Learning to Detect Malicious URLs

Detecting malicious Web sites from lexical and host-based features of URLs

Aim

Uniform Resource Locators (URLs) are the primary means by which users locate resources on the Internet. Our goal is to detect malicious Web sites from the lexical and host-based features of their URLs. Our aim is

- binary classification of URLs where positive examples are malicious URLs and negative examples are benign URLs

Related work/Literature review

Related work/Literature review

[1] Learning to Detect Malicious URLs

JUSTIN MA, University of California, Berkeley, LAWRENCE K. SAUL, STEFAN SAVAGE and GEOFFREY M. VOELKER, University of California, San Diego.

http://cseweb.ucsd.edu/~savage/papers/TIST11.pdf

[2] Leveraging Machine Learning to Improve Unwanted Resource Filtering

Sruti Bhagavatula∗ Christopher Dunn† Chris Kanich∗ Minaxi Gupta† Brian Ziebart∗

https://www.cs.uic.edu/~ckanich/papers/bhagavatula2015leveraging.pdf

Dataset Details

The feature vectors for this paper have been provided at the following URL:

https://archive.ics.uci.edu/ml/datasets/URL+Reputation

Features

The list of attributes in a feature vector are:

- Having_IP_Address { -1,1 }
- URL_Length { 1,0,-1 }
- Shortining_Service { 1,-1 }
- Having_At_Symbol { 1,-1 }
- Double_slash_redirecting { -1,1 }
- Prefix_Suffix { -1,1 }
- Having_Sub_Domain { -1,0,1 }
- SSLfinal_State { -1,1,0 }
- Domain_registeration_length { -1,1 }
- Favicon { 1,-1 }
- Port { 1,-1 }
- HTTPS_token { -1,1 }
- Request_URL { 1,-1 }
- URL_of_Anchor { -1,0,1 }
- Links_in_tags { 1,-1,0 }
- SFH { -1,1,0 }
- Submitting_to_email { -1,1 }
- Abnormal_URL { -1,1 }
- Redirect { 0,1 }
- On_mouseover { 1,-1 }
- Right Click { 1,-1 }
- Pop-Up Window { 1,-1 }
- Attribute Iframe { 1,-1 }
- Attribute age_of_domain { -1,1 }
- Attribute DNSRecord { -1,1 }
- Attribute web_traffic { -1,0,1 }
- Attribute Page_Rank { -1,1 }
- Attribute Google_Index { 1,-1 }
- Attribute Links_pointing_to_page { 1,0,-1 }
- Attribute Statistical_report { -1,1 }
- Attribute Result { -1,1 }

Implementation

Implementation

We have implemented and trained

- Perceptron
- SVM

  to classify URLs as malicious(+1) or benign(-1).

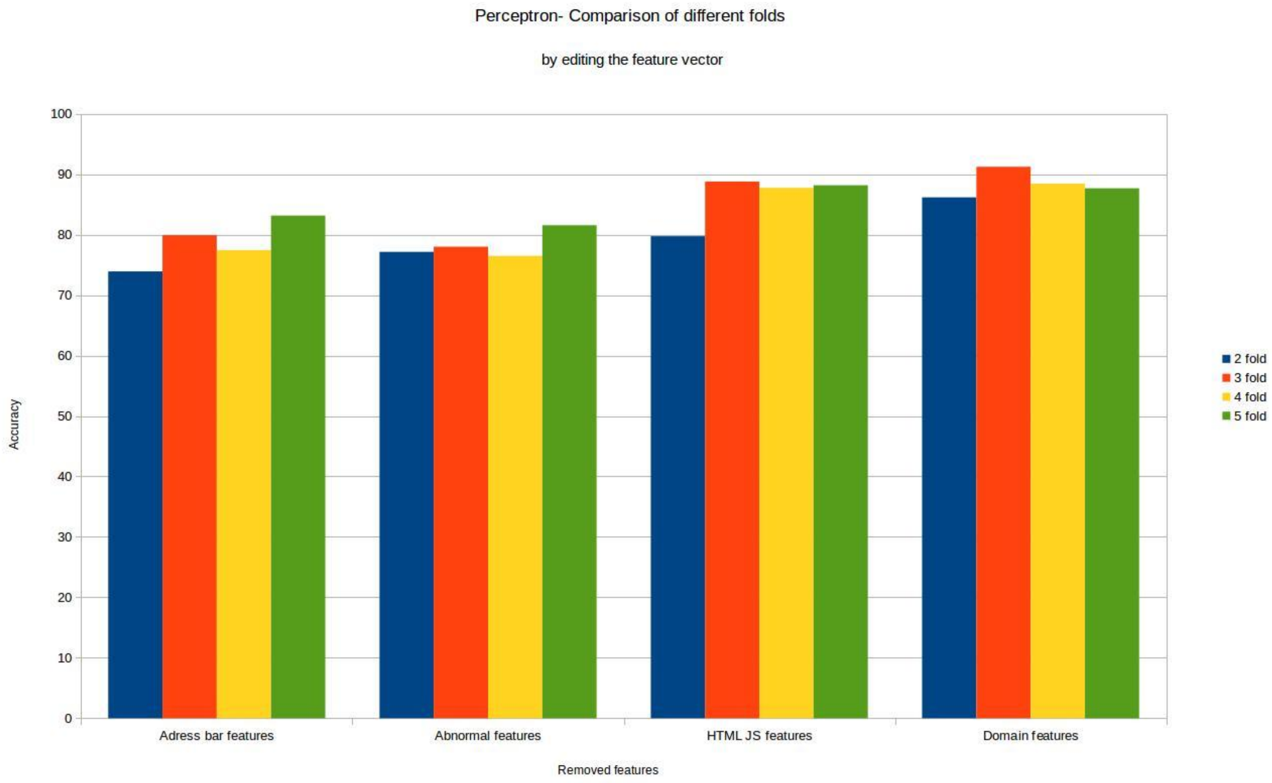
Implemented on two-fold, three-fold, four-fold and five-fold.

Experimented with the feature vectors by removing/keeping a combination of the following classes:

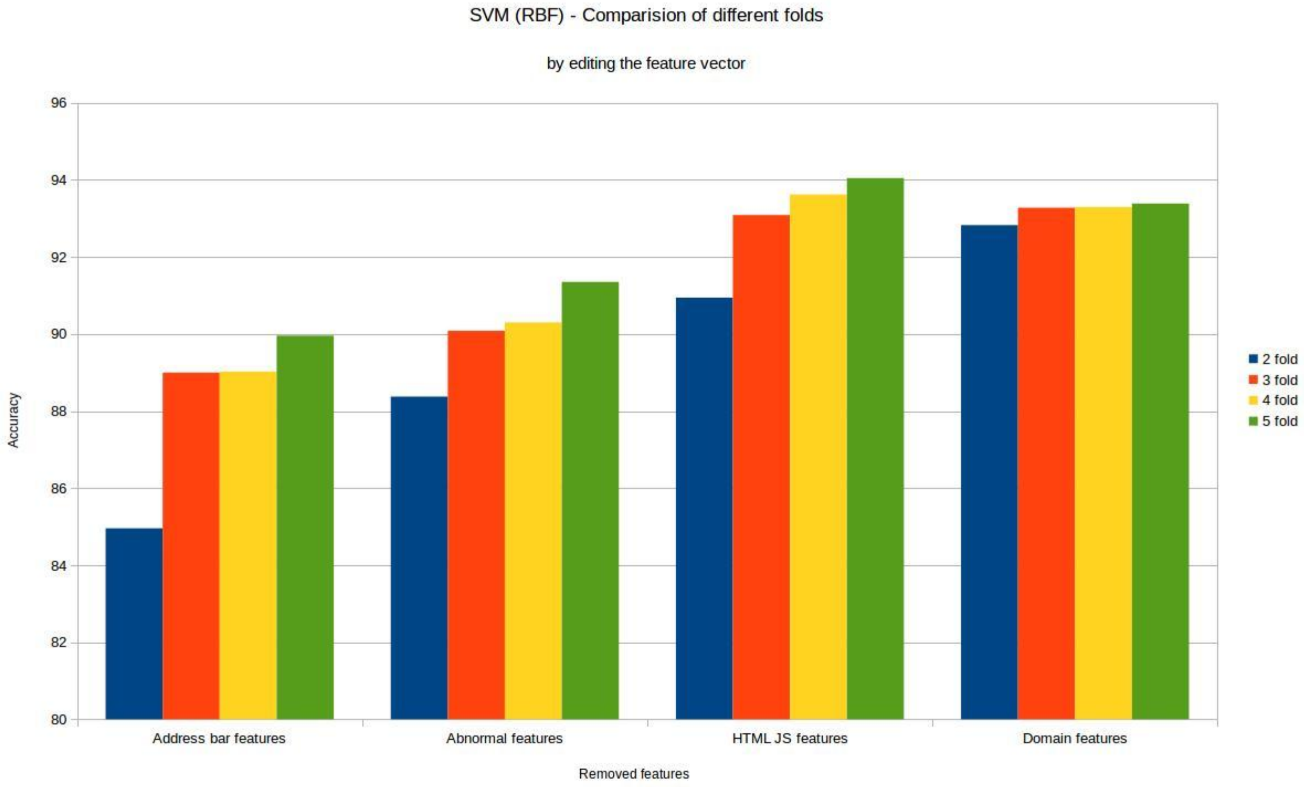
- Address bar features
- Abnormal features
- HTML and Javascript features
- Domain-based features

The respective accuracies of different folds have been analysed using the following graphs

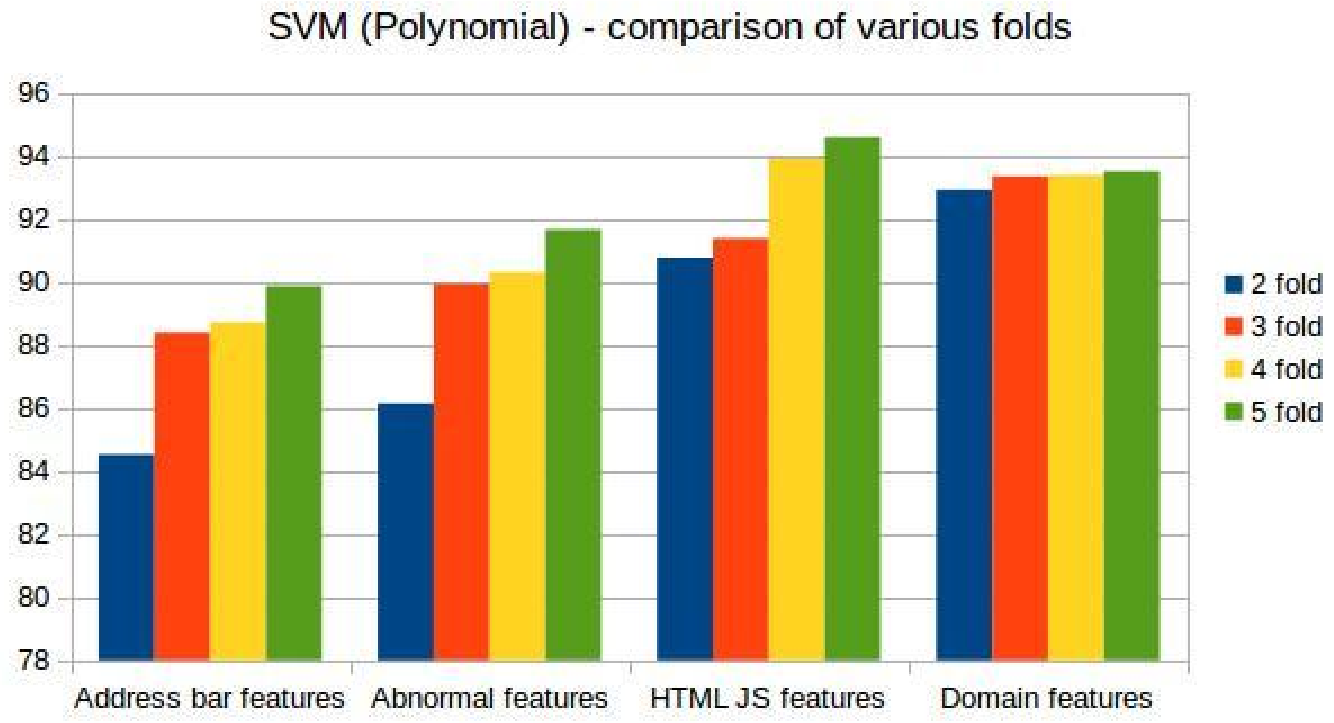
Graphs: K-fold on Perceptron



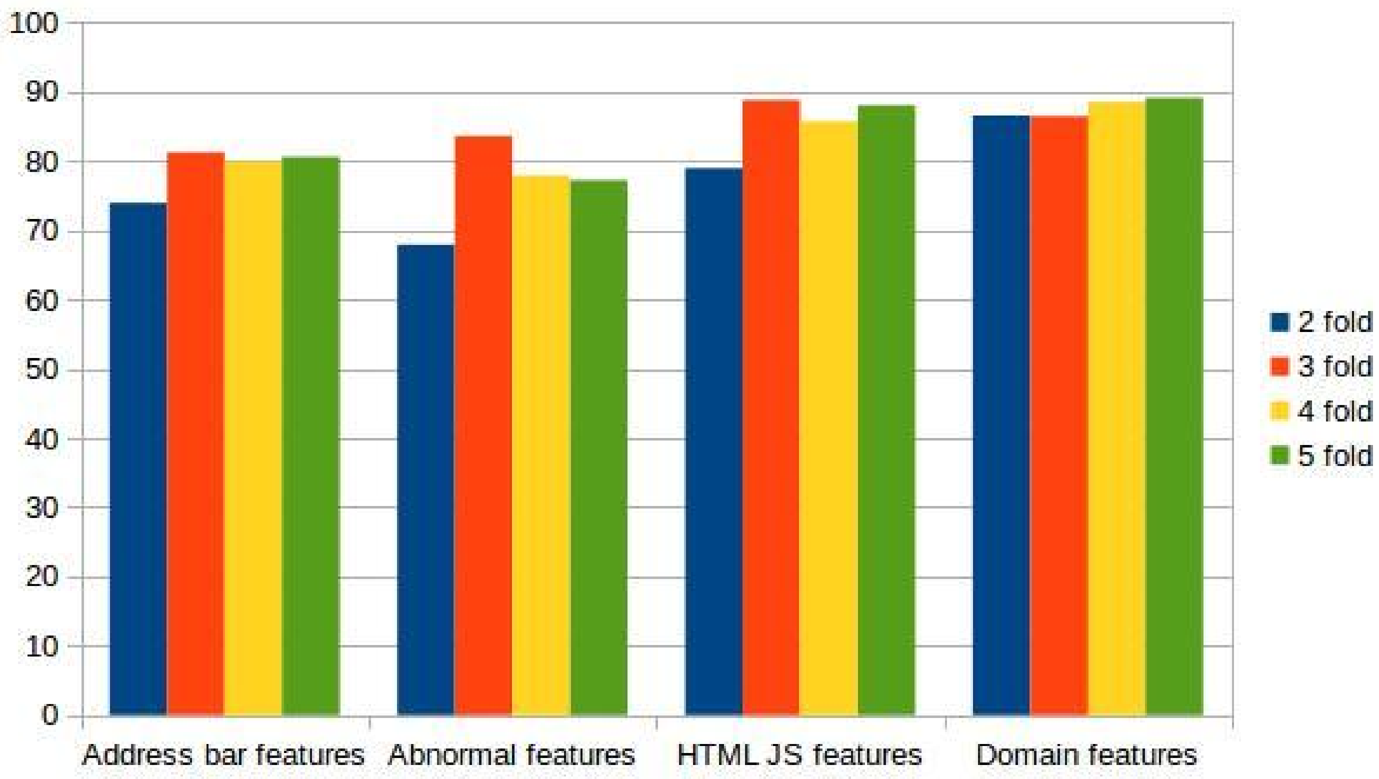
Graphs: K-fold on SVM(RBF)



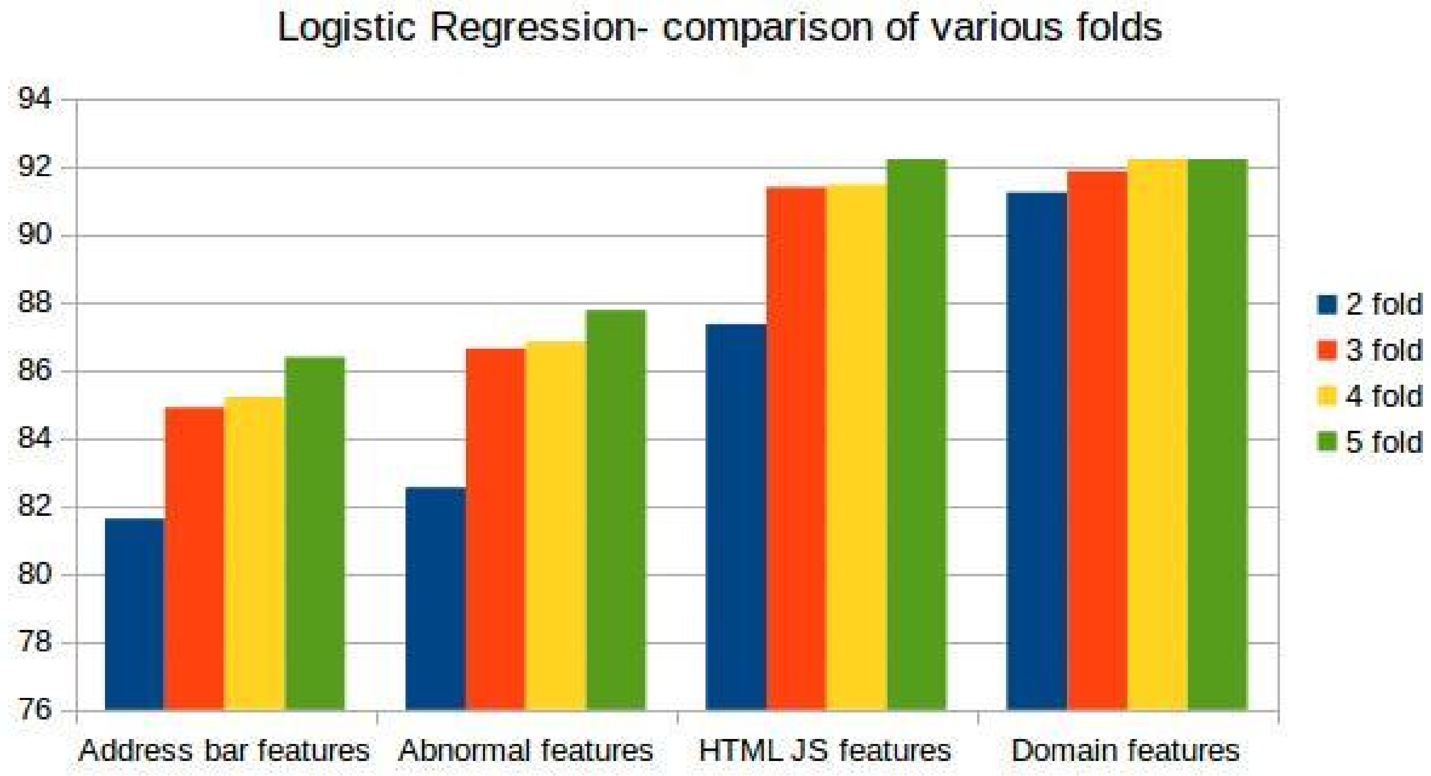
Graphs: K-fold on SVM(Polynomial)



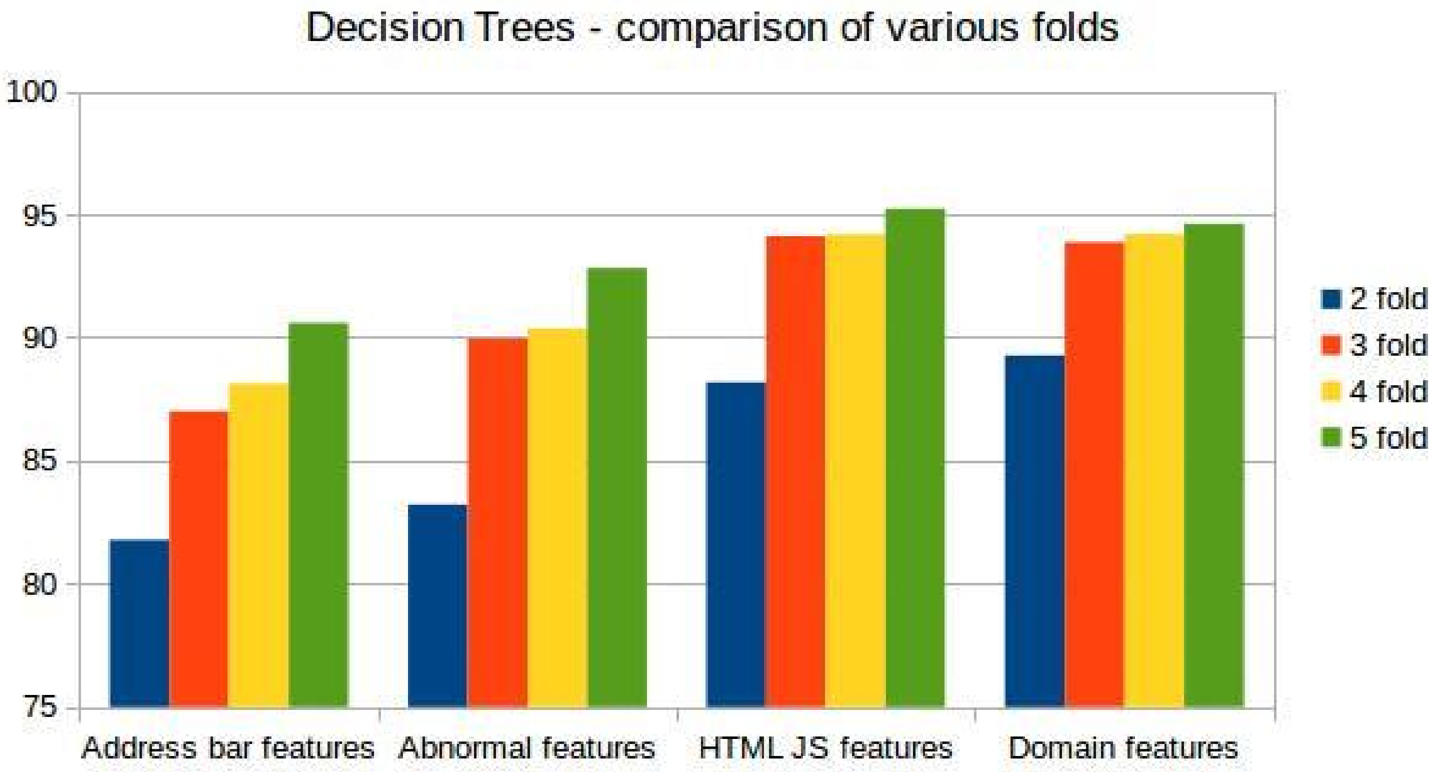
Graphs: K-fold on Passive-Aggressive



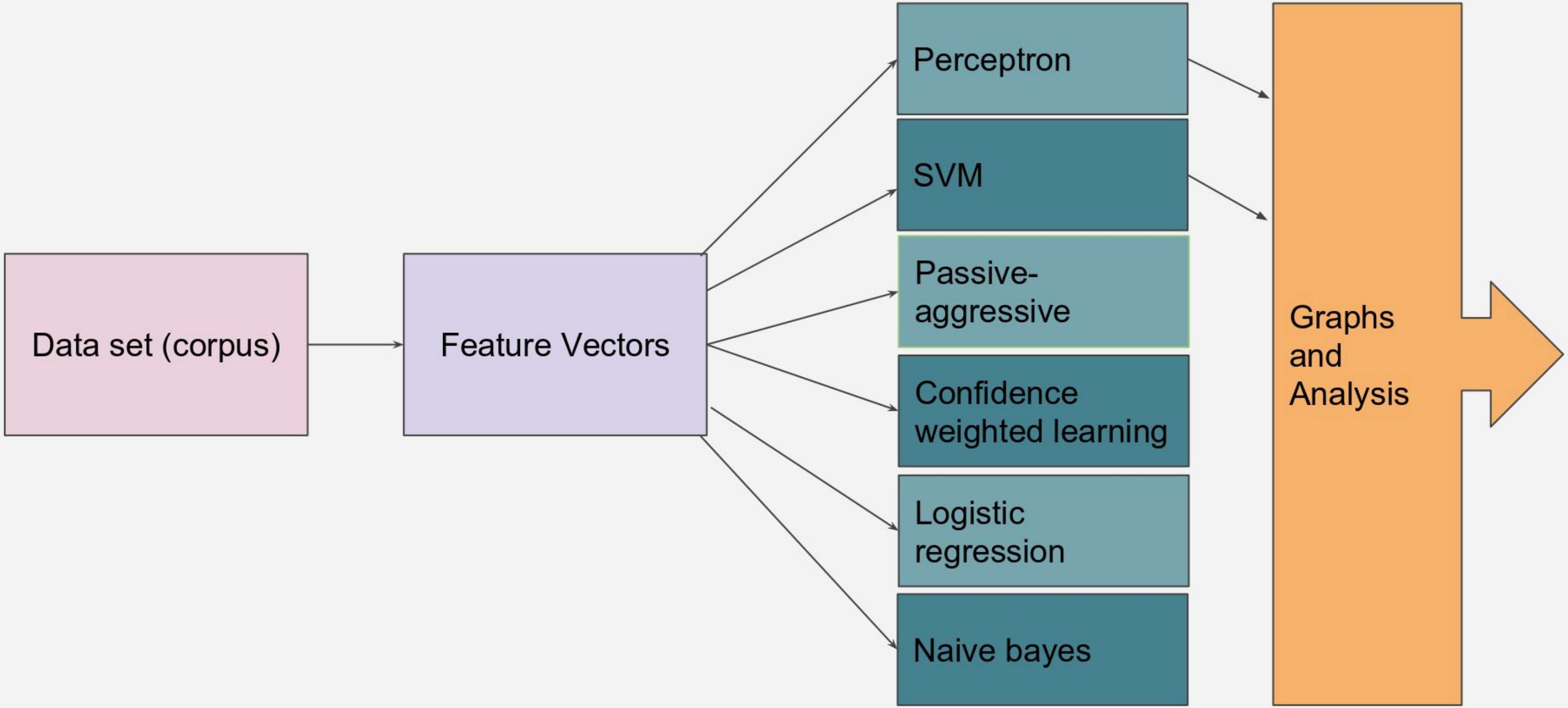
Graphs: K-fold on Logistic Regression



Graphs: K-fold on Decision-Trees



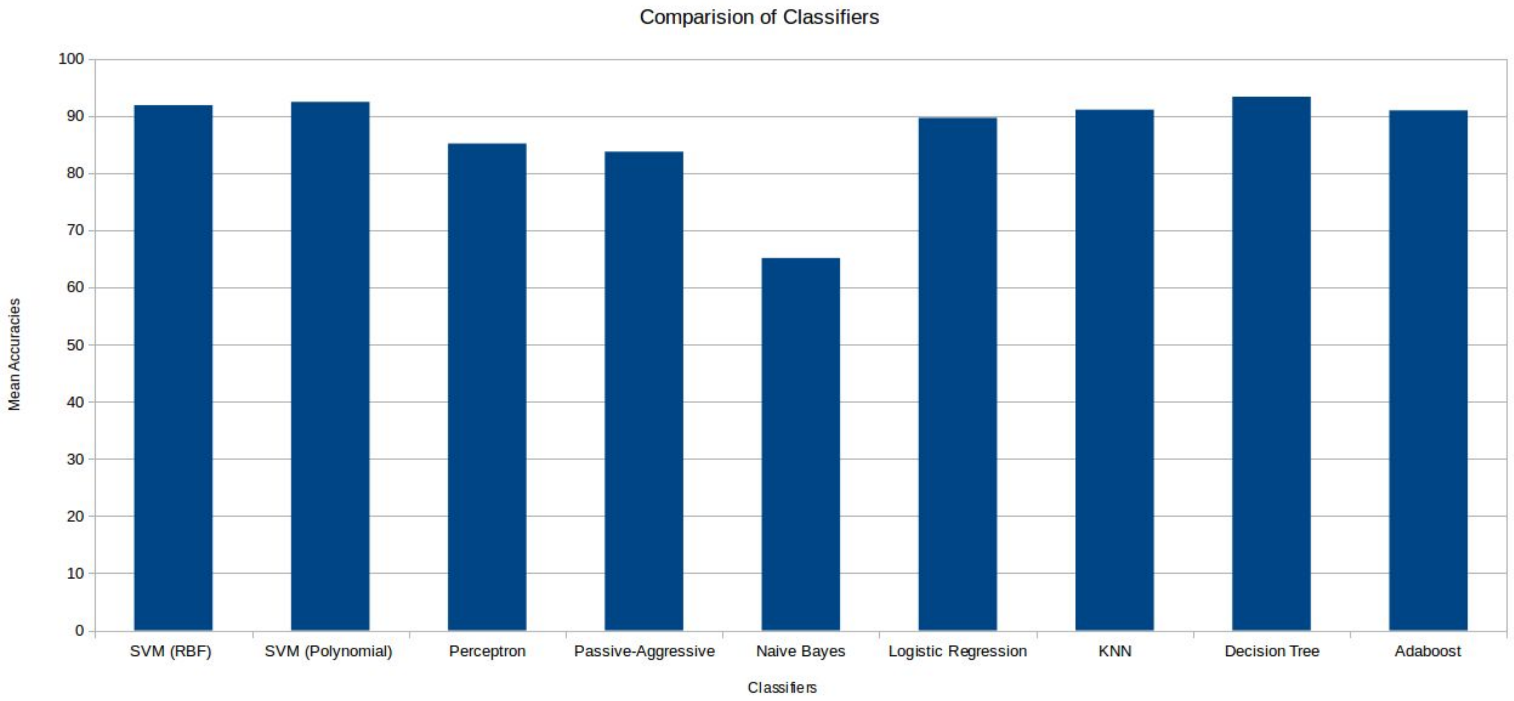
The Workflow



Analysis

Comparative Analysis

The graph for the comparative performances of the classifiers implemented is given



Analysis

- The performance for SVM RBF and Polynomial kernels are comparable (92.18605 and 92.40316).
- Perceptron and Passive Aggressive have similar performances(86.448377 and 88.791333).
- Gaussian- Naive Bayes gives poorest accuracy (65.112043).
- Decision tree is giving the highest accuracy (93.816575).

Analysis

- Classification accuracy suffered maximum dip when Address bar features were removed. This was followed by the classification with Abnormal feature removal. This was consistent across all classifiers.
- HTML and Javascript features and Domain based features followed next with both their contributions varying across different classifiers.

Thank You

Lasya Venneti 201356157

Shreekavitha P. 201356194

Kaveri Anuranjana 201325199