

**Phishing and Malware Website Detection**

**Vineet Gupta**

**Z1790212@students.niu.edu**

**Aparajita Deshpande**

**Z1761845@students.niu.edu**

**Guided By:-**

**Dr. Hamed Alhoori**

**Introduction:**

Malicious Web sites are a cornerstone of Internet criminal activities. They host a variety of unwanted content ranging from spam-advertised products, to phishing sites, to dangerous “drive-by” exploits that infect a visitor’s machine with malware. As a result, there has been broad interest in developing system to prevent the end user from visiting such sites. Identification of attack types is useful since the knowledge of the nature of a potential threat allows us to take a proper reaction as well as a pertinent and effective countermeasure against the threat.

**Previous Techniques**:

Security communities have developed blacklisting services which are based on various techniques:-

* Manual Reporting.
* Web crawlers with site analysis heuristics.

The problems associated with these techniques are:-

* Large portion of malicious web sites are too new to be checked with large number of websites being created every day.
* Some client-side system analysis the content of Web sites when are visited, which employs run-time latency and exposes users to the browser-based vulnerabilities.

**Classification of the URL’s**

* Safe: Safe website with no harm.
* Phishing: Phishing websites are fake websites that are designed to emulate a genuine bank or company webpage and asked to supply the details.
* Malware: Websites are created which harm computer, steal personal information and enable malicious individuals to cause the financial, electronic or any other kind of harm.

**Approach and Implementation:**

**Feature Extraction:**

We categorized features into four categories:

* Lexicon Based Features.
* Web site popularity Features.
* Host and Domain based Features.
* Source content Features.

**Lexicon Based Features**:

We distinguish URL into:-

* Hostname.
* Path.

Then we retrieve the bag of words such as (strings delimited by ‘/’, ‘?’, ‘.’, ‘=’, ‘-‘)

Lexicon Based Features:

* Length of hostname (Integer)
* Length of entire URL (Integer)
* Number of dots in URL (Integer)
* Average domain token length (Real)
* Longest domain token length (Integer)
* IP address presence (Binary)
* Security sensitive word presence (Binary)
* Path token count (Integer)
* Domain token count (Integer)
* Average path token length (Real)
* Longest path token length (Integer)
* Brand name presence (Binary)

**Web URL popularity Features**:

Intuitively, malicious sites have low manipulated popularity scores. So, this feature can be considered as an important factor to measure a site’s reputation.

Site popularity Features consists of three sub features:

* Number of links pointing to that site.
* Real traffic rank of that website. This can be retrieved from Alexa APIs
* Whether domain lies within a well reputable sites list

Number of the links pointing to a site can be acquired from Google. Real traffic rank feature can be retrieved from Alexa exposed APIs. Third feature is a Boolean feature, which indicates whether the domain is within a well reputable sites list. This list contains 1,000,000,000 domains with good reputation. It can be accessed from Amazon.com.

**Host and DNS Based Features**:

Based on the observation that malicious sites trend to be hosted on less popular domains. We have included following host and DNS based features:

* Domain’s autonomous system number.
* Number of resolved IPs.
* If the domain contains valid PTR record.
* Country and its corresponding IP belongs to.
* Number of registration information.

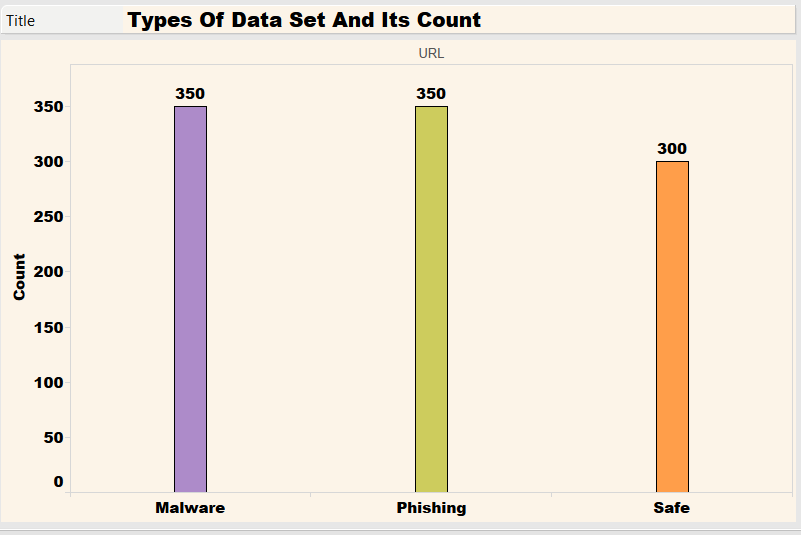
**Source Content Features**:

We calculated the total number of html tags, hyperlinks, iframe, escape, underscore and exe counts exists in web source code.

**Dataset**:

We collected 300 safe URLs from DMOZ Open Directory project. DMOZ is one of the largest human edited directory of the world. It classifies URLs into different categories.

Collected 350 samples of spam URLs from Spam domain blacklist (filtered by jwSpamSpy). Collected 350 Malware containing URLs from DNS-BH project.



**Preprocessing**:

We pre-process the features which are not consistent with others. For example, the range of traffic rank of web URL is much larger than that of other features. We map the feature into a much smaller range and it turns out to significantly increase the accuracy.

**Training**:

After carefully selecting the features, we used three learning algorithms:

* Supported Vector Machine (SVM)
* Naïve Bayes
* Random Forest

In training dataset, URLs are labeled. We used scikit-learn package to train our models and compared the results.

**Support vector machine** **(SVM):**

In a machine learning, support vector machines (SVMs) are supervised learning models with associated learning algorithms that analyze data and recognize patterns, used for classification and regression analysis. The basic SVM takes a set of input, which of two possible classes forms the output, making it a non-probabilistic binary linear classifier. An SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible.

**Naïve Bayes**:

Naïve Bayes classifier assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature. Even if the features depends on each other, all of these features contribute independently to the probability.

**Random Forest**:

Random forests are an ensemble learning method for classification (and regression) that operate by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes output by individual trees. The algorithm for including a random forest was developed by Leo Breiman and Adele Cutler. The method combines Breiman’s “bagging” idea and the random selections of features, introduced independently by Ho and Amit and Geman in order to construct a collection of decision trees with controlled variation.

**Model Evaluation Procedures**:

We used following methods to evaluate all three models (SVM, Naïve Bayes and Random Forest):

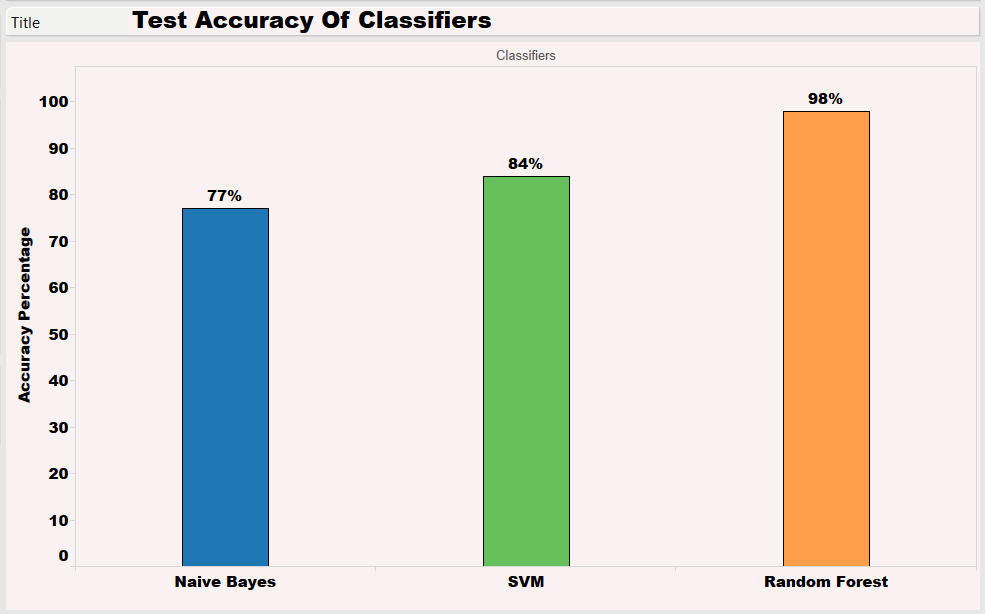
* Training / Test split.
* Cross validation.

**1. Training / Test split Method**:

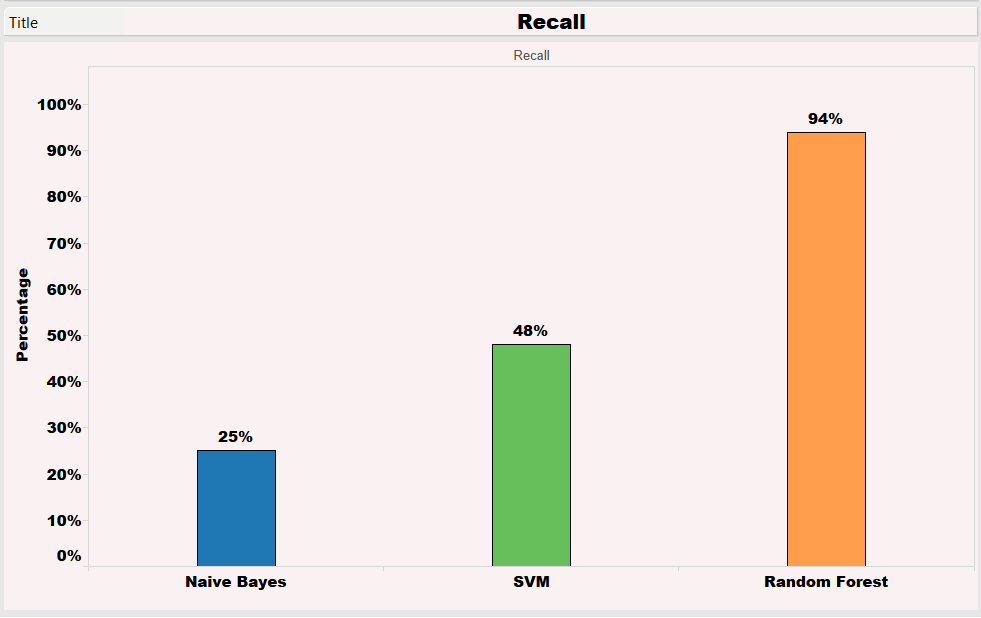
We split the dataset into two pieces knows as training and test datasets. We train the models on the training set and we evaluate the model by testing its performance on the testing sets. The resulting evaluation metrics is known as testing accuracy which is a better estimate of out of sample performance than training accuracy because we train and test the model on different sets of data which help us to avoid overfitting.

Classification Accuracy: percentage of the correct prediction. Recall: When the actual value is positive, how often is the prediction correct. Precision: When a positive value is predicted, how often is the prediction correct.

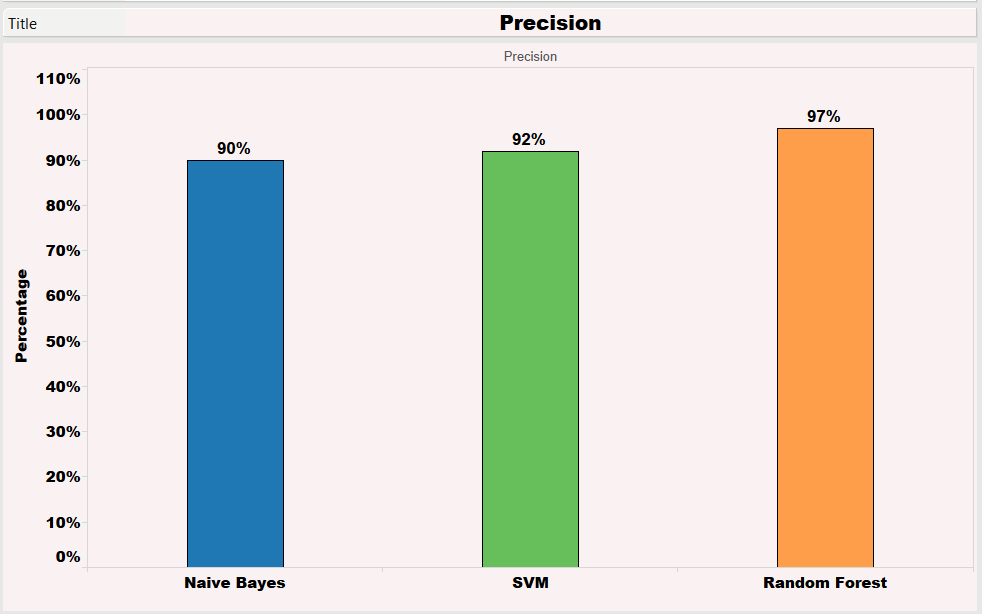
**Test Accuracy Graph**:



**Recall Graph**: Shown below.

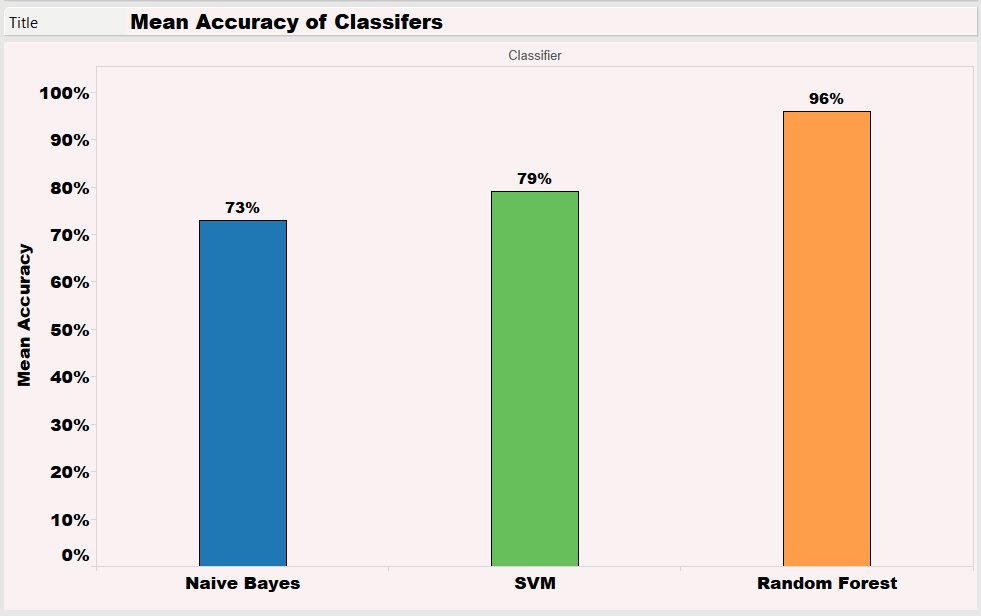


**Precision Graph**: Shown below.



**2. Cross Validation Method**:

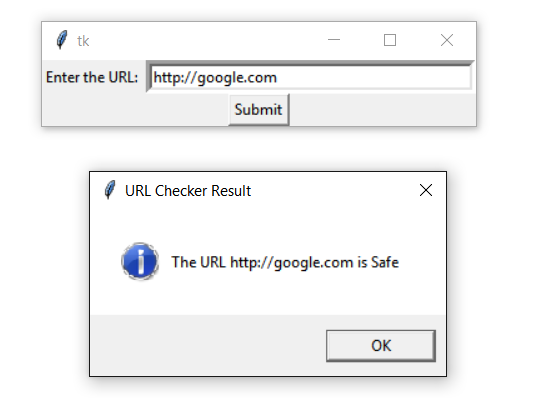
We used 10-fold method to train-test our system. This method is more accurate estimate of out-of-sample accuracy and more efficient use of data (Every observation is used for both testing and training)



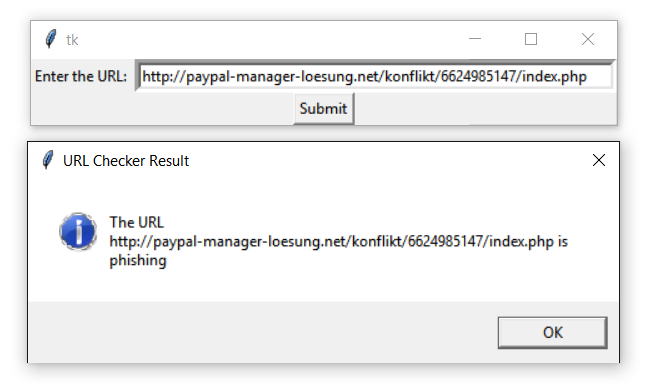
We observed that Random forest classifier gives highest mean accuracy (cross validated accuracy), precision and recall among all three classifiers, which proves that it is the best learning algorithm for Phishing and Malicious URL detection.

**Output and Conclusion:**

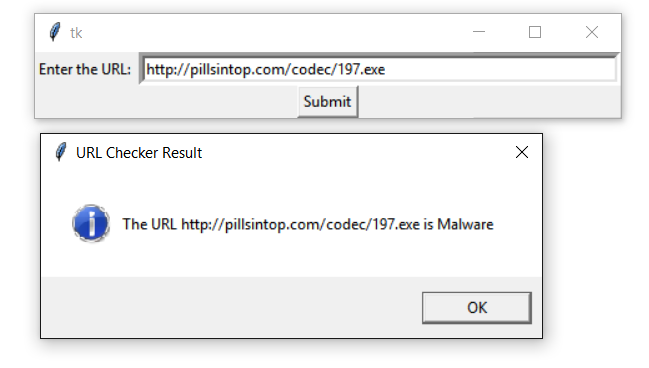
1. **Safe Website:**

****

1. **Phishing Website**:



1. **Malware Website**:



**Application and Uses**:

* Identifying spam links which prevents spams from spreading and further helps in filtering applications etc.
* URL classification for security purposes in various applications such as blacklisting, internet security software etc.
* This system can be efficiently used to identify phishing links so that user don’t fall prey to such links and don’t give away their confidential information.
* Learning based classification can be used in filtering applications for mobiles which can filter spam calls and messages from benign messages efficiently.

**Tools and Libraries used**: Python scikit, pandas, pygeoip, tkinter, Tableau

**References**:

* Learning based Phishing Websites Detection using Suspicious URLs, Haotian Liu, Xiang Pan, Zhengyang Qu, Northwestern University, IL, USA.
* Detecting Phishing Web Links and Identifying Their Attack Types, Hyunsang Choi and Heejo Lee, Korea University.
* Learning to Detect Phishing URLs, Justin Tung Ma, University of California, San Diego.