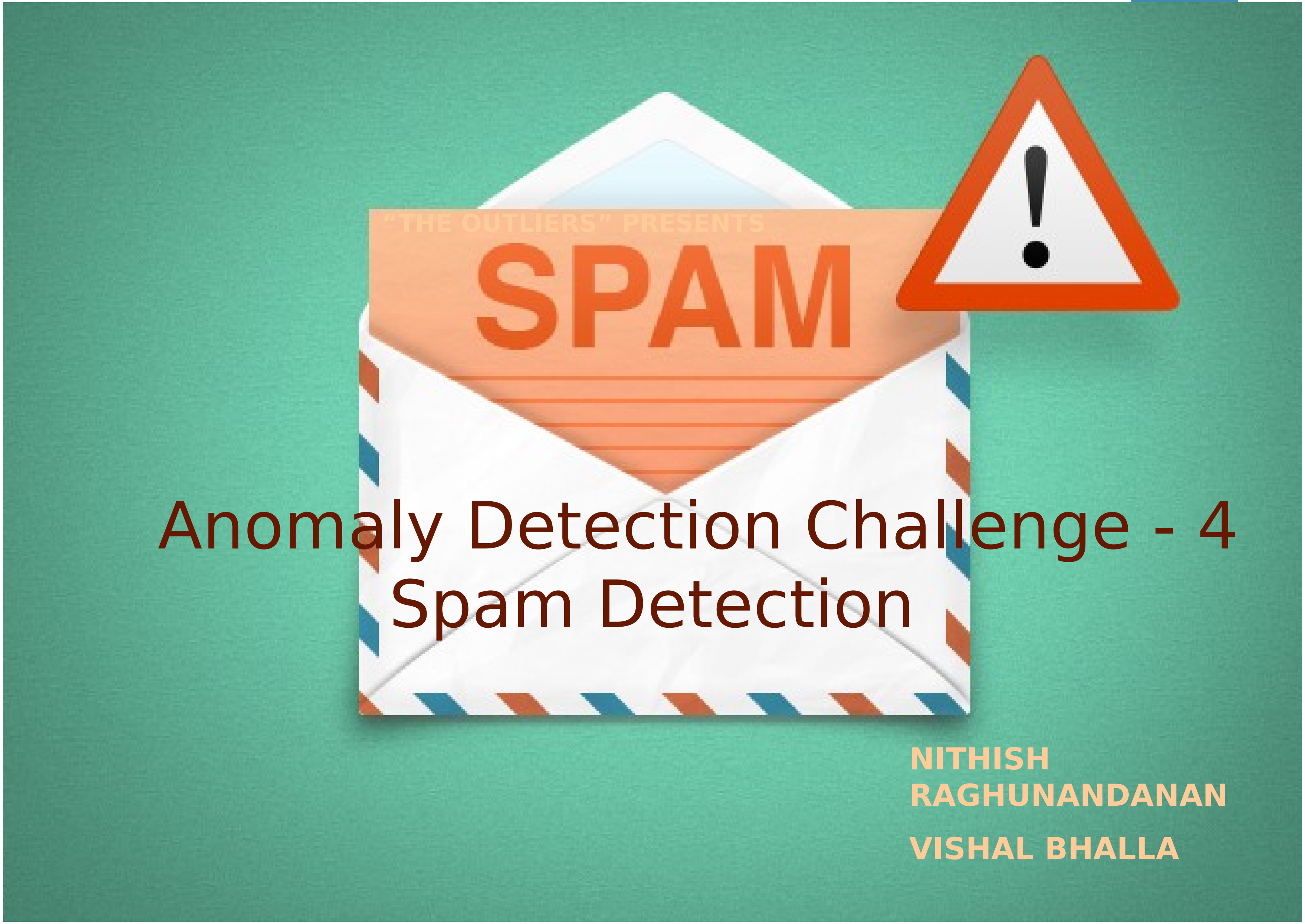


“THE OUTLIERS” PRESENTS

Anomaly Detection Challenge - 4
Spam Detection

NITHISH RAGHUNANDANAN

VISHAL BHALLA

Picture courtesy Volusion

Agenda

- Introduction
- System Pipeline
- Data Preprocessing & Analysis
- Feature Engineering
- Static Rules
- Models
- Kaggle Results
- Conclusion
- Key Takeaways

Introduction

- Aim: Classify mail into Spam (0) or Ham (1).
- Problem Type: Binary Classification
- Samples:
  Dataset is based on the CSDMC2010 SPAM corpus.
  - Training Set: 2500
  - Test Set: 1827
- Features: Emails in RFC822 format.
- Classification
  - There are 2 decision classes: 0(Spam) and 1(Ham).

System Pipeline

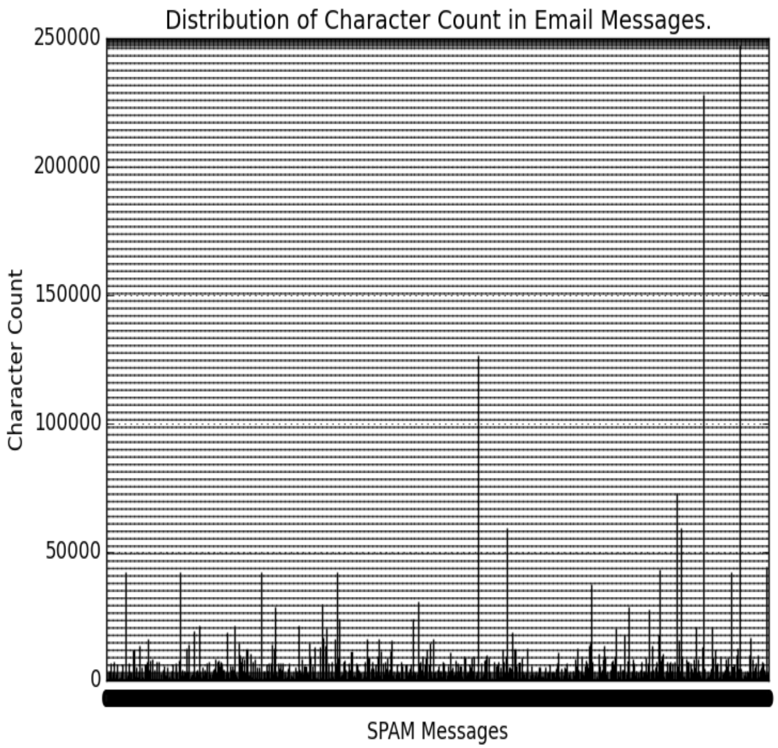
Data Visualization → Feature Selection → Evaluation of Models

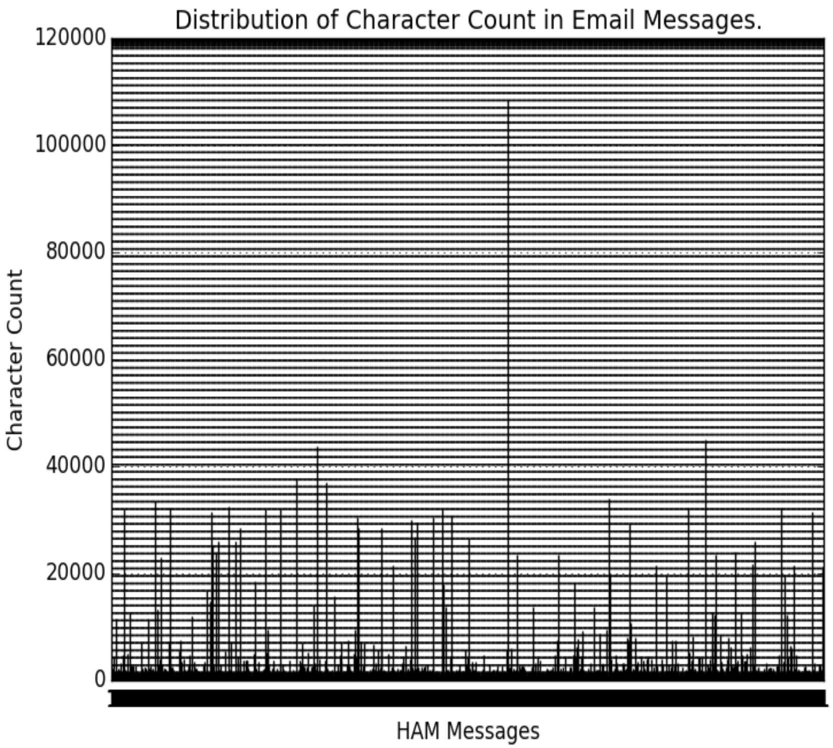
- Data Visualization
  - Tabulate & analyse the structure of the data.
- Feature Selection
  - Encoding selected parts of the data as features.
- Evaluation of Models
  - Classification accuracy using K Fold Stratified Cross Validation.

Data Analysis

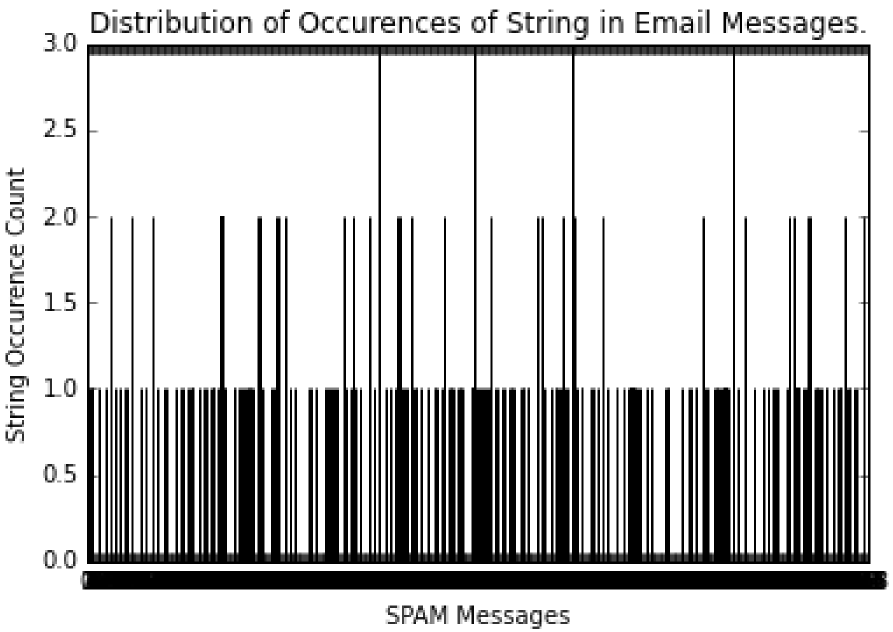
- Email Fields
  - Subject - Length, Content & Presence of some characteristic Spam words in the Email Subject
  - Body - Length, Content of Email Body
  - PGP Signatures - Presence/Absence of PGP signatures in the Email Body
  - SpamAssassin Classification - Spam mails classified as [spam] in Email subject
  - Date - Date of the Email
  - Sender - Presence of Sender field in the Email header

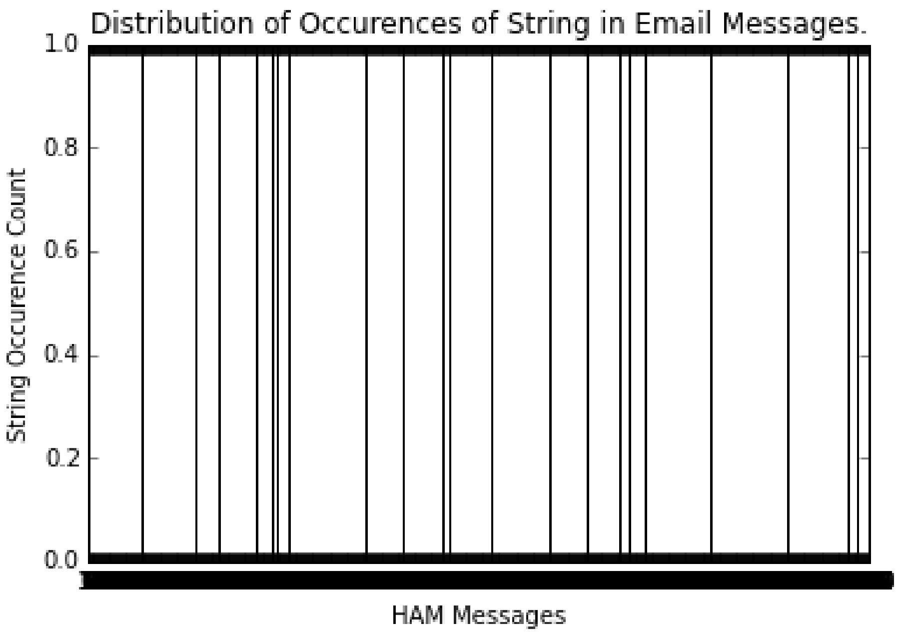
Distribution of Character counts in Email Message Body



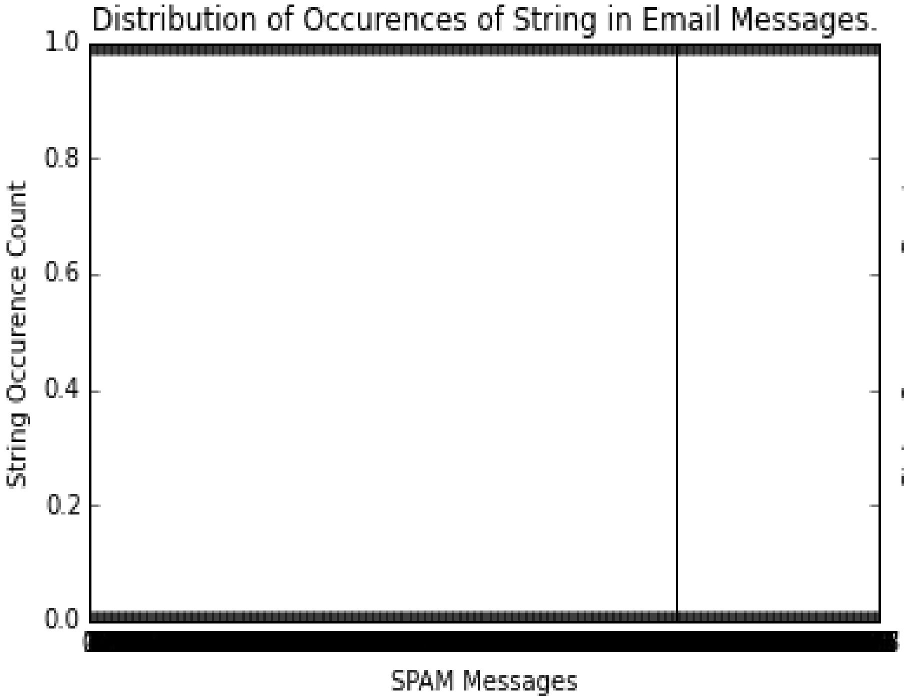


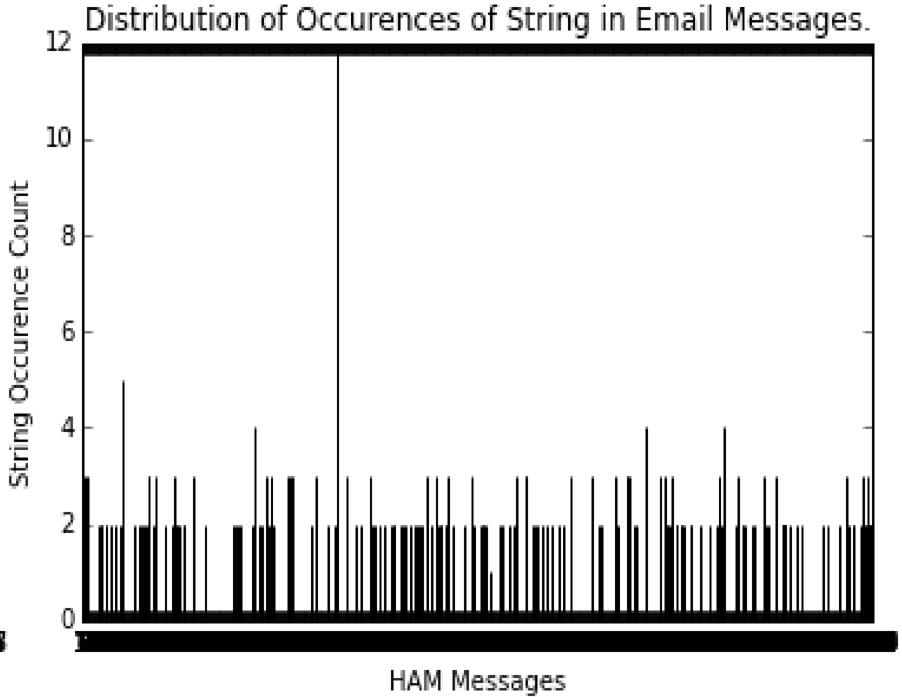
Distribution of Capital letter words in Email Message Body





Presence/Absence of PGP Signature in Email Message Body





Feature Engineering

- Spaminess/Haminess of Email Content [1]
  - ✓ Spamminess(Word) = Number of occurrences of word in Spam emails / Total Occurrences of word in Emails
  - ✓ Spamminess(Message) = Product of Spamminess(Words) in the Email

- Spaminess/Haminess of Top Spam/Ham Words
  - ✓ Spaminess & Haminess of extreme K words in terms of spaminess/haminess considered.

Feature Engineering(2)

- Spaminess of Top K Spam Words in Emails
  - ✓ Spaminess of extreme K words in terms of spaminess/haminess considered.

- Ratio of Spam words
  - ✓ Ratio of Spam words to total words

Static Rules

- Rule 1
  - ✓ SpamAssassin Classification in Email Subject.
- Rule 2
  - ✓ Presence of PGP Signature in the Email Body.
- Rule 3
  - ✓ Date of Email in extreme future/past.
- Rule 4
  - ✓ Presence of Sender information.
- Rule 5
  - ✓ Too many capitalized words in the Email Body.

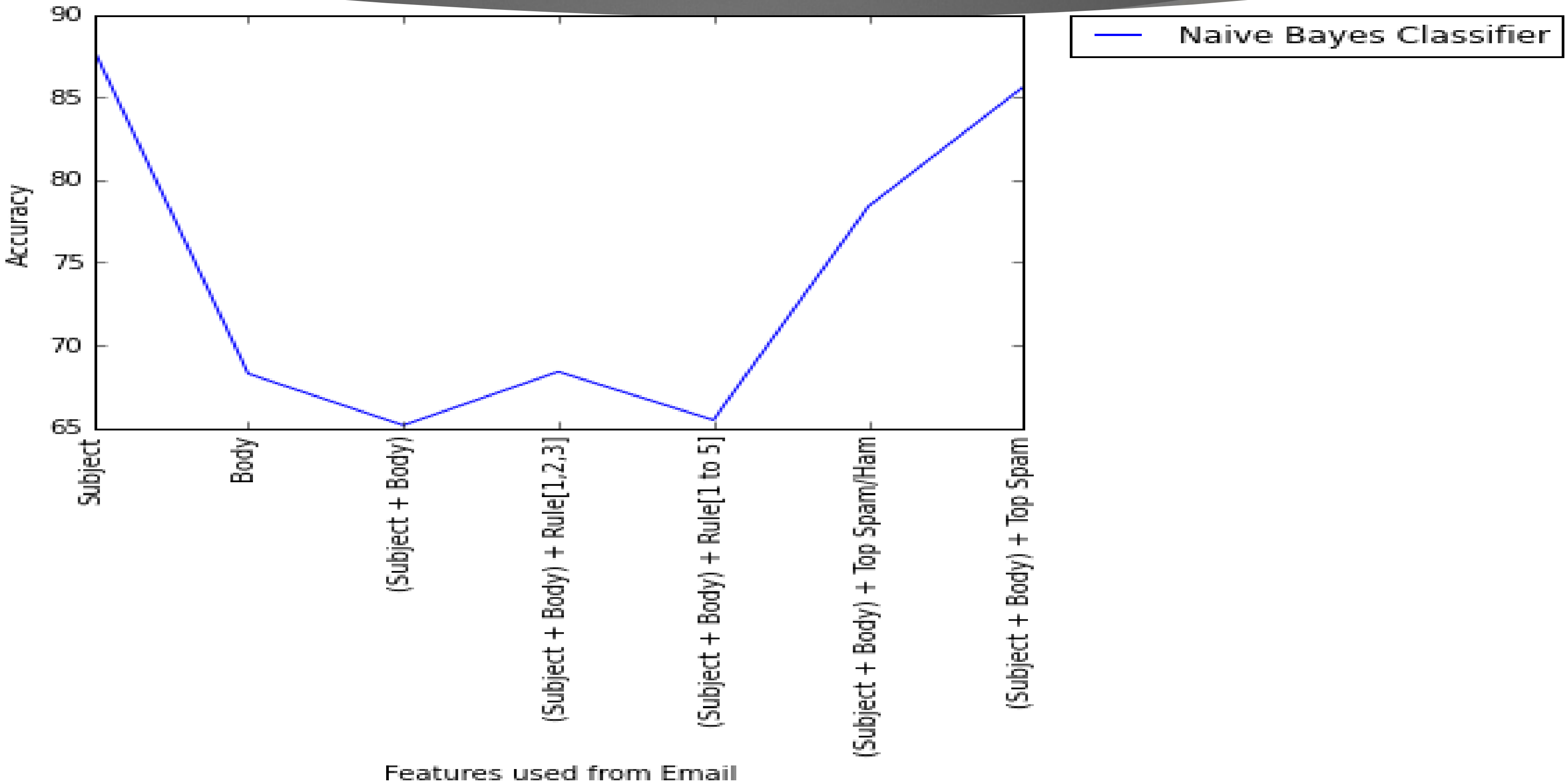
Models

- We tried different types of Binary Classification Models to fit our data
  - Naive Bayes
  - K Nearest Neighbors (kNN)
  - Random Forests (RF)
  - Adaboost
  - Vowpal Wabbit (VW)

Naive Bayes

- Classification Criteria
  - ✓ Spaminess > Haminess
  - ✓ Top K Spam/Ham
  - ✓ Top K Spam > Threshold

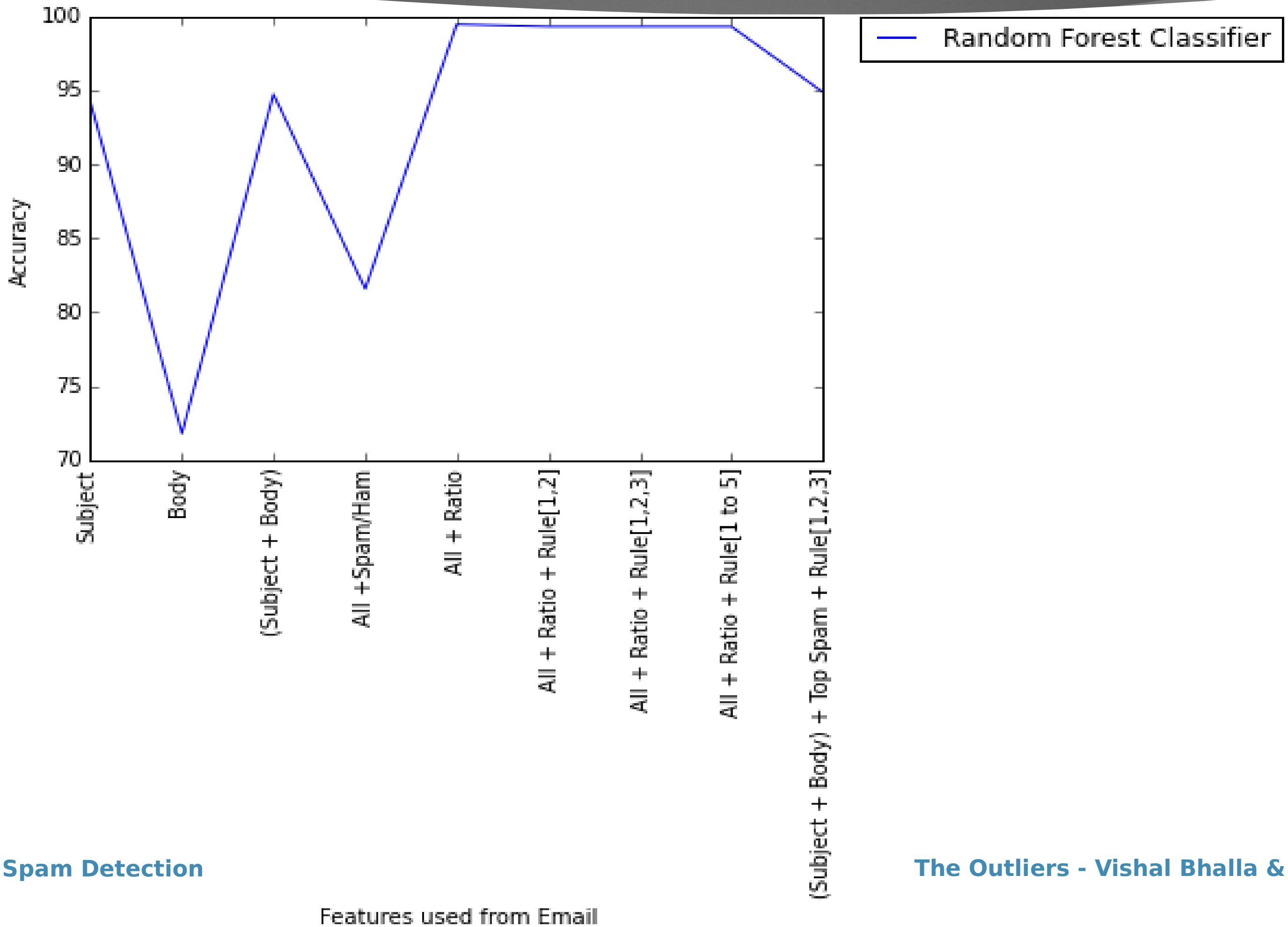
Naive Bayes(2)



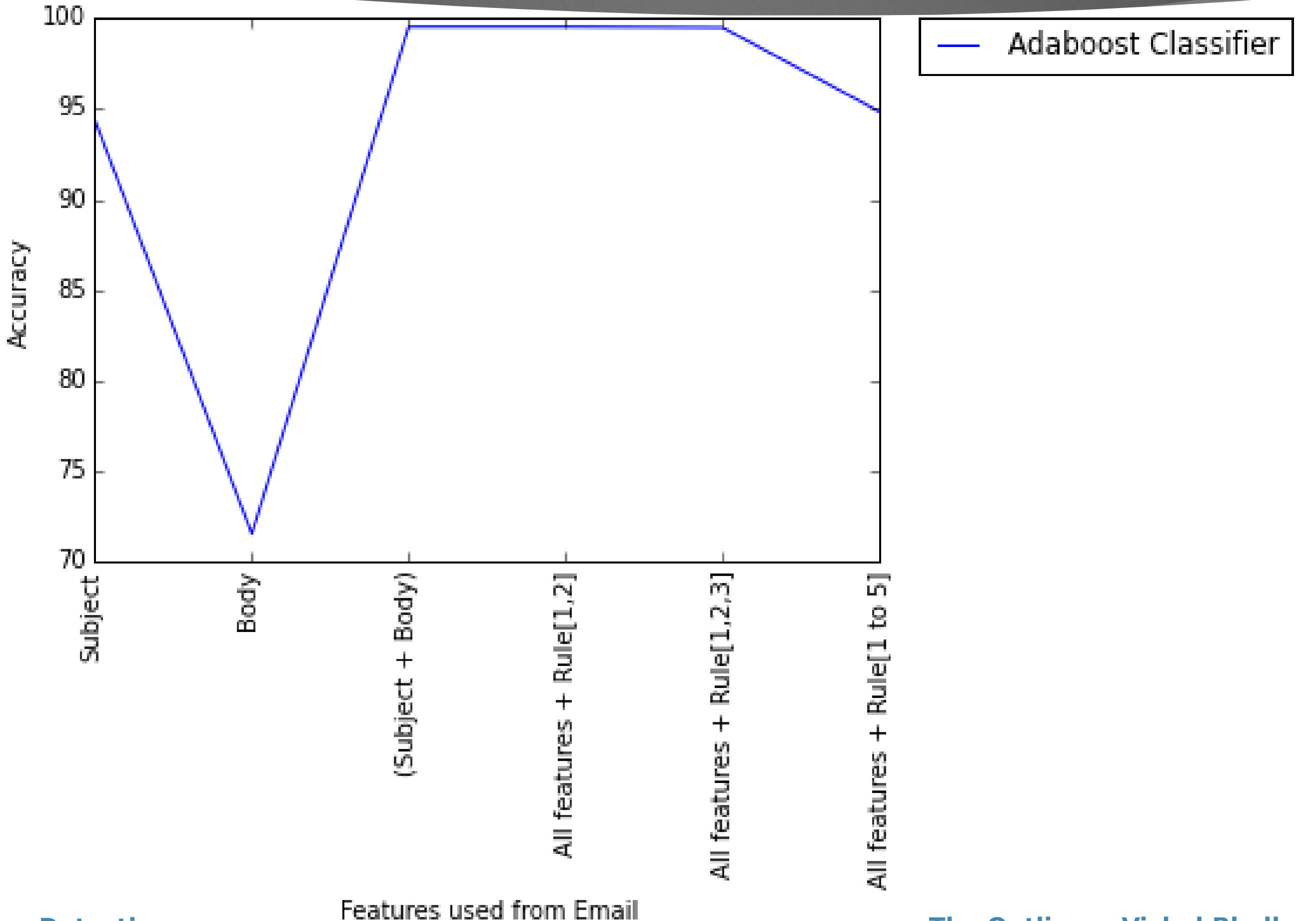
Random Forests

- Email fields used as feature vector
  - Subject
  - Body
  - Combined (Subject and Body)
  - All features (Combined + Special words, capital letter words & PGP) + Spaminess > Haminess metric
  - All features + Ratio of Spam words metric
  - All features + Ratio of Spam words metric + Static Rules
  - All features + Top K Spam > Threshold metric

Random Forests



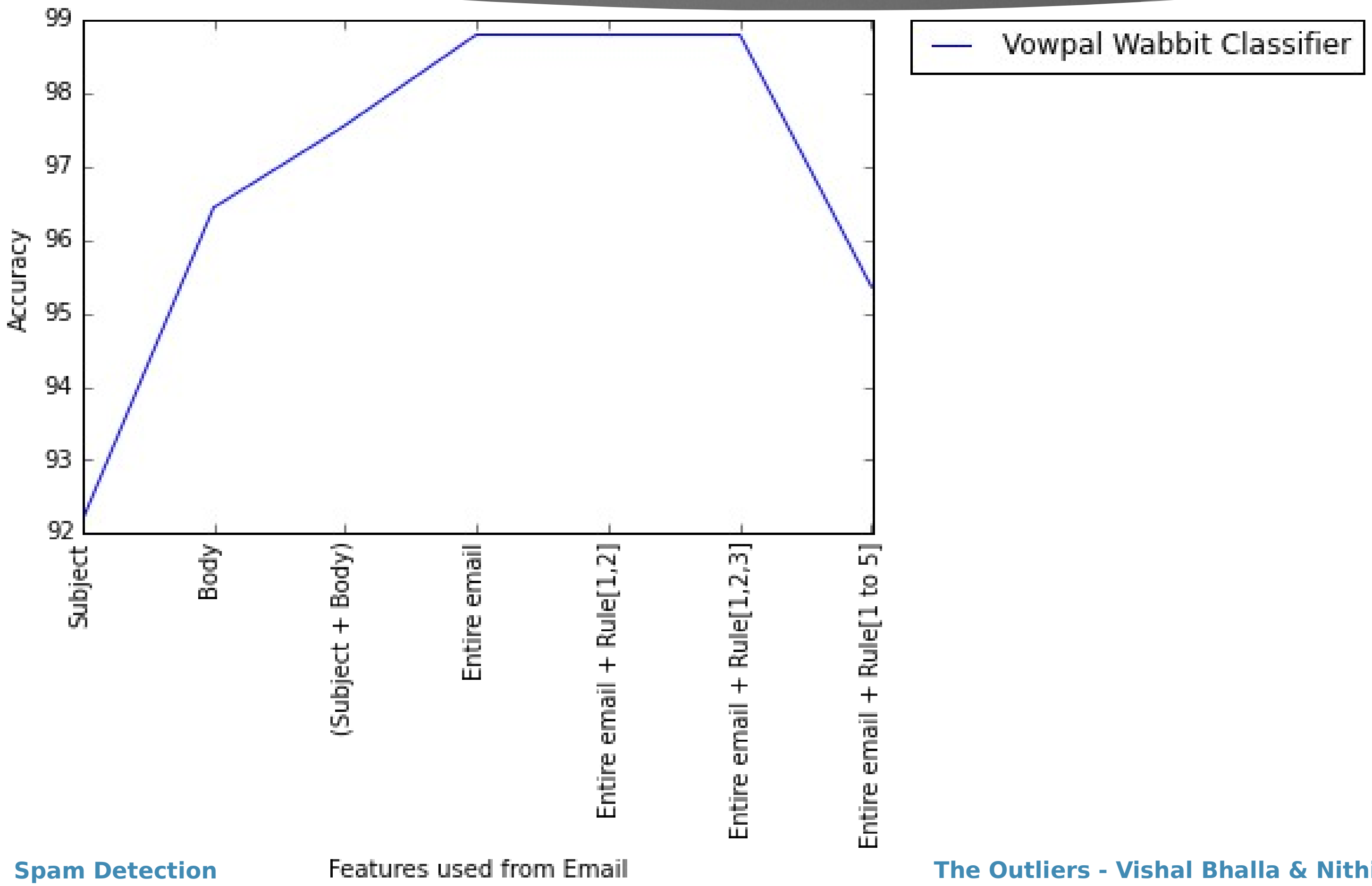
Adaboost


Adaboost Classifier
Accuracy
100
95
90
85
80
75
70
Subject
Body
(Subject + Body)
All features + Rule[1,2]
All features + Rule[1,2,3]
All features + Rule[1 to 5]
Features used from Email

Vowpal Wabbit

- Fast, efficient library for Online Machine Learning.
- Program developed originally at Yahoo! Research, and currently at Microsoft Research [2].
- Adaptive Learning with minimization of loss.
- Features
  - ✓ Email Subjects
  - ✓ Email Body
  - ✓ Email Subjects + Body
  - ✓ Email as a string with headers

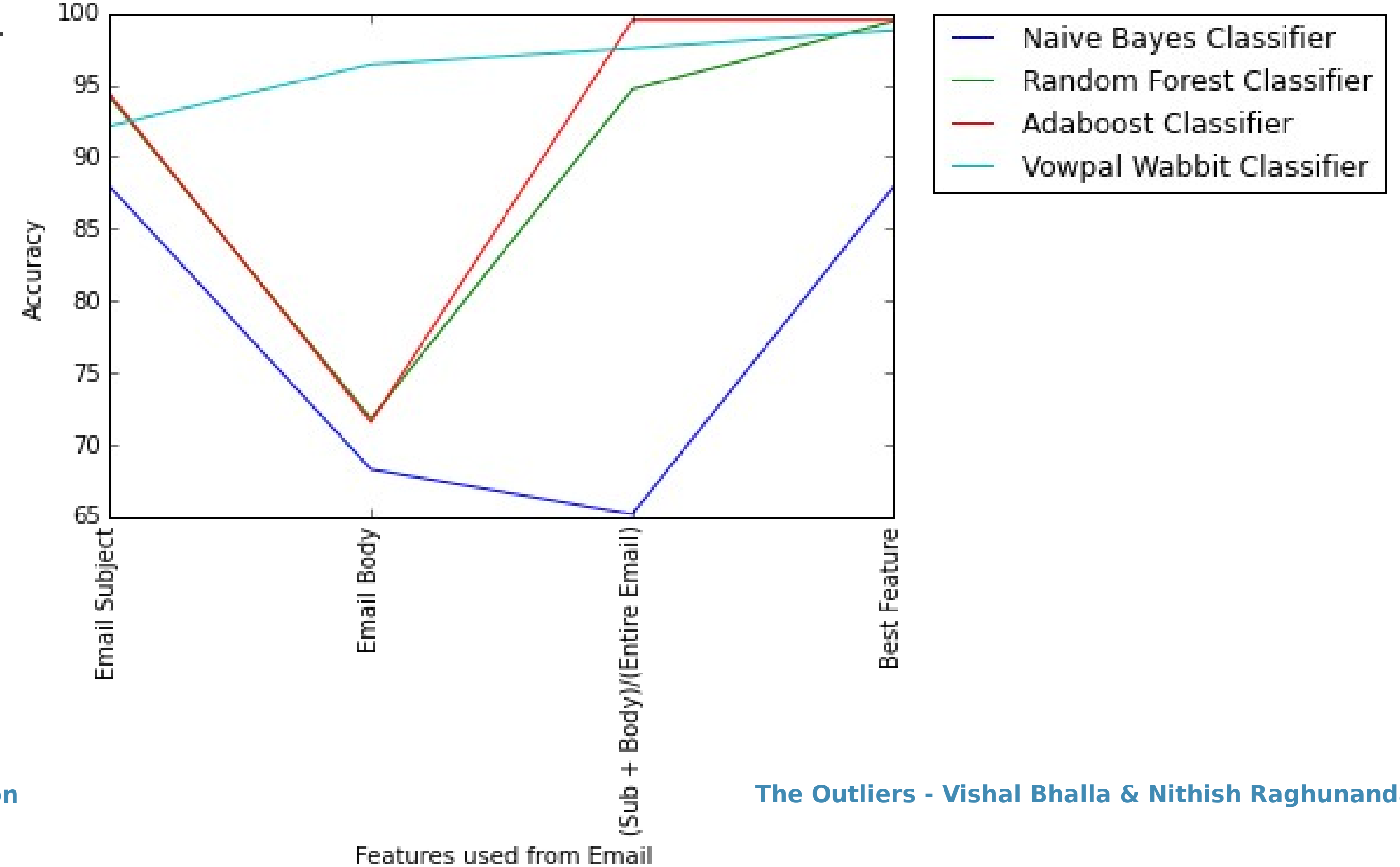
Vowpal Wabbit(2)



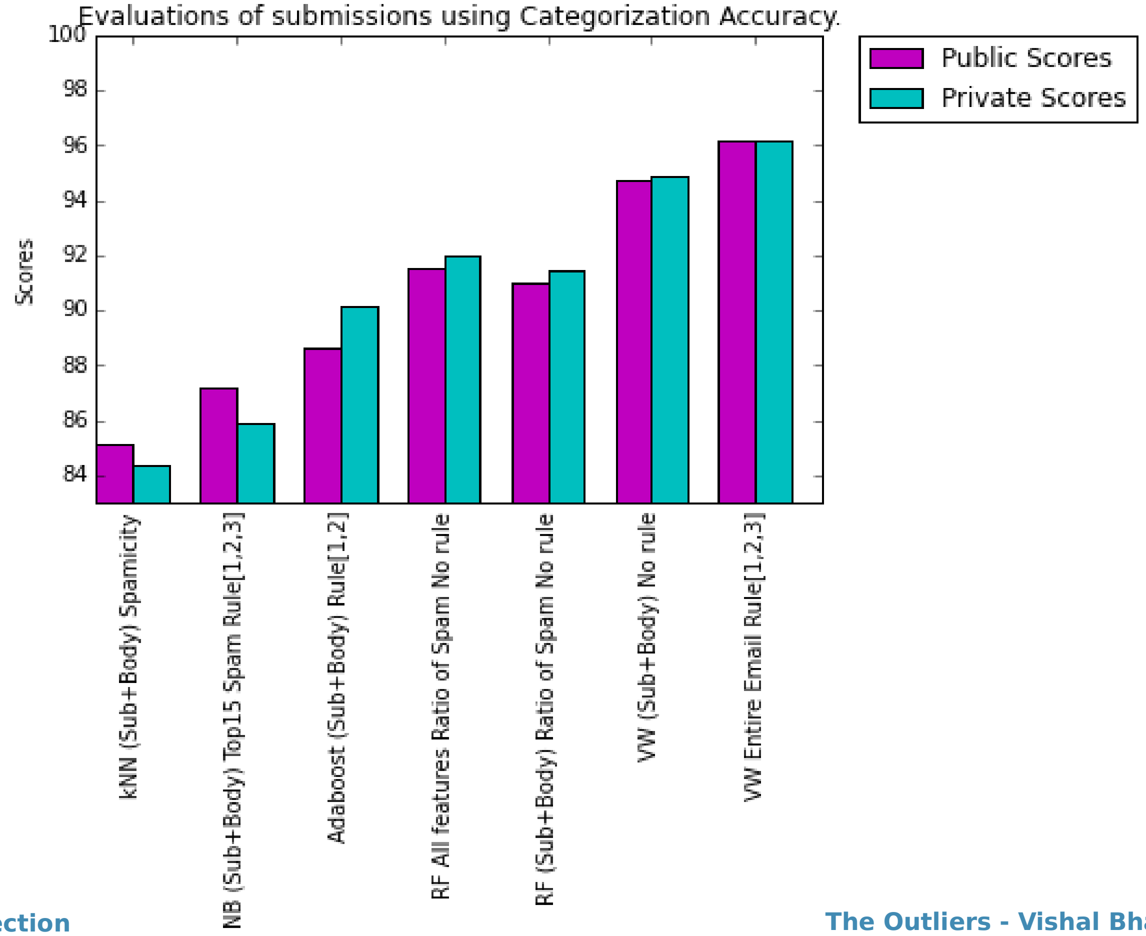
Evaluation of the Models

Criteria

Classification Accuracy on Stratified K-Fold Cross Validation with a split of 4:1.



Kaggle Results



Conclusion

- The best results were observed for the classification by Vowpal Wabbit.
- Adding all information as features for the classifier helps.
- Naive Bayes depends on the calculation of Spamminess/ Haminess parameters.
- Not all static rules are important!
- Rules 1 [SpamAssasinator], Rule 2 [PGP Signature] & Rule 3 [Date] were the most useful.

Key Takeaways

- Vowpal Wabbit is useful in dealing with raw Strings & is extremely fast.
- Better Metric for computing Spamminess or Spammicity of words.

References

- [1] Metric for computing Spamminess or Spammicity of words
  Awad, W. A., and S. M. ELseuofi. "Machine Learning methods for E-mail Classification." International Journal of Computer Applications (0975–8887) 16.1 (2011).

- [2] Vowpal Wabbit

  Langford, John, L. Li, and A. Strehl. "Vowpal wabbit." URL https://github. com/JohnLangford/vowpal_wabbit/wiki (2011).

Questions?

Thank You !