NLP Classification on Subreddits

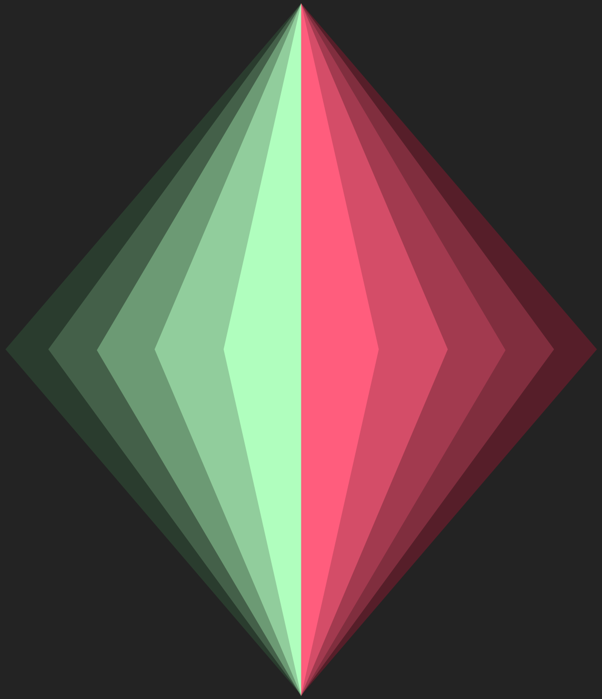


Table of contents

01

Overview

Brief Introduction & Problem Statement

02

Data Scraping, Cleaning and EDA

Process, Text Preprocessing, EDA

03

Modelling

Training ML Models & Performance Evaluation

04

Conclusion

Limitations & Recommendations

01

Overview

What is NLP?

TL;DR

Natural Language Processing is a hyponym of computer science and artificial intelligence that focuses on interactions between computers and human languages.

Goal: read, understand and decipher human languages in a way that it is valuable.

Distinguish Subreddits

Distinguish posts between two subreddits:

r/datascience r/SoftwareEngineering

Problem Statement

Which classification model can best distinguish which subreddit a post belongs to?

02 Data Scraping, Cleaning and EDA

Data Scraping

Data extracted using Reddit’s API

→ only 1000 posts per day

→ 25 posts each time → use time.sleep() function to allow for breaks

→ Json format

Data Cleaning

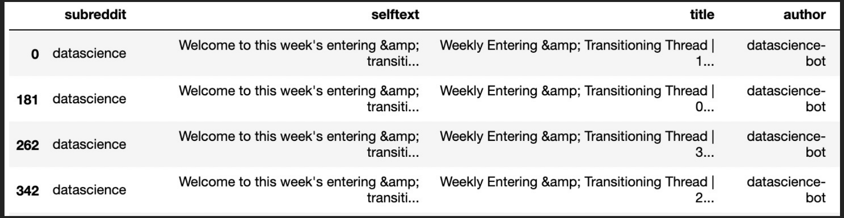
→ Drop Duplicate Values

→ Fill Null Values

→ Drop posts by bots and moderators

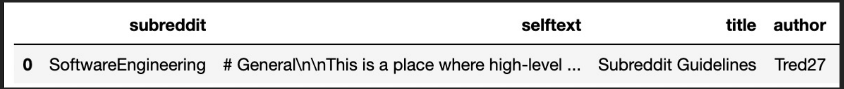
→ Combine selftext and title columns

r/datascience



	subreddit	selftext	title	author
0	datascience	Welcome to this week's entering &amp; transiti...	Weekly Entering &amp; Transitioning Thread | 1...	datascience-bot
181	datascience	Welcome to this week's entering &amp; transiti...	Weekly Entering &amp; Transitioning Thread | 0...	datascience-bot
262	datascience	Welcome to this week's entering &amp; transiti...	Weekly Entering &amp; Transitioning Thread | 3...	datascience-bot
342	datascience	Welcome to this week's entering &amp; transiti...	Weekly Entering &amp; Transitioning Thread | 2...	datascience-bot

r/SoftwareEngineering



	subreddit	selftext	title	author
0	SoftwareEngineering	# General\n\nThis is a place where high-level ...	Subreddit Guidelines	Tred27

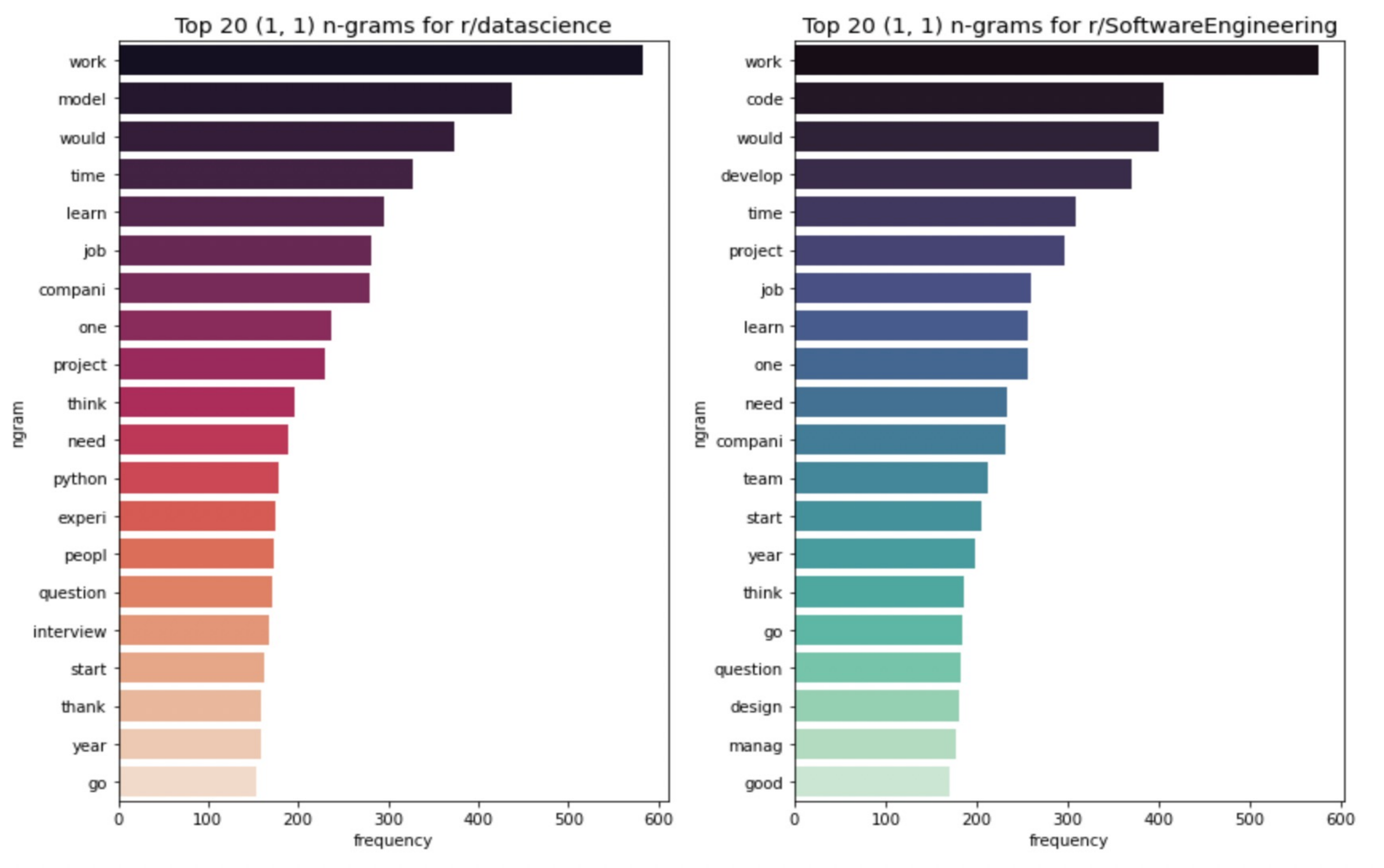
Text Preprocessing

- → Json Format → Remove html tags and URLs
- → Remove nonalphanumeric and other characters
- → Lemmatize and Stemming
- → Stopwords
  - NLTK
  - Other stopwords: data, science, software, engineer, www, reddit, com, like, use, know

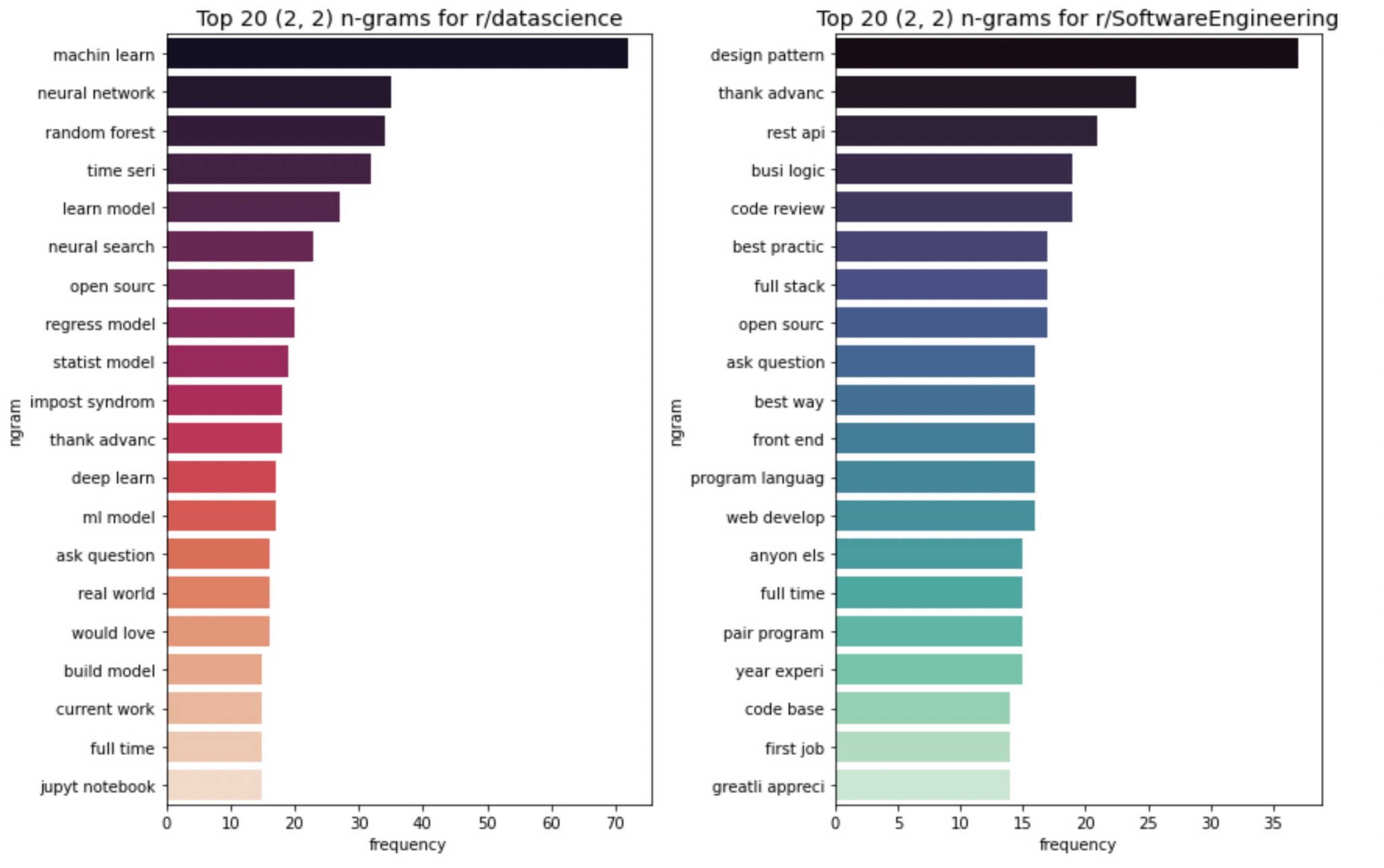
EDA - Bag of N grams

- → Extension of Bag of Words, n is any sequence of n tokens
- → More context around each word

Frequency of 1 - grams



Frequency of 2 - grams



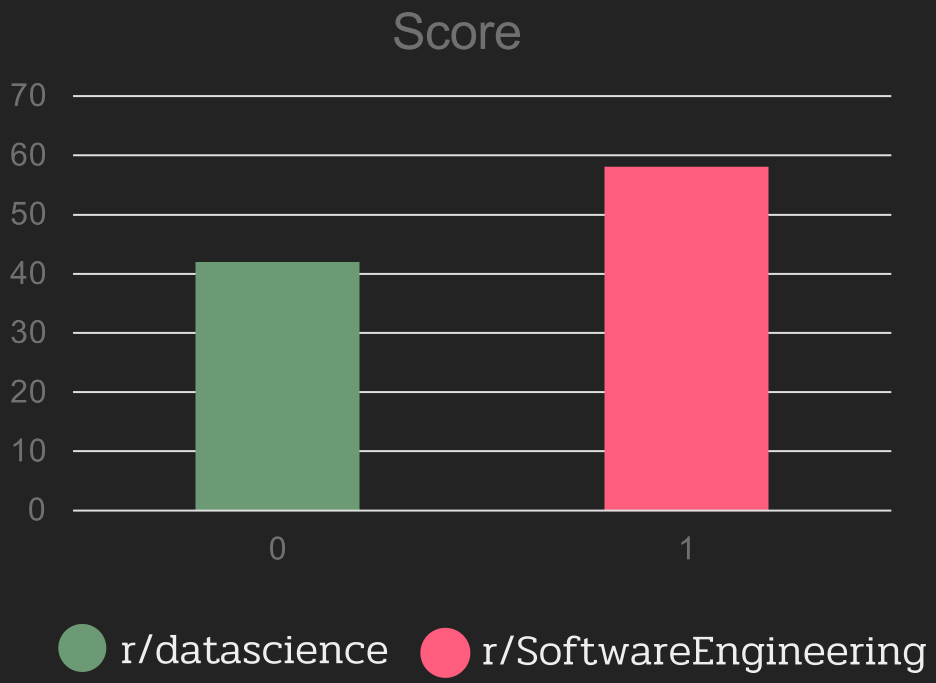
03

Modelling

Modelling Process

- → Establish a baseline model
- → Train Test Split
- → Create Pipeline → Vectorizer and Classifier
- → GridSearchCV → Tune Hyperparameters
- → Fit Model
- → Evaluate Model using Evaluation Metrics
  - → Confusion Matrix
  - → ROC Curve
  - → Feature Importance

Baseline Model



r/datascience [0]

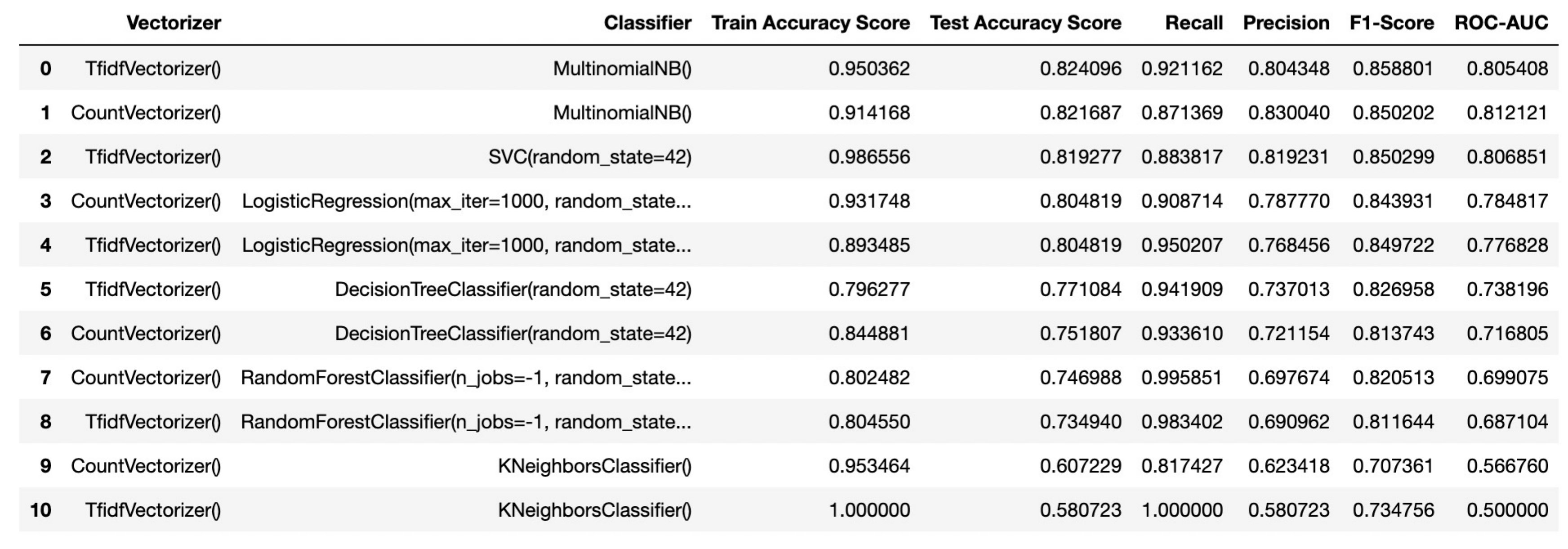
58.1%

r/SoftwareEngineering [1]

41.9%

Model Results

- Top 5 models scored > 80%
- TFIDF Vectorizer performed better on average
- K Neighbors Classifier performed worst despite high Train Accuracy Score

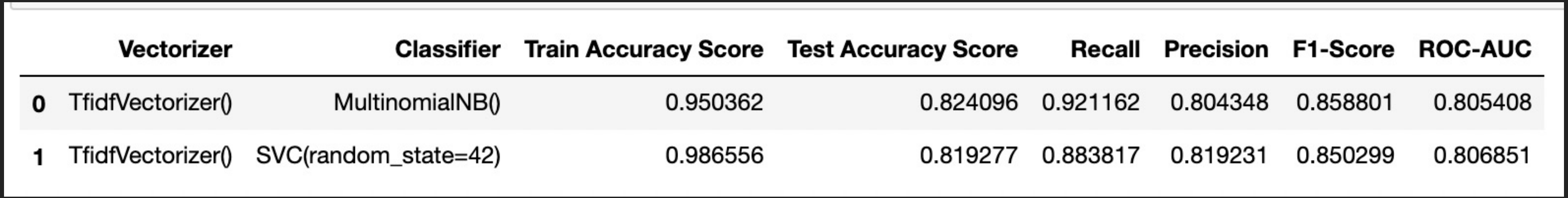


	Vectorizer	Classifier	Train Accuracy Score	Test Accuracy Score	Recall	Precision	F1-Score	ROC-AUC
0	TfidfVectorizer()	MultinomialNB()	0.950362	0.824096	0.921162	0.804348	0.858801	0.805408
1	CountVectorizer()	MultinomialNB()	0.914168	0.821687	0.871369	0.830040	0.850202	0.812121
2	TfidfVectorizer()	SVC(random_state=42)	0.986556	0.819277	0.883817	0.819231	0.850299	0.806851
3	CountVectorizer()	LogisticRegression(max_iter=1000, random_state...	0.931748	0.804819	0.908714	0.787770	0.843931	0.784817
4	TfidfVectorizer()	LogisticRegression(max_iter=1000, random_state...	0.893485	0.804819	0.950207	0.768456	0.849722	0.776828
5	TfidfVectorizer()	DecisionTreeClassifier(random_state=42)	0.796277	0.771084	0.941909	0.737013	0.826958	0.738196
6	CountVectorizer()	DecisionTreeClassifier(random_state=42)	0.844881	0.751807	0.933610	0.721154	0.813743	0.716805
7	CountVectorizer()	RandomForestClassifier(n_jobs=-1, random_state...	0.802482	0.746988	0.995851	0.697674	0.820513	0.699075
8	TfidfVectorizer()	RandomForestClassifier(n_jobs=-1, random_state...	0.804550	0.734940	0.983402	0.690962	0.811644	0.687104
9	CountVectorizer()	KNeighborsClassifier()	0.953464	0.607229	0.817427	0.623418	0.707361	0.566760
10	TfidfVectorizer()	KNeighborsClassifier()	1.000000	0.580723	1.000000	0.580723	0.734756	0.500000

Top 2 Models

Multinomial Naïve Bayes - TFIDF Vectorizer

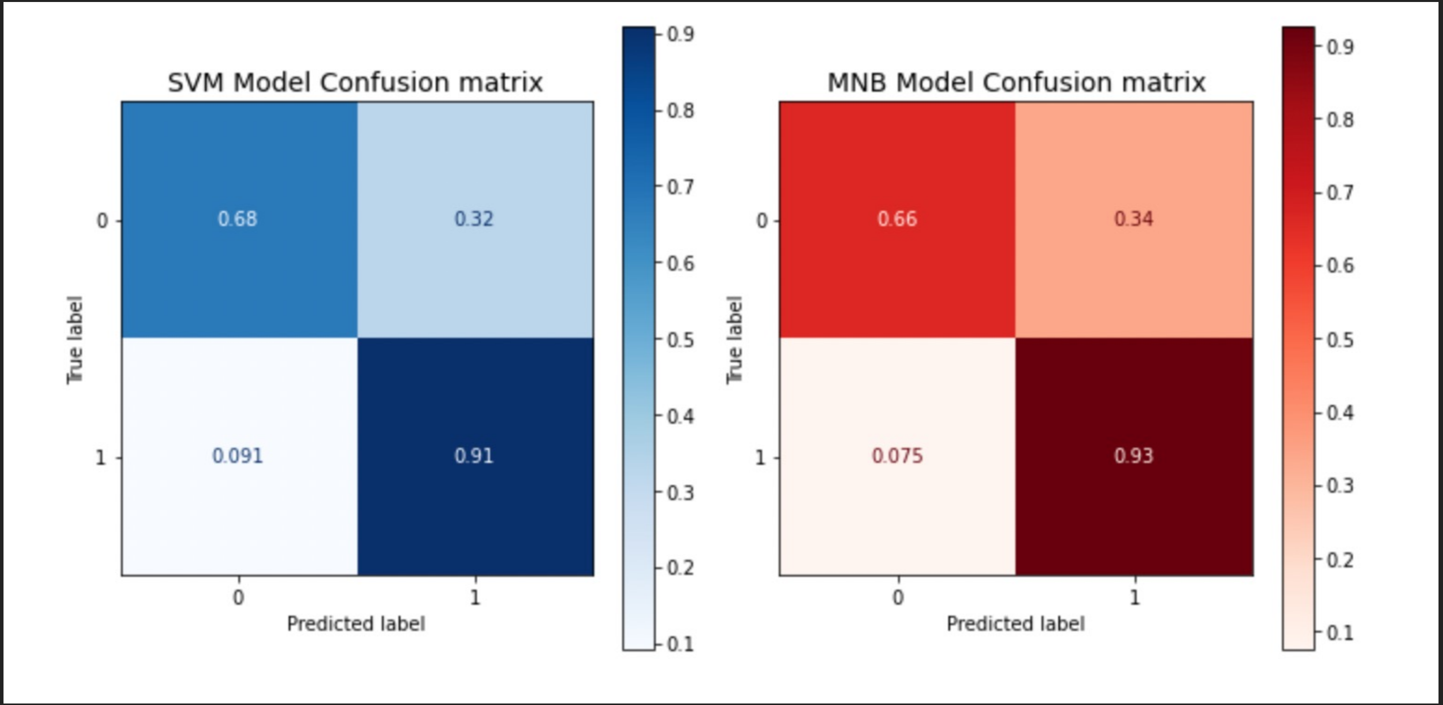
Support Vector Classifier – TFIDF Vectorizer



	Vectorizer	Classifier	Train Accuracy Score	Test Accuracy Score	Recall	Precision	F1-Score	ROC-AUC
0	TfidfVectorizer()	MultinomialNB()	0.950362	0.824096	0.921162	0.804348	0.858801	0.805408
1	TfidfVectorizer()	SVC(random_state=42)	0.986556	0.819277	0.883817	0.819231	0.850299	0.806851

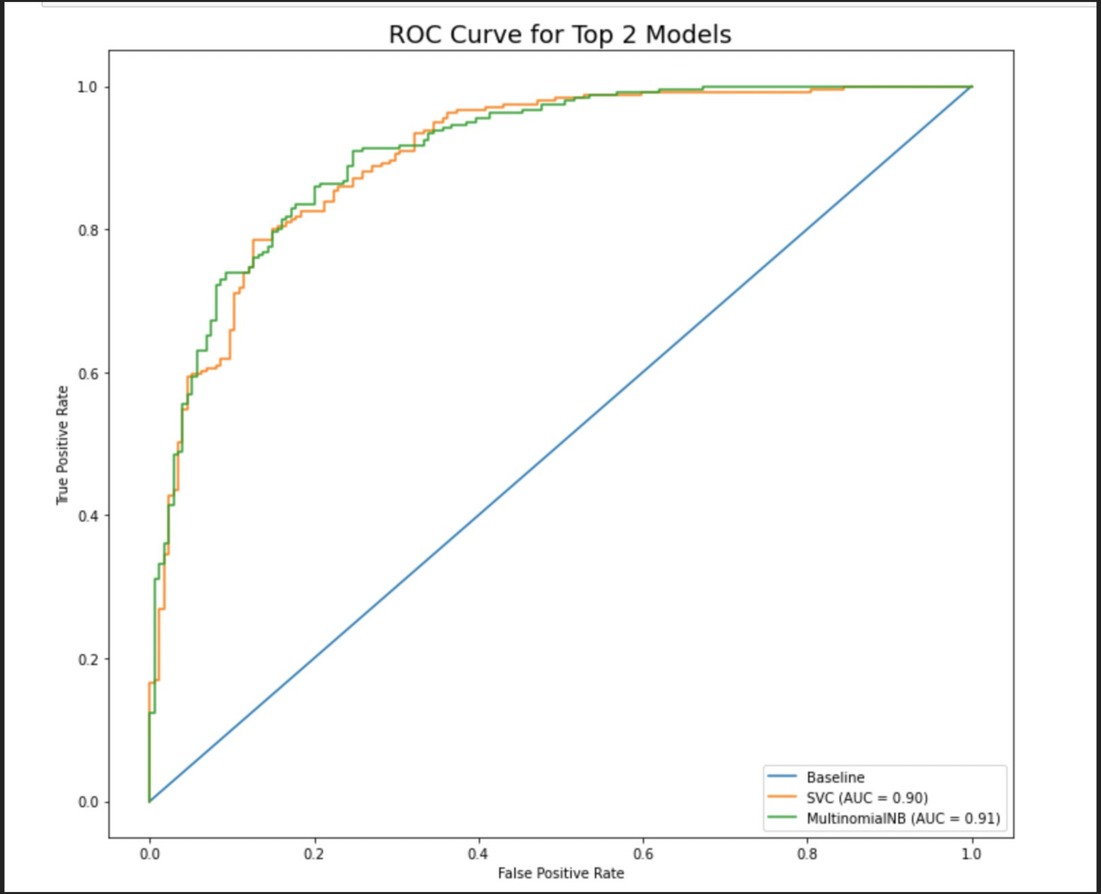
Confusion Matrix

- Comparable Results
- MNB better at predicting r/SoftwareEngineering

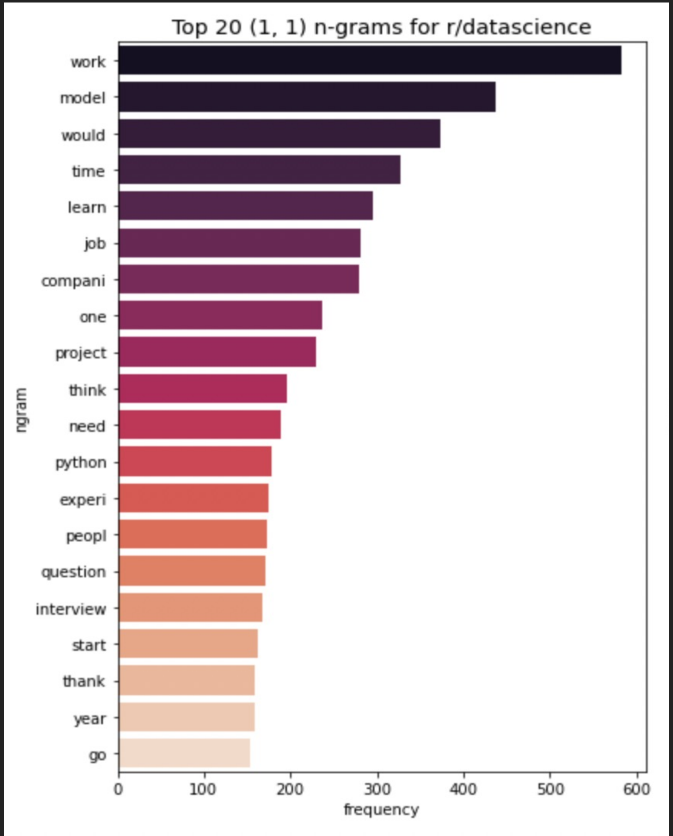


ROC Curve

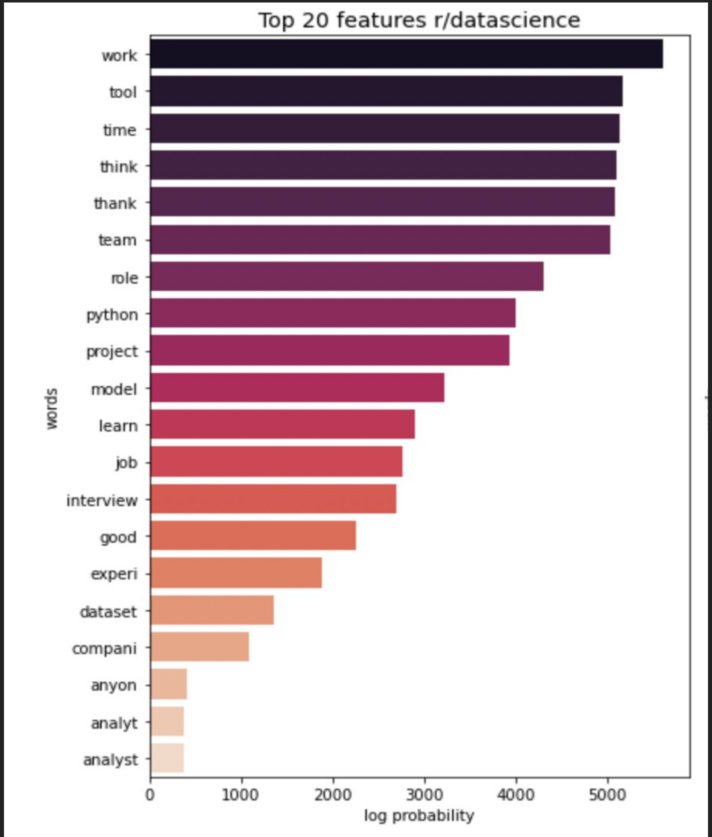
- Both have high ROC-AUC score
- MNB has slight edge over SVC



Feature Importance - MNB

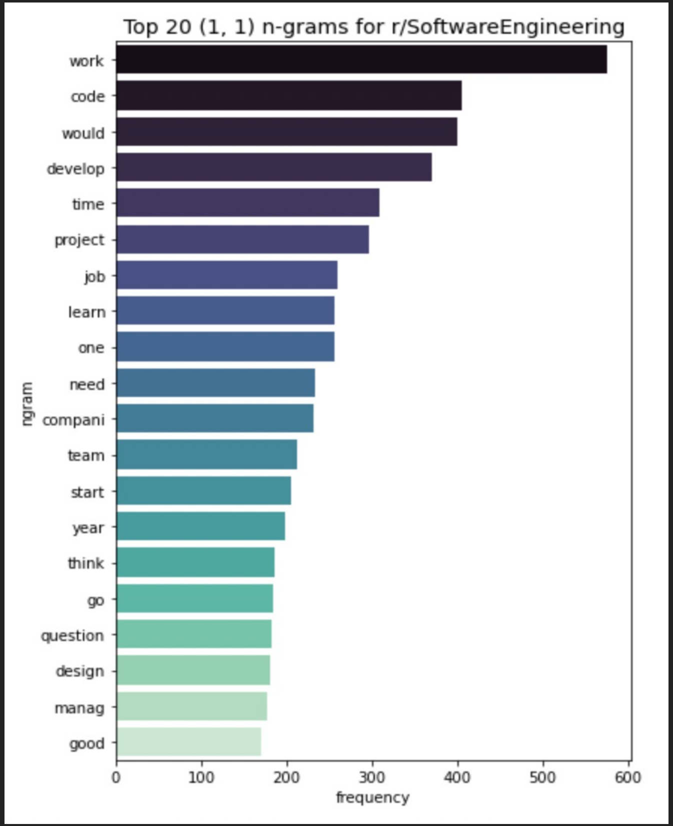


Top 20 (1,1) n-grams

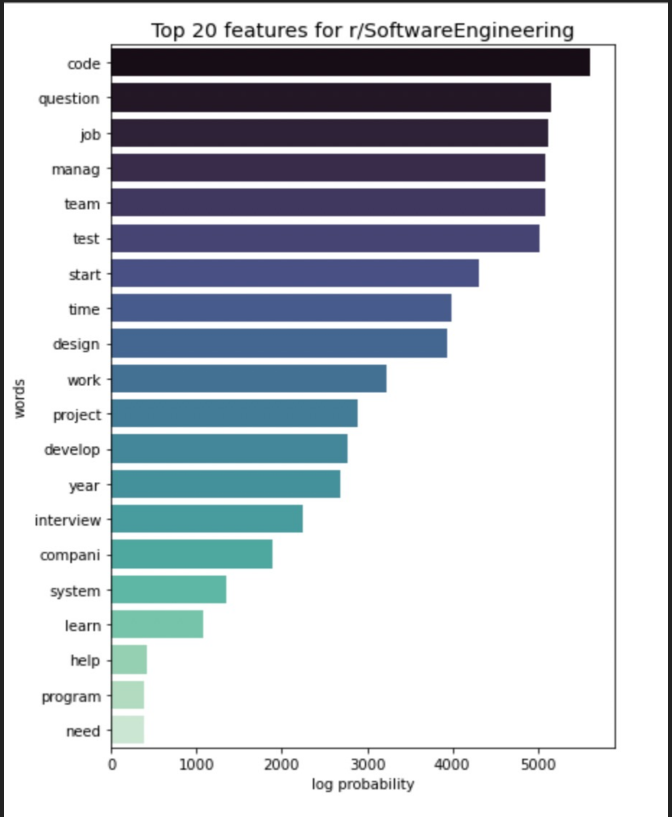


Top 20 Important Features

Feature Importance -MNB



Top 20 (1,1) n-grams



Top 20 Important Features

Feature Importance - SVC

- Best features for a Support Vector Classifier → coefficient values of the features. However, this works best with the linear kernel

- SVC best estimator

```
Pipeline(steps=[('tfidf', TfidfVectorizer(max_df=0.9, ngram_range=(1, 2))),
                ('svc', SVC(C=1, kernel='sigmoid', random_state=42))])
```

- Blackbox Model

It is not possible to determine best features as our non-linear separable data is mapped into another higher dimensional place → cannot place weights on the features

Best Model: Multinomial Naive Bayes

Predict between classes fairly

Higher ROC Score

Inferential Characteristics (Feature Importance)

04 Conclusion

Limitations and Recommendations

Misclassifications

77 / 415 posts misclassified - 19%

More Data

Reddit API: Only 1000 posts, 25 each time

Timespan of Data Collection

Other Models

RandomisedSearch, HashingVectorizers

Stakeholder Research

User Behavior/Content Engagement

THANK YOU!