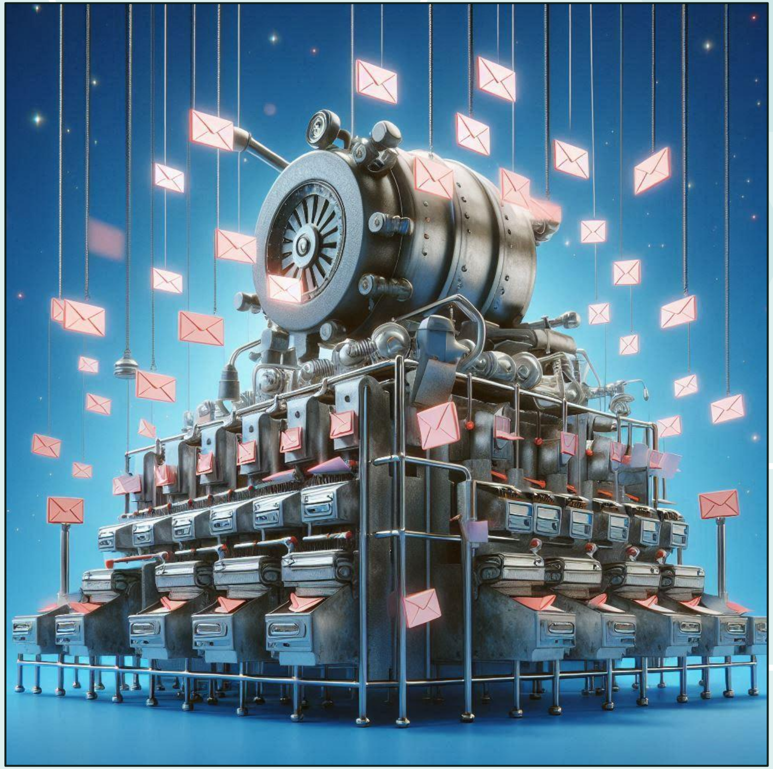
Spam Email Filter - Machine Learning Model



Agenda

01 Business Problem

02 Exploratory Data Analysis

03 Predictive Modeling

04 Conclusions

05 Recommendation

06 Next Steps

01 Business Problem

Our telecommunications company wants to break into the email market and needs to implement an effective SPAM filter.

We are charged with:

- Create a model that reliably detects SPAM in a way that will maximize customer satisfaction

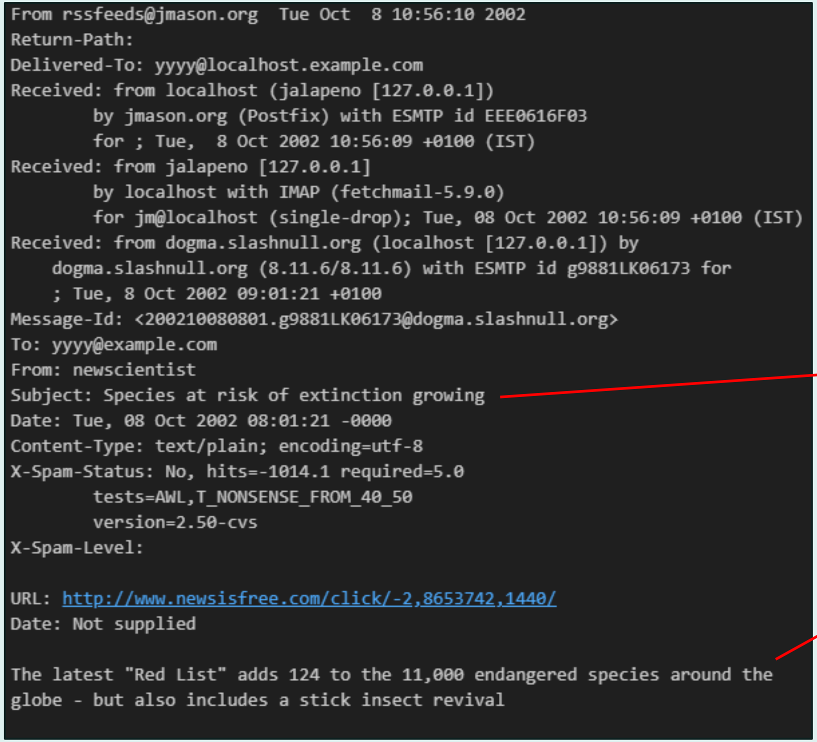
02 Data Analysis

Actual emails in the form of individual HTML files were used in the analysis

- 5280 HTML files in total
- Each email was previously labeled as HAM or SPAM and sorted into folders

Source: Index of /old/publiccorpus (apache.org)

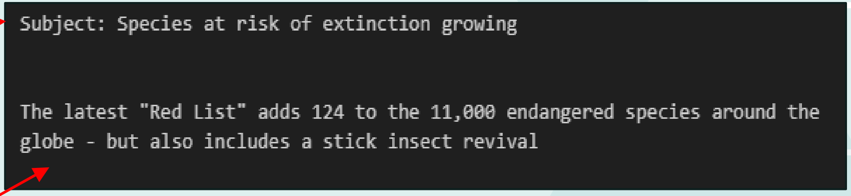
Email Importing and Preprocessing



```
From rssfeeds@jmason.org  Tue Oct  8 10:56:10 2002
Return-Path:
Delivered-To: yyyy@localhost.example.com
Received: from localhost (jalapeno [127.0.0.1])
        by jmason.org (Postfix) with ESMTP id EEE0616F03
        for ; Tue,  8 Oct 2002 10:56:09 +0100 (IST)
Received: from jalapeno [127.0.0.1]
        by localhost with IMAP (fetchmail-5.9.0)
        for jm@localhost (single-drop); Tue, 08 Oct 2002 10:56:09 +0100 (IST)
Received: from dogma.slashnull.org (localhost [127.0.0.1]) by
    dogma.slashnull.org (8.11.6/8.11.6) with ESMTP id g9881LK06173 for
    ; Tue, 8 Oct 2002 09:01:21 +0100
Message-Id: <200210080801.g9881LK06173@dogma.slashnull.org>
To: yyyy@example.com
From: newscientist
Subject: Species at risk of extinction growing
Date: Tue, 08 Oct 2002 08:01:21 -0000
Content-Type: text/plain; encoding=utf-8
X-Spam-Status: No, hits=-1014.1 required=5.0
        tests=AWL,T_NONSENSE_FROM_40_50
        version=2.50-cvs
X-Spam-Level:

URL: http://www.newsisfree.com/click/-2,8653742,1440/
Date: Not supplied

The latest "Red List" adds 124 to the 11,000 endangered species around the
globe - but also includes a stick insect revival
```



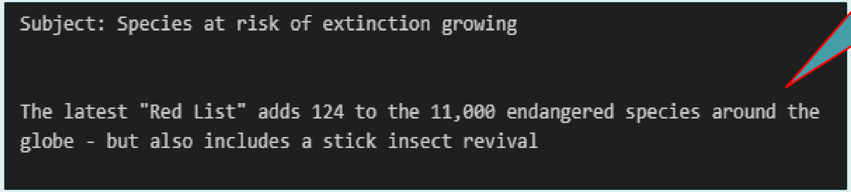
```
Subject: Species at risk of extinction growing


The latest "Red List" adds 124 to the 11,000 endangered species around the
globe - but also includes a stick insect revival
```

Subject Line vs Email Body

Subject Line:
- We expect 'dense' information about the nature of the email

Email Body:
- More information about the nature of the email, more verbose

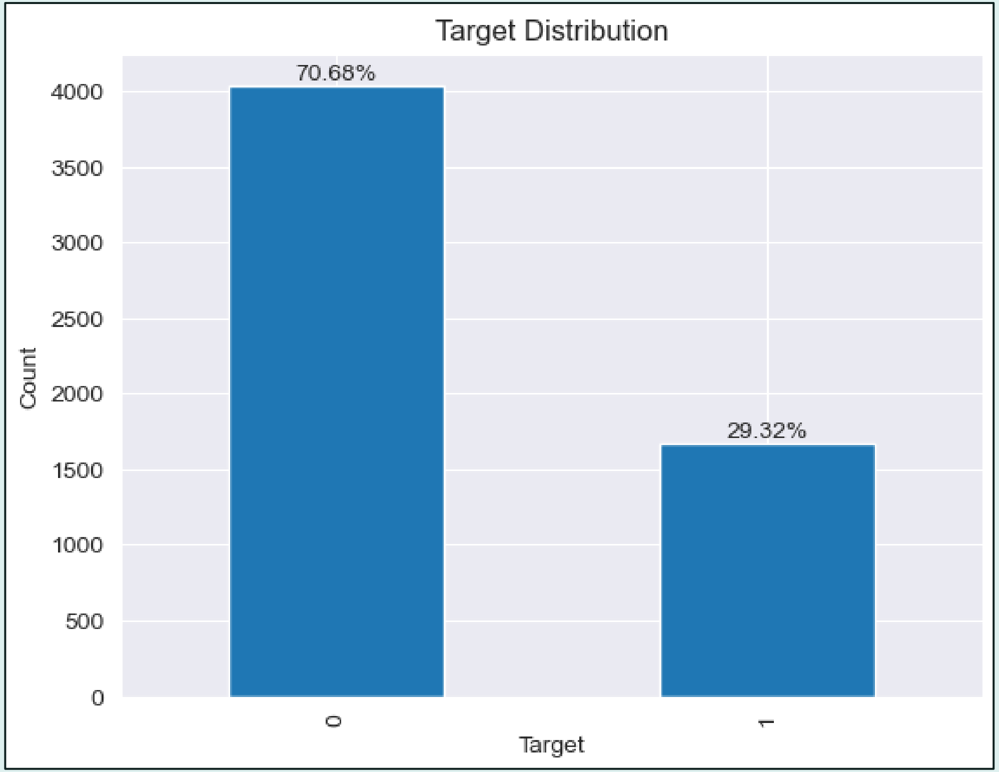


```
Subject: Species at risk of extinction growing


The latest "Red List" adds 124 to the 11,000 endangered species around the
globe - but also includes a stick insect revival
```

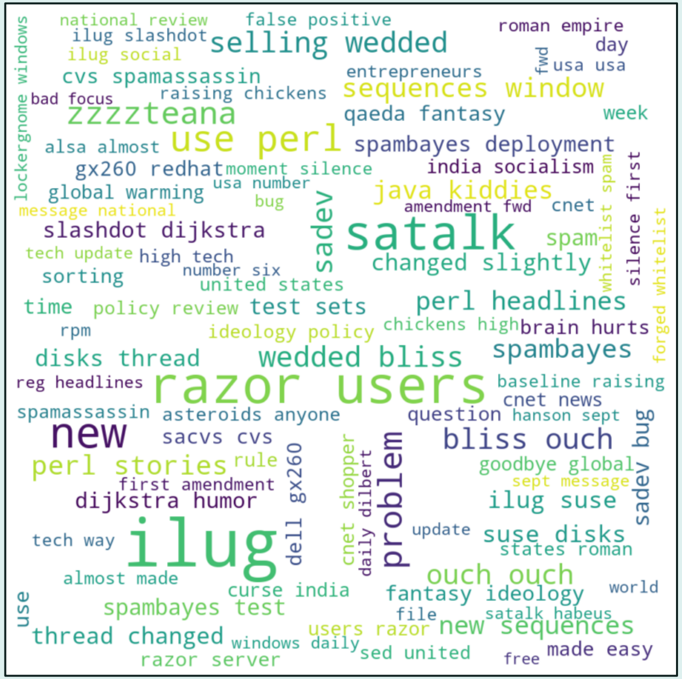
- ➢ Modeling the Subject and Body separately…

Target Distribution Imbalance

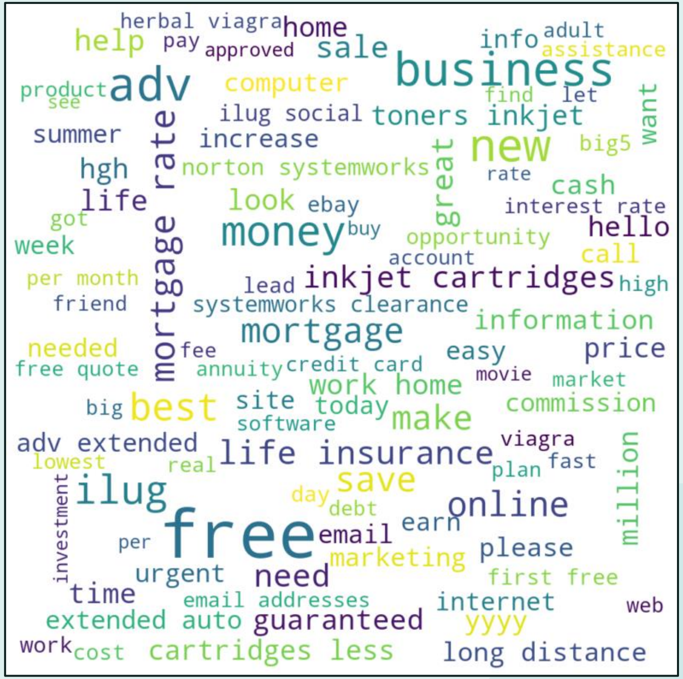


Subject Line Word Clouds by Class

HAM



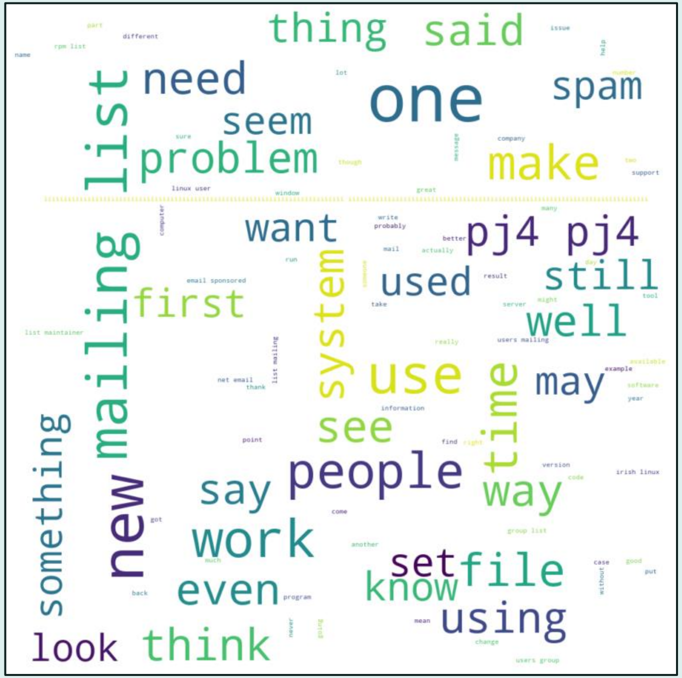
SPAM



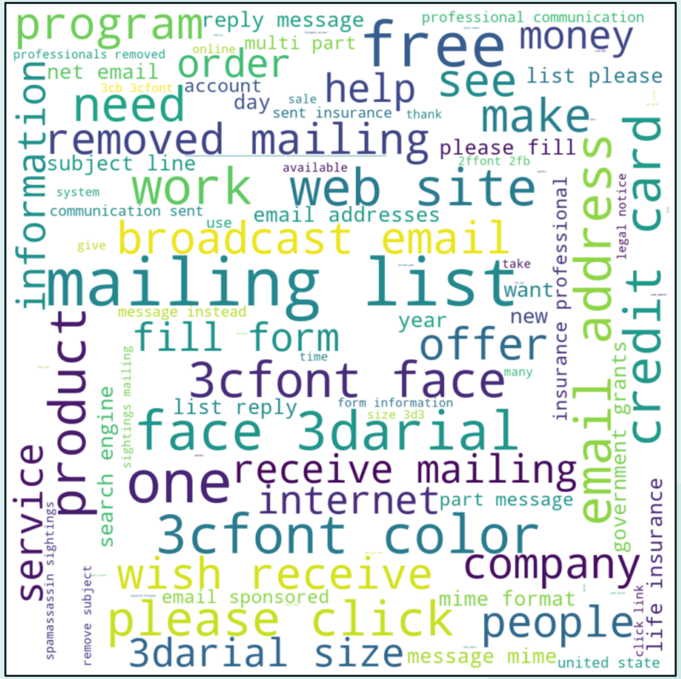
Note: Models are built on company internal email data, will need to adjust models trained on anonymized customer emails data after deployment

Email Body Word Clouds by Class

HAM



SPAM



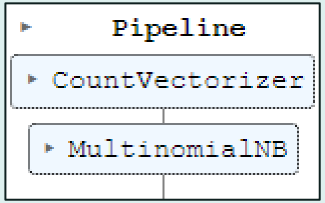
03 Predictive Modeling

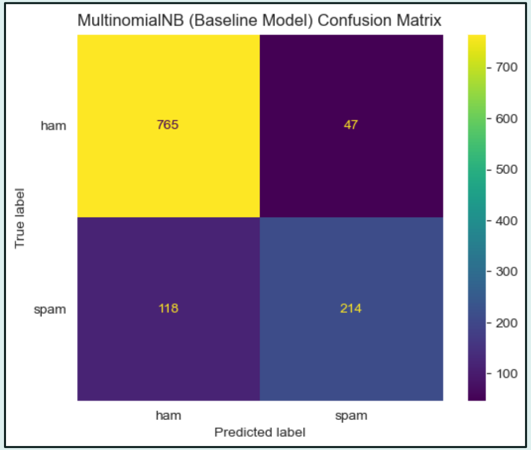
<ins>BUSINESS REQUIREMENT</ins>:

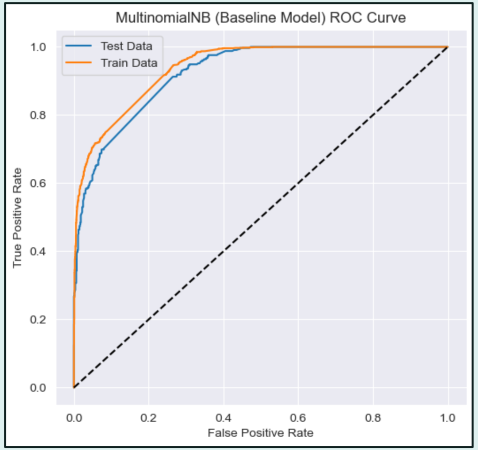
- Getting SPAM in the inbox irritates our customers, but missing an important message because we classified it as SPAM will really anger them!
- Optimizing models on **Precision (minimize False-Pos)**, then **Accuracy**
- Less than 1% of customers HAM emails are allowed to go to the SPAM classification

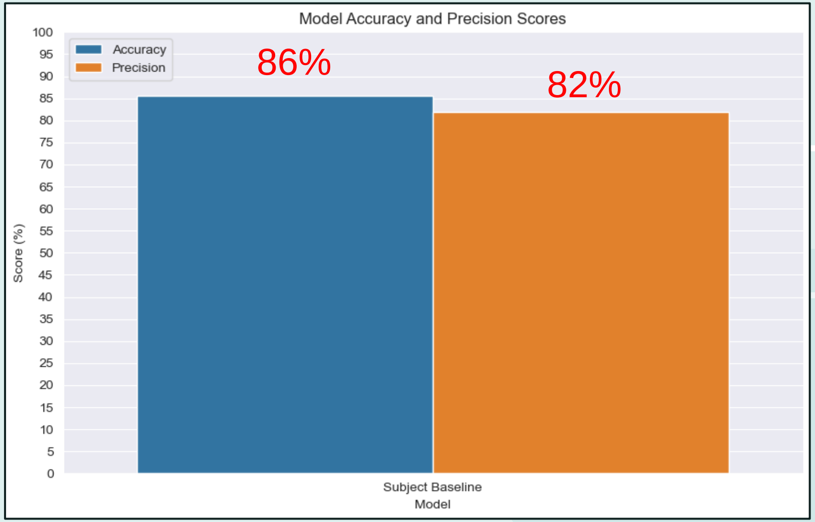
Subject Line Baseline Model

- Regex tokenizer of >2 characters
- Binary vectorizer
- Multinomial Bayes Classifier



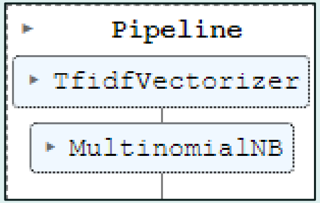


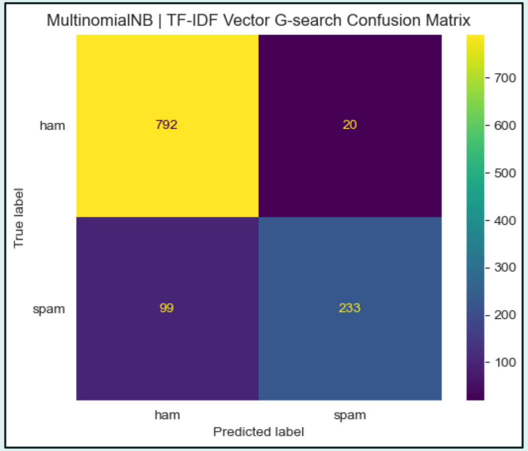




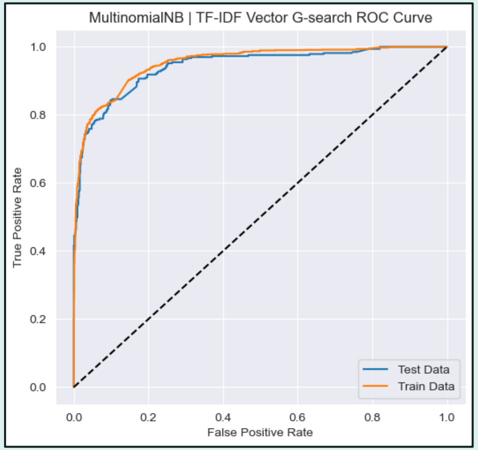
Subject Line Final Model

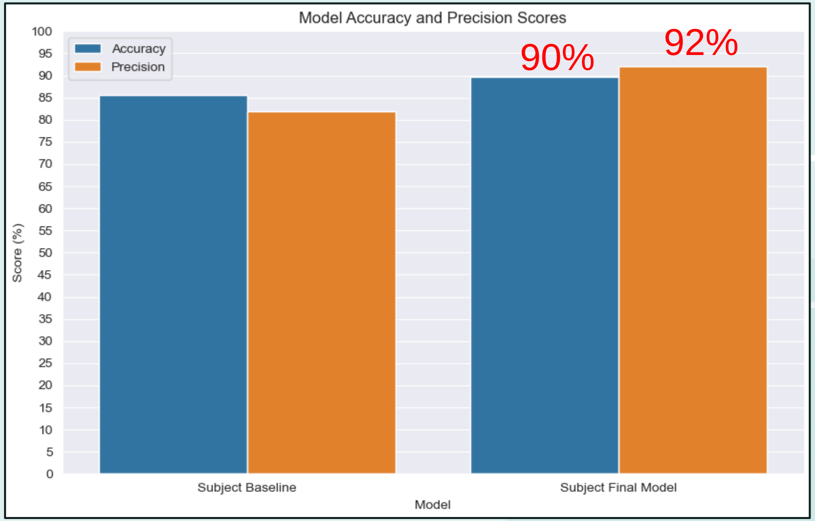
- Word_punct_tokenizer
- TFIDF vectorizer
- Multinomial Bayes Classifier



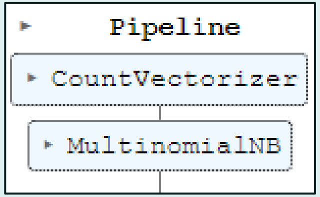

MultinomialNB | TF-IDF Vector G-search Confusion Matrix

	ham	spam
ham	792	20
spam	99	233


MultinomialNB | TF-IDF Vector G-search ROC Curve


Model Accuracy and Precision Scores

Email Body Baseline Model

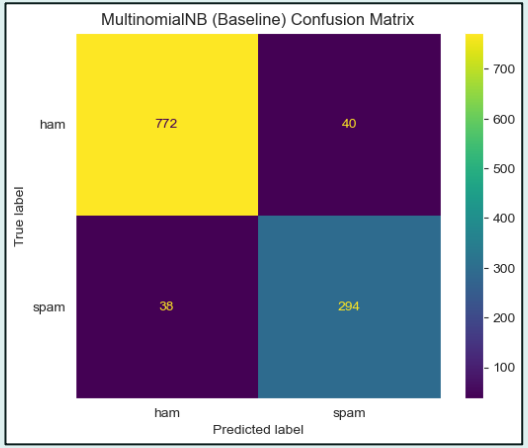


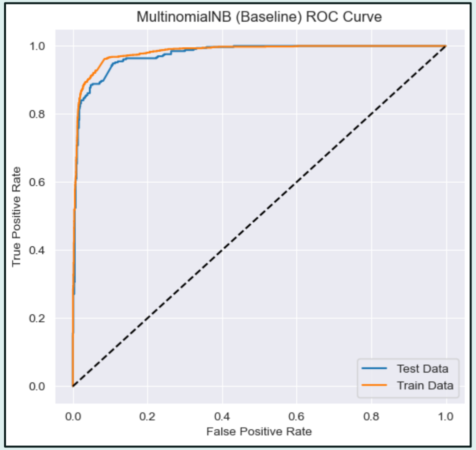
Pipeline

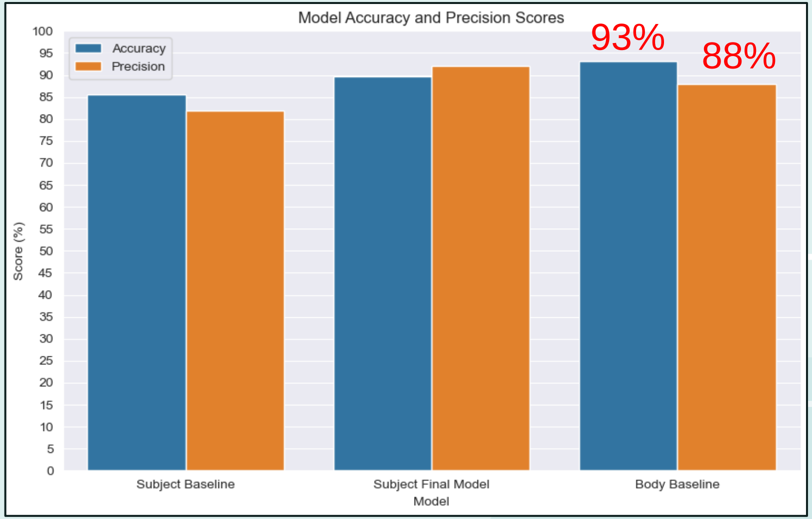
CountVectorizer

MultinomialNB

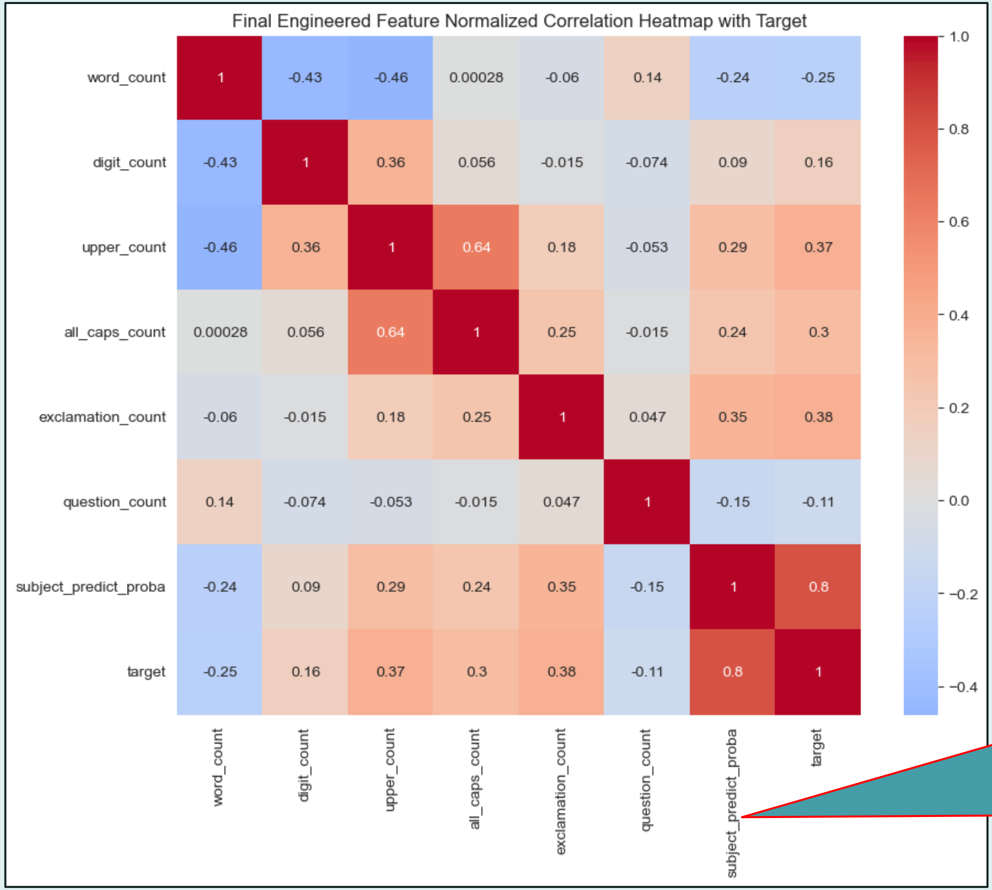
- Regex tokenizer of >2 characters
- Binary vectorizer
- Multinomial Bayes Classifier







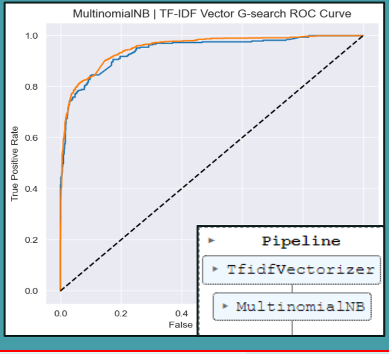
Email Body Feature Engineering



Final Engineered Feature Normalized Correlation Heatmap with Target

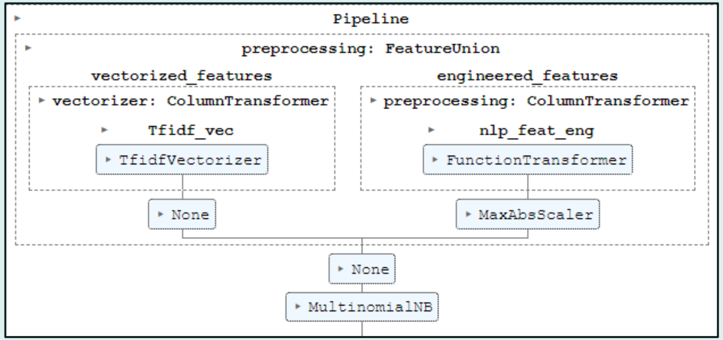
	word_count	digit_count	upper_count	all_caps_count	exclamation_count	question_count	subject_predict_proba	target
word_count	1	-0.43	-0.46	0.00028	-0.06	0.14	-0.24	-0.25
digit_count	-0.43	1	0.36	0.056	-0.015	-0.074	0.09	0.16
upper_count	-0.46	0.36	1	0.64	0.18	-0.053	0.29	0.37
all_caps_count	0.00028	0.056	0.64	1	0.25	-0.015	0.24	0.3
exclamation_count	-0.06	-0.015	0.18	0.25	1	0.047	0.35	0.38
question_count	0.14	-0.074	-0.053	-0.015	0.047	1	-0.15	-0.11
subject_predict_proba	-0.24	0.09	0.29	0.24	0.35	-0.15	1	0.8
target	-0.25	0.16	0.37	0.3	0.38	-0.11	0.8	1

SUBJECT LINE PREDICTIONS



MultinomialNB | TF-IDF Vector G-search ROC Curve

Email Body With Feature Engineering



Pipeline

preprocessing: FeatureUnion

vectorized_features

vectorizer: ColumnTransformer

Tfidf_vec

TfidfVectorizer

None

engineered_features

preprocessing: ColumnTransformer

nlp_feat_eng

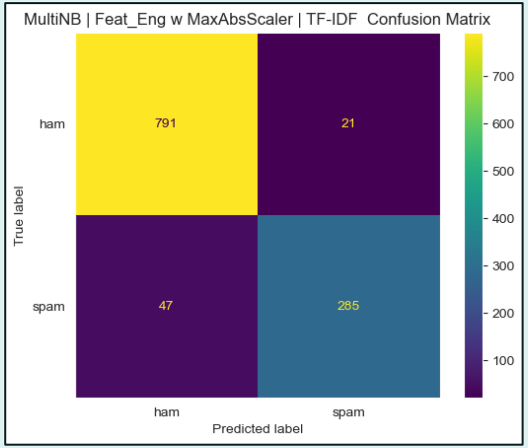
FunctionTransformer

MaxAbsScaler

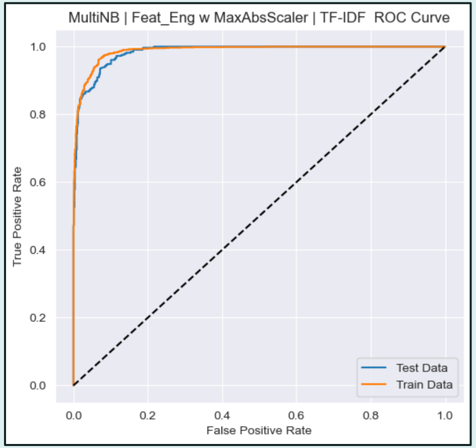
None

MultinomialNB

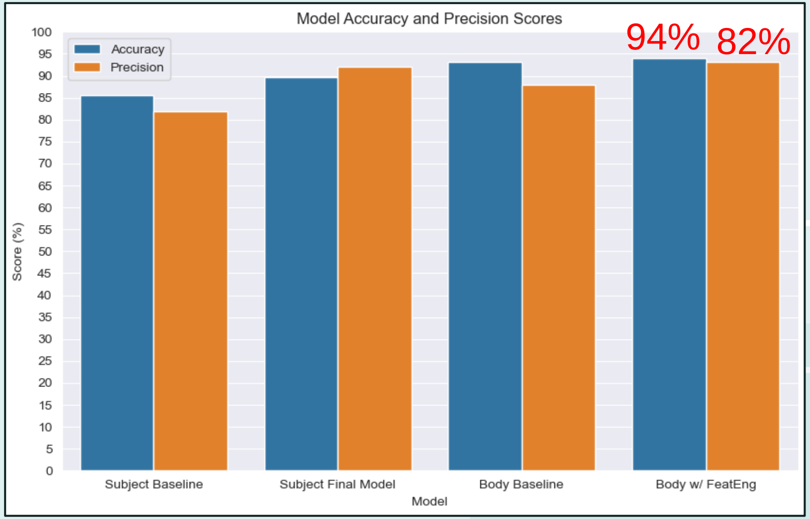
- Regex tokenizer of >2 characters
- TF-IDF vectorizer
- Multinomial Bayes Classifier



MultiNB | Feat_Eng w MaxAbsScaler | TF-IDF Confusion Matrix



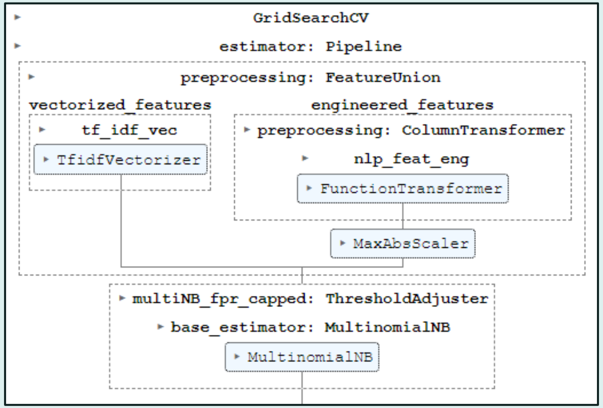
MultiNB | Feat_Eng w MaxAbsScaler | TF-IDF ROC Curve



Model Accuracy and Precision Scores

94% 82%

Email Body – Best Naive Bayes Model



GridSearchCV

estimator: Pipeline

preprocessing: FeatureUnion

vectorized_features

tf_idf_vec

TfidfVectorizer

engineered_features

preprocessing: ColumnTransformer

nlp_feat_eng

FunctionTransformer

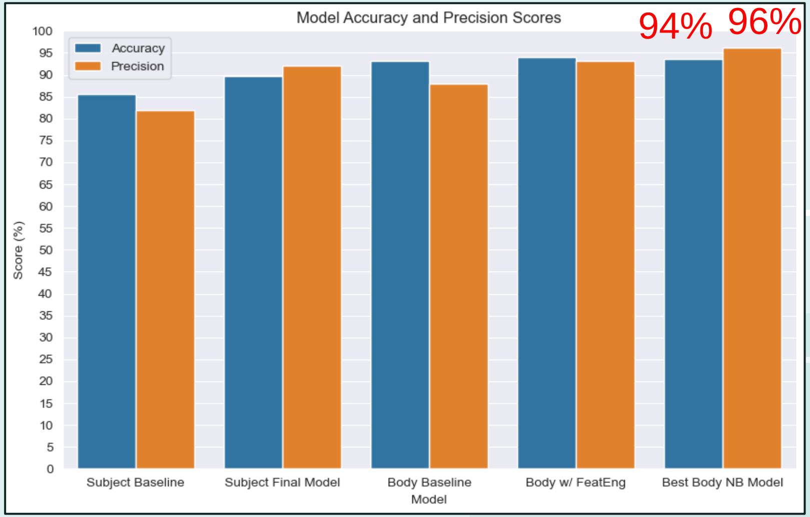
MaxAbsScaler

multiNB_fpr_capped: ThresholdAdjuster

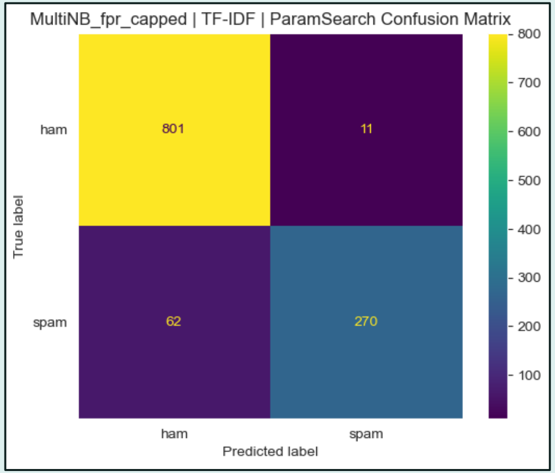
base_estimator: MultinomialNB

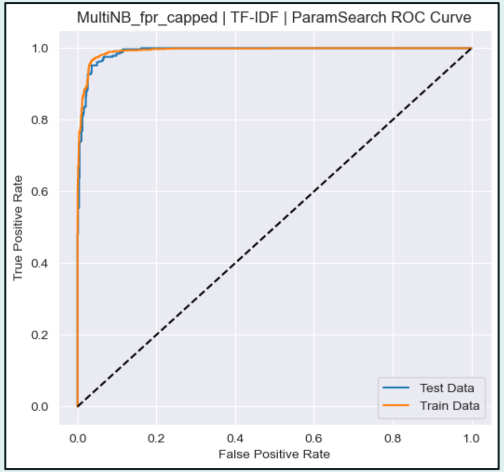
MultinomialNB

- Hyperparameter Gridsearch
- NB Model wrapped in Prob Threshold Adjuster (<1% FPR)



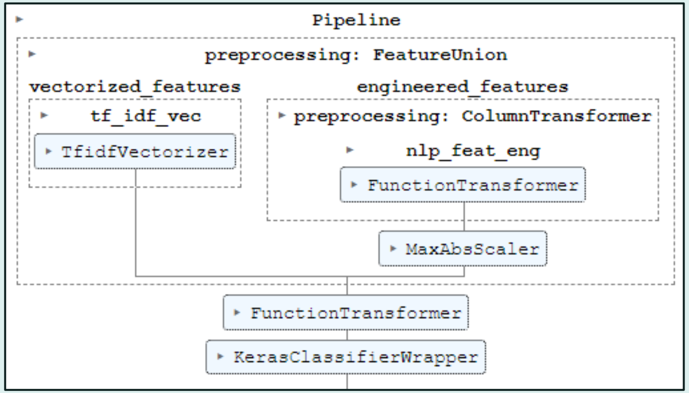
94% 96%





Email Body – Neural Network Baseline

- One hidden layer
- Threshold Adjusted (<1% FPR)



Pipeline

preprocessing: FeatureUnion

vectorized_features

tf_idf_vec

TfidfVectorizer

engineered_features

preprocessing: ColumnTransformer

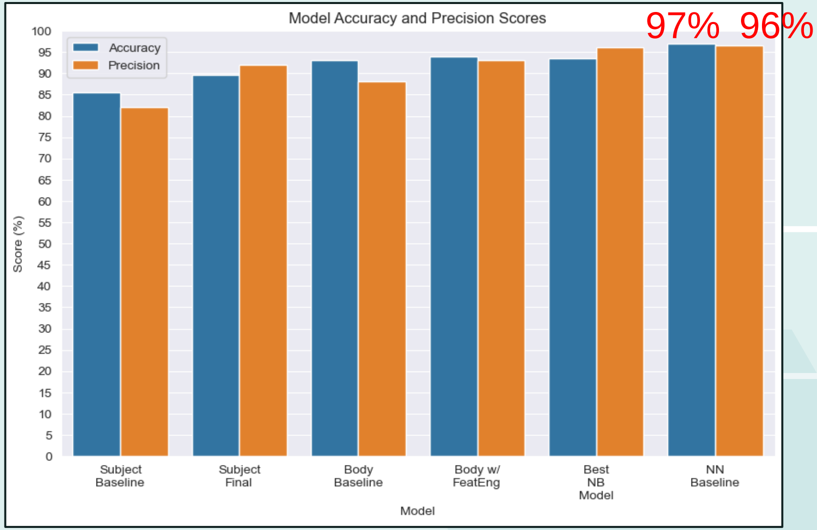
nlp_feat_eng

FunctionTransformer

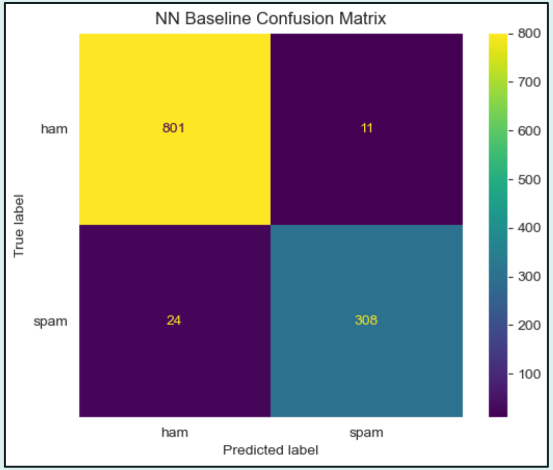
MaxAbsScaler

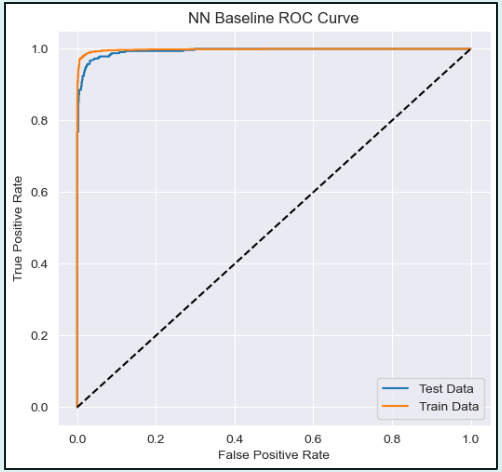
FunctionTransformer

KerasClassifierWrapper

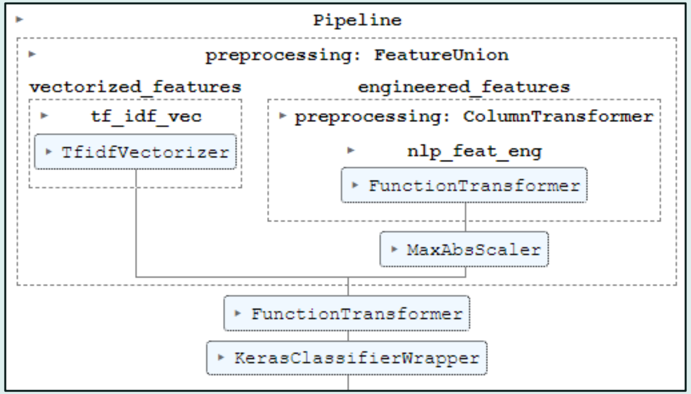


97% 96%





Email Body – Neural Network Final



Pipeline

preprocessing: FeatureUnion

vectorized_features

tf_idf_vec

TfidfVectorizer

engineered_features

preprocessing: ColumnTransformer

nlp_feat_eng

FunctionTransformer

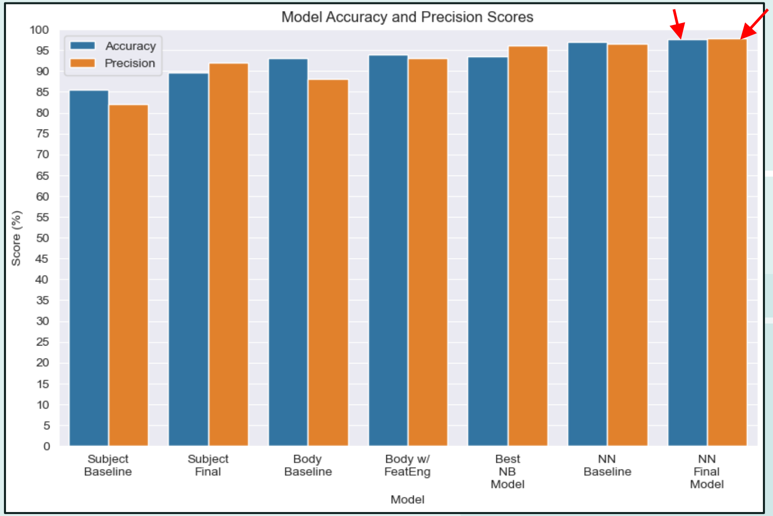
MaxAbsScaler

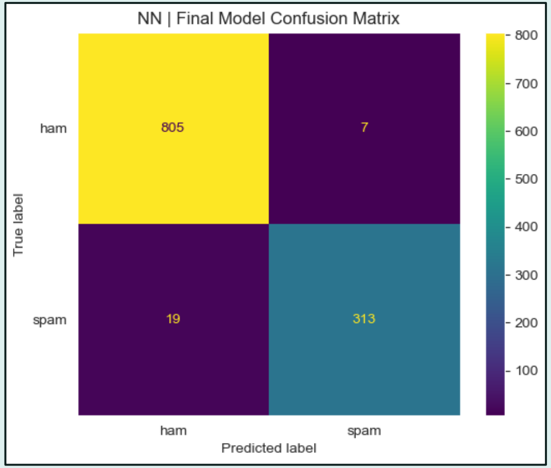
FunctionTransformer

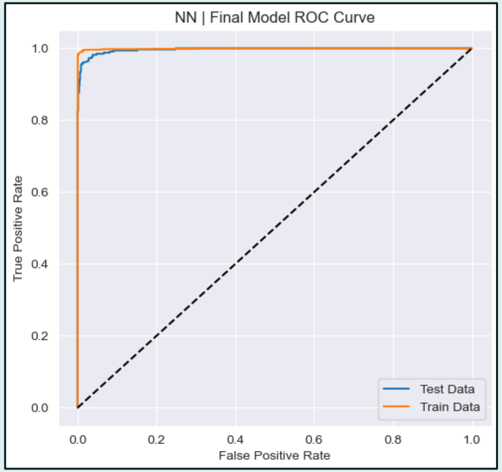
KerasClassifierWrapper

- 5 hidden layers
- 2 dropout layers
- Gridsearch using Hyperband()
- Threshold Adjusted (<1% FPR)

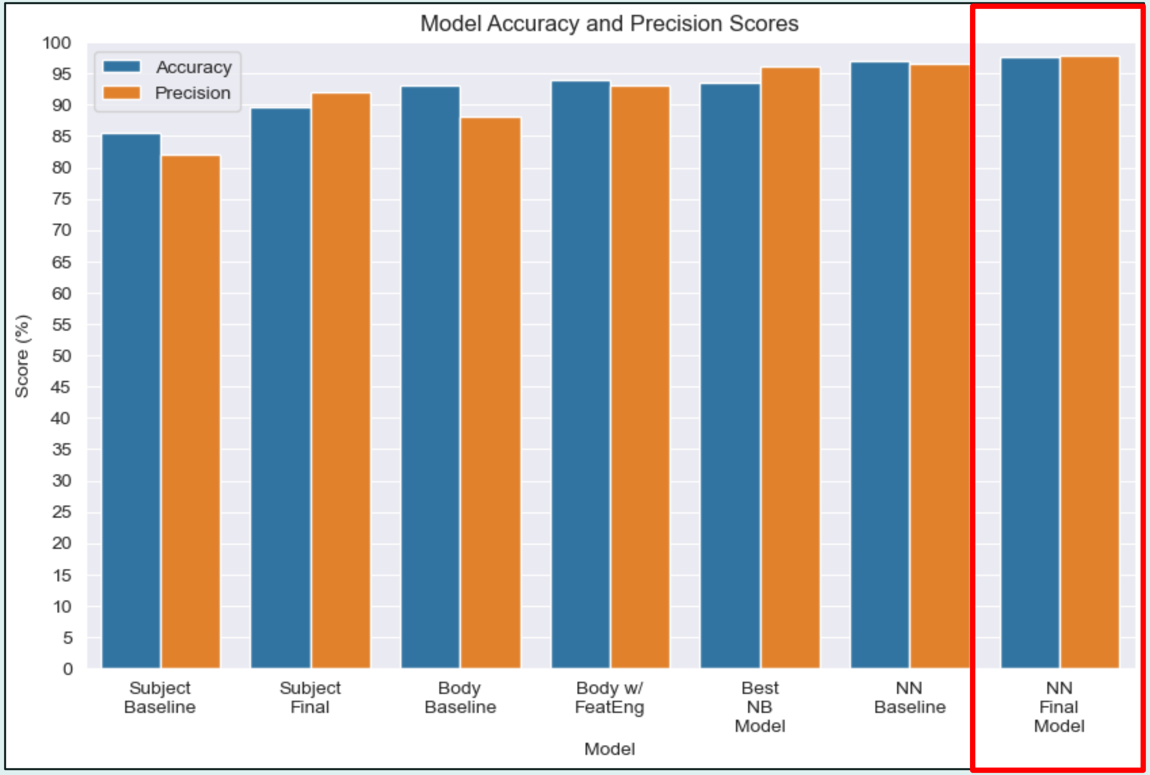
98% 98%







04 Conclusion



Final Model Test Metrics:

- Precision: **97.7%**
- Accuracy: **97.8%**

Final Business Metrics:

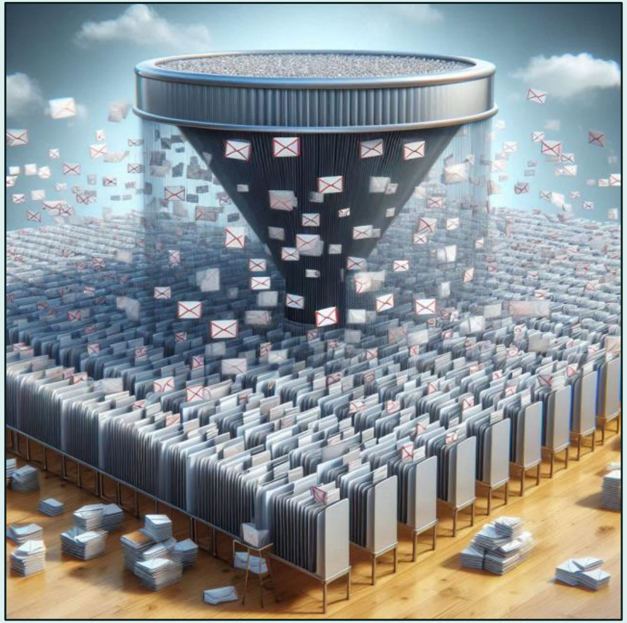
- False Pos Rate (HAM going to SPAM): **0.9%**
- False Neg Rate (SPAM going to HAM): **6%**

05 Recommendation

1. Implement the best Neural Network model during initial Email service rollout

2. Continue to retrain and retune the models on new anonymized customer data

3. Investigate other potential model improvements (see next steps)



06 Next Steps

1. Continue to improve the model
    1. Rerun and tune the models with the ‘Cheat words’ removed
    2. Investigate tokenizing based on wordcloud algorithm
    3. Check other model architectures (RandomForest, XGBoost)
    4. Tune NN for subject-line only, perhaps effective and more efficient
    5. Investigate verbiage and nature of SPAM, are there features we can capture
    6. Investigate the False cases; are there features we can capture
    7. Investigate keeping URLs instead of scrubbing them preprocess

2. Acquire larger data sets of up-to-date emails

Thanks!

Questions? Please contact me at:

Dale DeFord
daledeford@gmail.com