0.36	0.89

There is not definite model used as this is works on switching model principles and will be used as per the training of new data is validated. But this were time being results at a certain point of time. After ingesting data to multiple binary and multi-class classifications model the pipeline exits the output through certain threshold and switch the model accordingly. In this case, Xgboost which is a Boosting algorithm which take random samples from data and evaluates them and then run evaluations. Through this all the weak learners are ingested into the stronger model creating a better model at the end of samples.

Section 5: Conclusions, Model Limitations and Extensions

5.1 Conclusions

The results from the components from the pipeline are promising to be deployed in any batch or stream applications which takes input as URL, hence classifying the category of the URL efficiently into its correct class. Hence saving user from lot of vulnerabilities and risks associated with a particular website. Also this is very scalable model as its application is suitable in almost every internet based services and products.

5.2 Model Limitations

5.2.1Custom Feature Engineering

A lot of hands-on feature engineering is required making it harder to automate the process. One way around this is to use word embedding from a pretrained word2vec model to obtain the numerical vector representation for the breakdowns of the URL. This will lead to a better model capacity as this word embedding better embodies the semantic meaning of the words of the language.

5.3 Model Extension

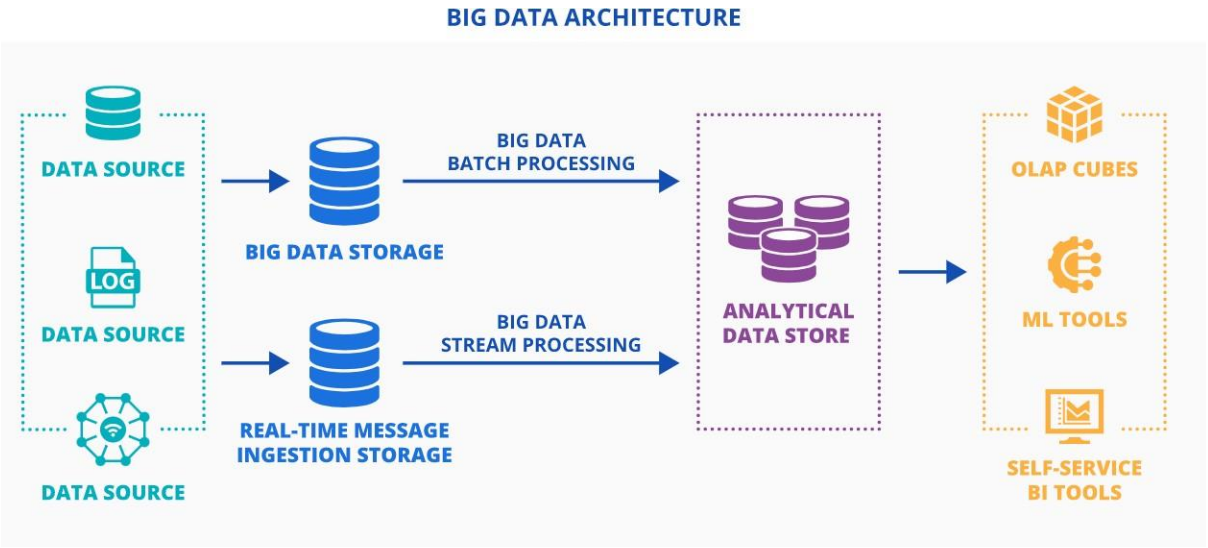


Image reference: https://www.scnsoft.com/analytics/big-data/databases

- Streaming technology (Kafka) to continuously analyze the incoming data to ingest in the application to make decisions

- Multiple Machine Learning models are deployed as per performance and are switched accordingly to predict and classify the probability of the URL classes

- Software systems to ingest, store, manipulate and make live predictions on stream data or trained batch data

Section 6: References

Papers Referred

- Classification of URL into Malicious or Benign using Machine Learning Approach Deebanchakkarawarthi G1 , Parthan AS2 , Sachin Lal3 , Surya A4 Assistant Professor, Department of Computer Engineering, JCT College of Engineering and Technology, Coimbatore, Tamilnadu, India1

- Fast webpage classification using URL features Min-Yen Kan Hoang Oanh Nguyen Thi Department of Computer Science, School of Computing, 3 science drive 2, Singapore