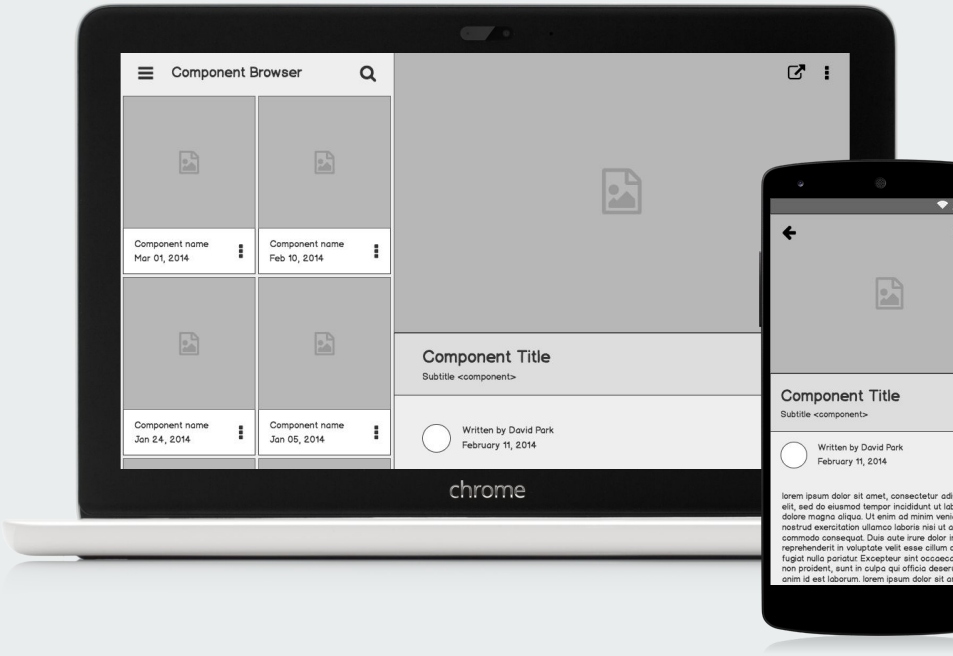
Web Scraping

Aakash Sharma
DSI Program



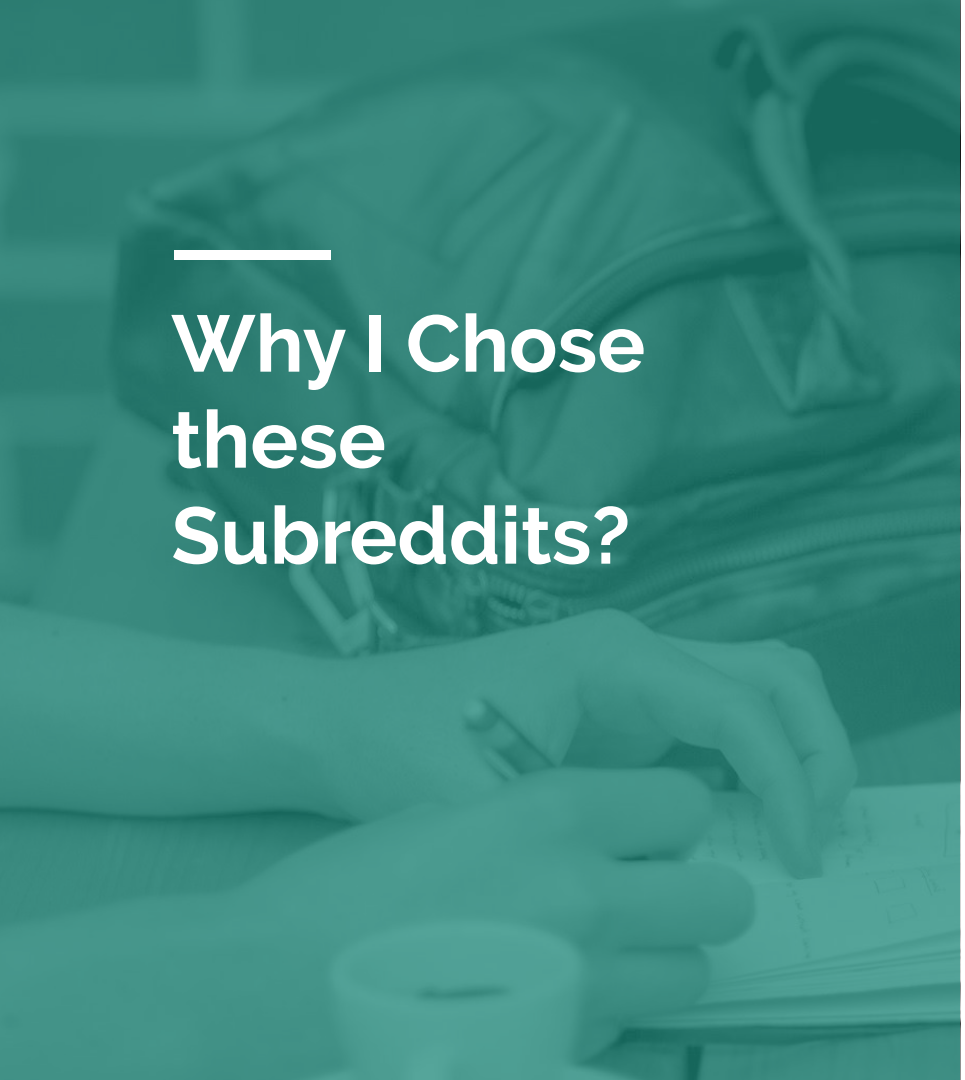
Reddit Itinerary

1. The Problem
2. Project Approach
3. The Model
4. Diagrams
5. Executive Summary

The Problem

Use Reddit's API & collect posts from two subreddits.

Use NLP to train a classifier on which subreddit a given post came from. This is a binary classification problem.

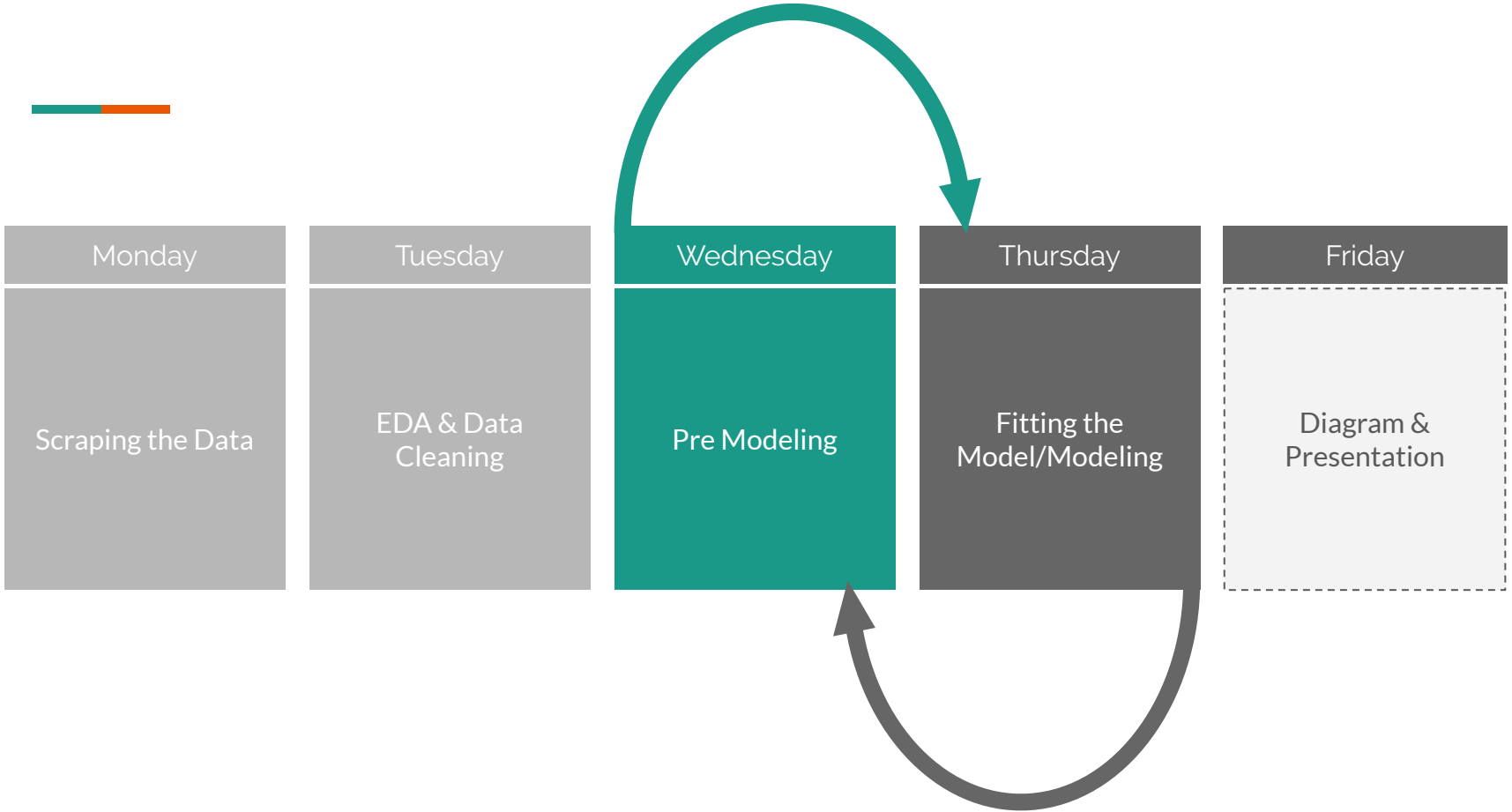


Why I Chose these Subreddits?

Hacking & Cyber Security Subreddits

- Focus around what I specialized in
- Interested me
- Cool project to talk about?

My Approach


Monday
Tuesday
Wednesday
Thursday
Friday
Scraping the Data
EDA & Data Cleaning
Pre Modeling
Fitting the Model/Modeling
Diagram & Presentation

Data Scraping the API's

```
https://www.reddit.com/r/hacking/top.json?t=all
https://www.reddit.com/r/hacking/top.json?t=all&after=t3_85mkmm
https://www.reddit.com/r/hacking/top.json?t=all&after=t3_axaz6t
https://www.reddit.com/r/hacking/top.json?t=all&after=t3_64junt
https://www.reddit.com/r/hacking/top.json?t=all&after=t3_7s5j11
https://www.reddit.com/r/hacking/top.json?t=all&after=t3_51rwd3
https://www.reddit.com/r/hacking/top.json?t=all&after=t3_6bilzs
https://www.reddit.com/r/hacking/top.json?t=all&after=t3_b30psu
https://www.reddit.com/r/hacking/top.json?t=all&after=t3_7cl3c5
```

```
https://www.reddit.com/r/cybersecurity/top.json?t=all
https://www.reddit.com/r/cybersecurity/top.json?t=all&after=t3_aeiwwo
https://www.reddit.com/r/cybersecurity/top.json?t=all&after=t3_azy3bn
https://www.reddit.com/r/cybersecurity/top.json?t=all&after=t3_cb61hr
https://www.reddit.com/r/cybersecurity/top.json?t=all&after=t3_b3srdc
https://www.reddit.com/r/cybersecurity/top.json?t=all&after=t3_azcty3
https://www.reddit.com/r/cybersecurity/top.json?t=all&after=t3_b9c3j2
https://www.reddit.com/r/cybersecurity/top.json?t=all&after=t3_bn1i08
https://www.reddit.com/r/cybersecurity/top.json?t=all&after=t3_aso2iy
https://www.reddit.com/r/cybersecurity/top.json?t=all&after=t3_7yu87h
```

What I Did & What it Does

Regex: Extracting info from any text by searching for one or more matches of a specific search pattern.

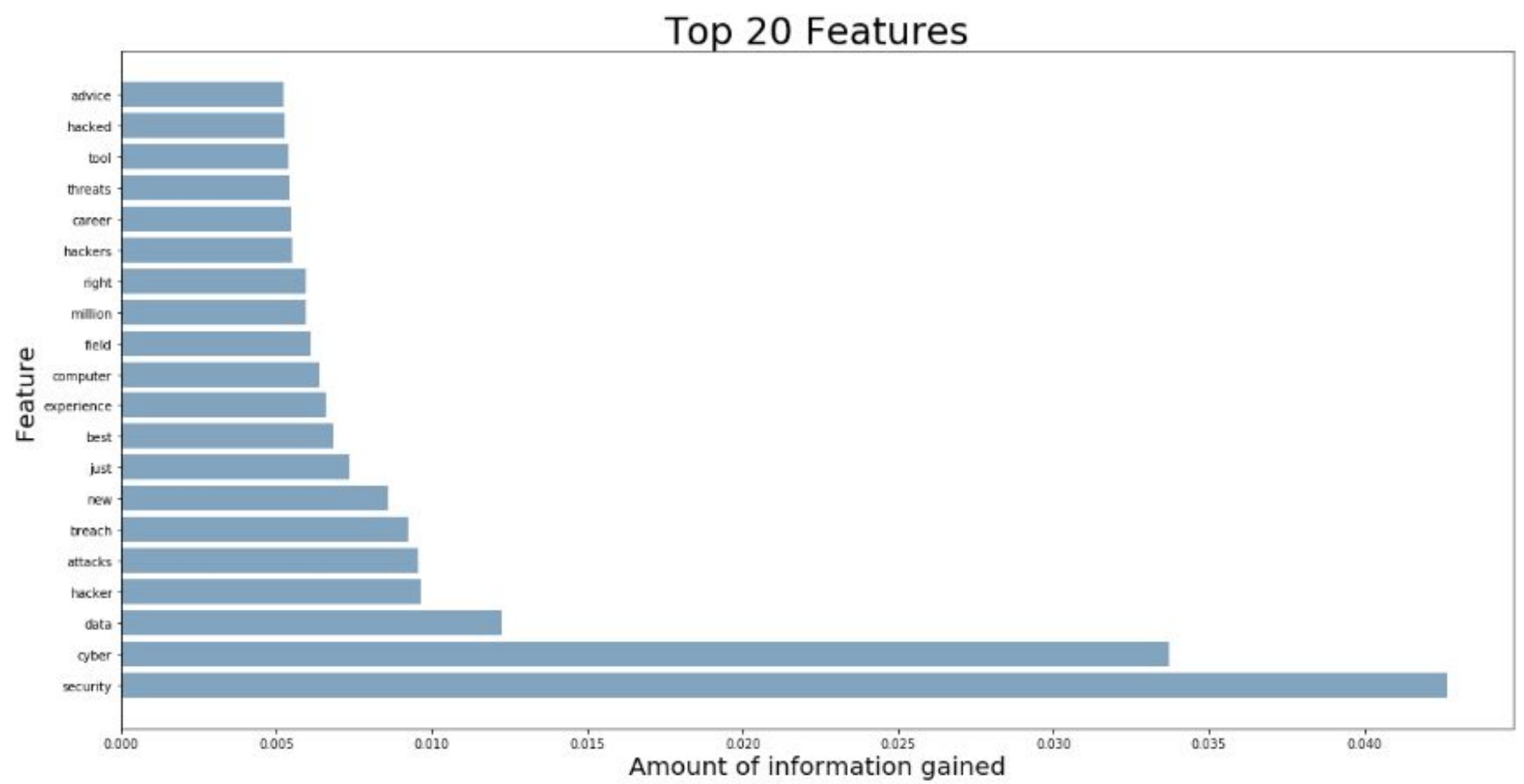
TFIDF: Term frequency/inverse document frequency, is a numerical statistic that shows how important a word is to a document.

Grid Search: Grid search selects the best of a family of models, parametrized by a grid of parameters.

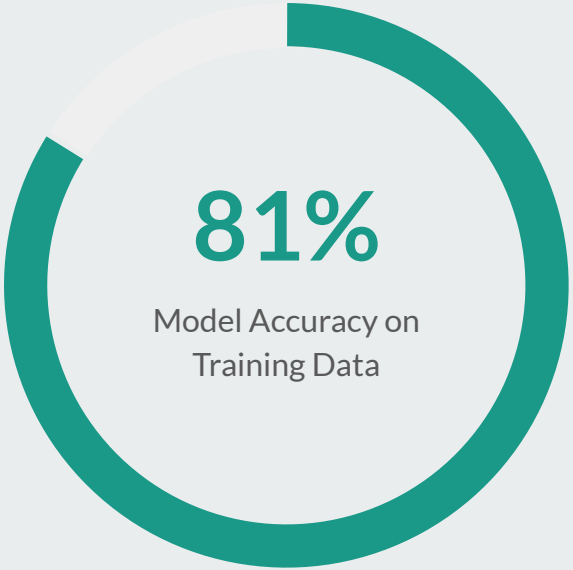
Naive Bayes: Simple, fast, accurate & reliable in calculating the probability of each tag for a given text & output of the tag with the highest one in NLP.

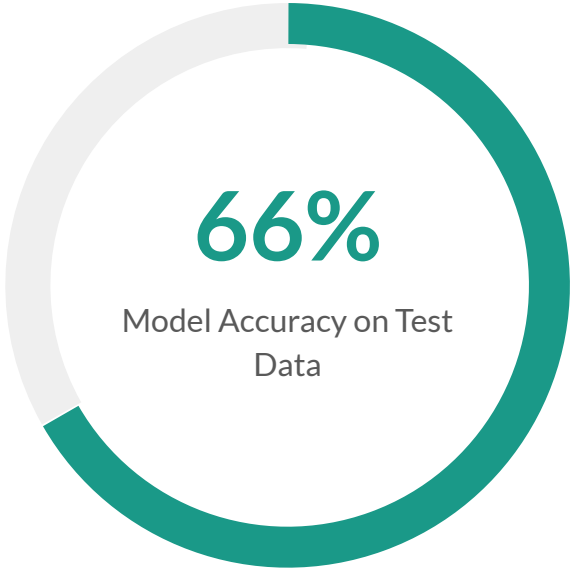
Random Forest: Supervised learning algorithm that creates a forest & makes an ensemble of random decision trees.

Supporting Information

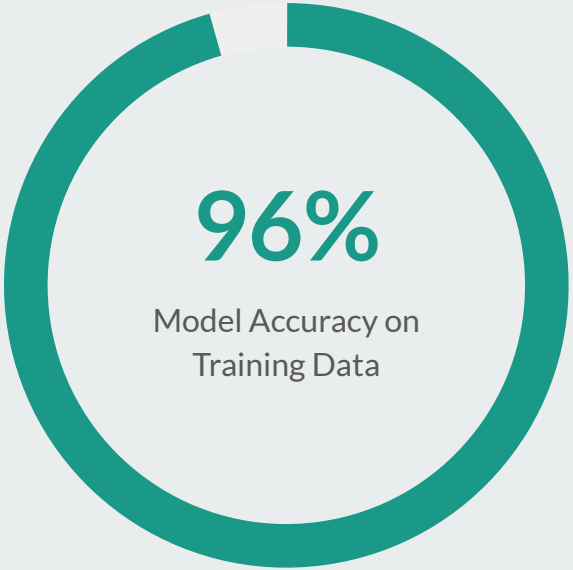


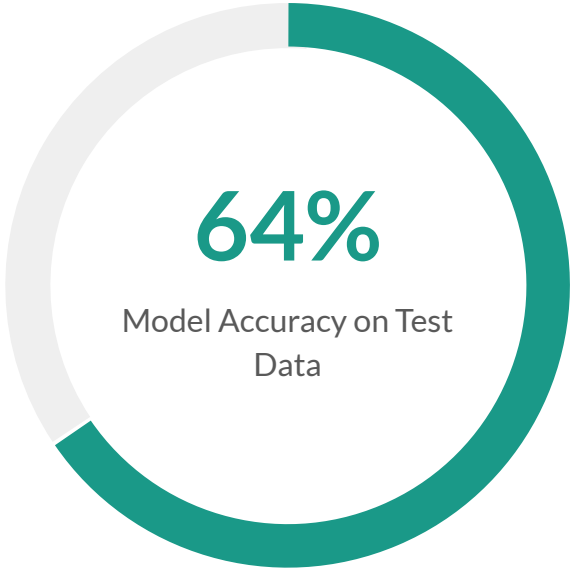
Supporting Information: Naive Bayes



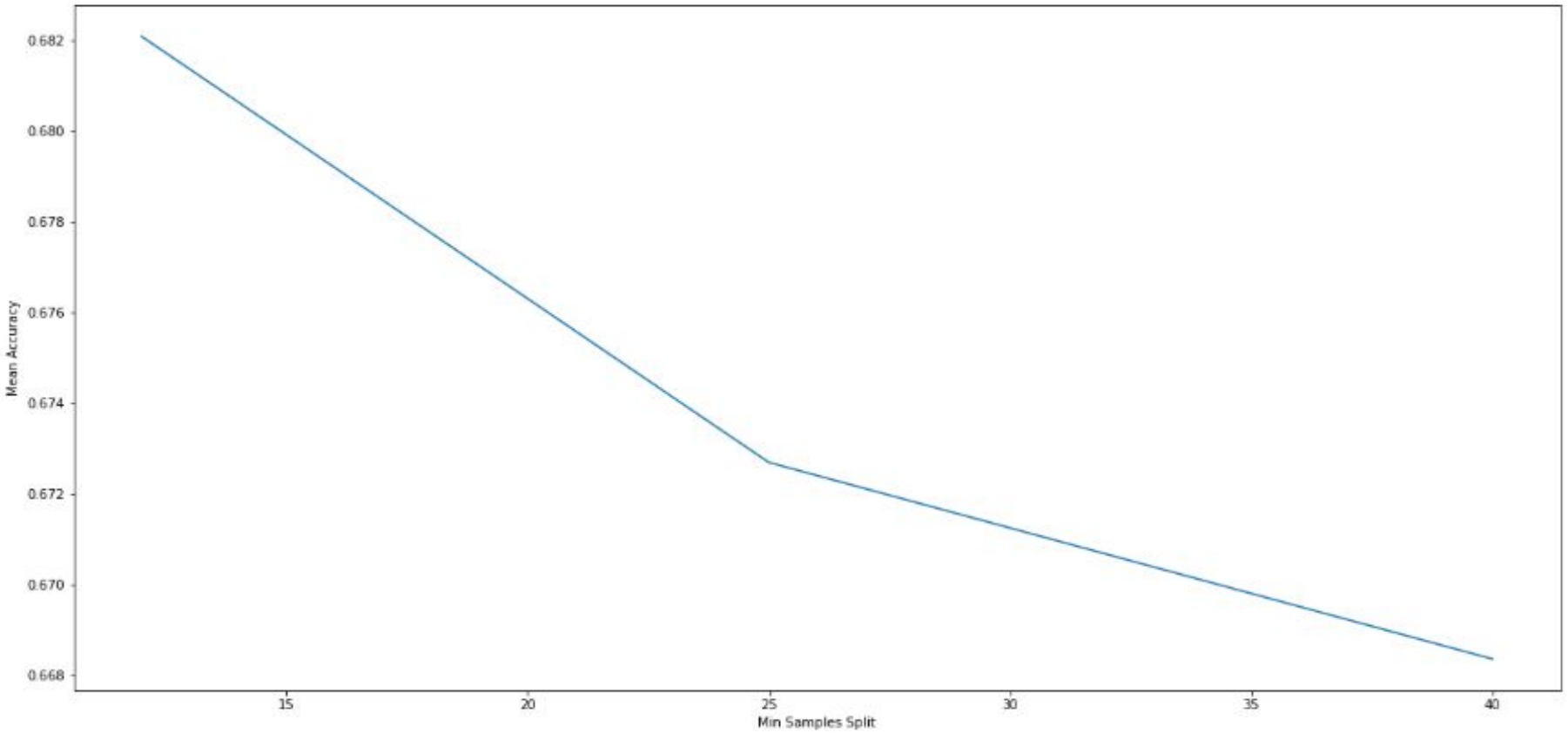


Supporting Information: Random Forest





Supporting Information: Accuracy of Parameters

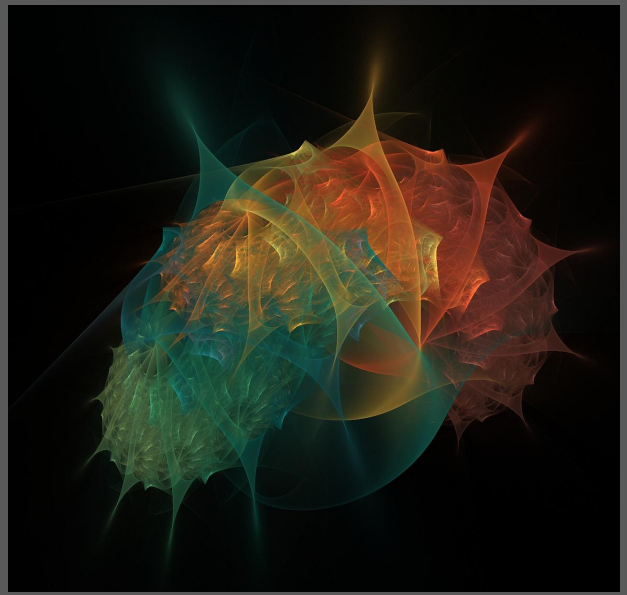


Word Cloud: Specialized Features

Word Cloud: Most Common Cyber Security

Word Cloud: Most Common Hacking

Executive Summary



Questions?