Spam SMS Detection Model

Jake Greenberg & Matthew Gulbin

Outline

- Business Problem
- Data Overview
- Preprocessing
- Baseline Model
- Final Models:
  - Logistic Regression
  - Multinominal Naive Bayes
- Business Recommendation
- Next Steps

Business Problem

In our day-to-day lives, we are plagued by spam text messages. These messages are not only annoying, but also pose a security risk to our customers.

<u>Solution:</u> Build a spam detection model that will block potentially harmful messages from users' inboxes while minimizing the number of false positives

Data Overview

Using a dataset of 5,574 text messages from 2015, consisting of both spam and non-spam entries.

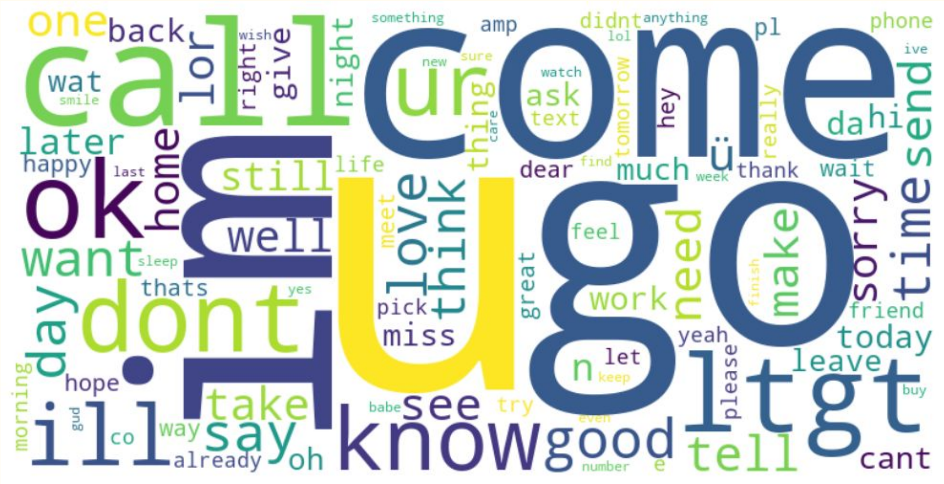
86% non-spam
14% spam

Data Preprocessing

1. Lower case
2. Remove punctuation
3. Remove *stopwords* (very common words like “and”, “the”, “a”, etc.)
4. *Lemmatization* (reducing every word to its root form; ex: “running” → “run”)
5. Applied a TF-IDF Vectorizer, which generated a score for each word based on how frequently it occurred in each string and the whole dataset

Spam

Not Spam



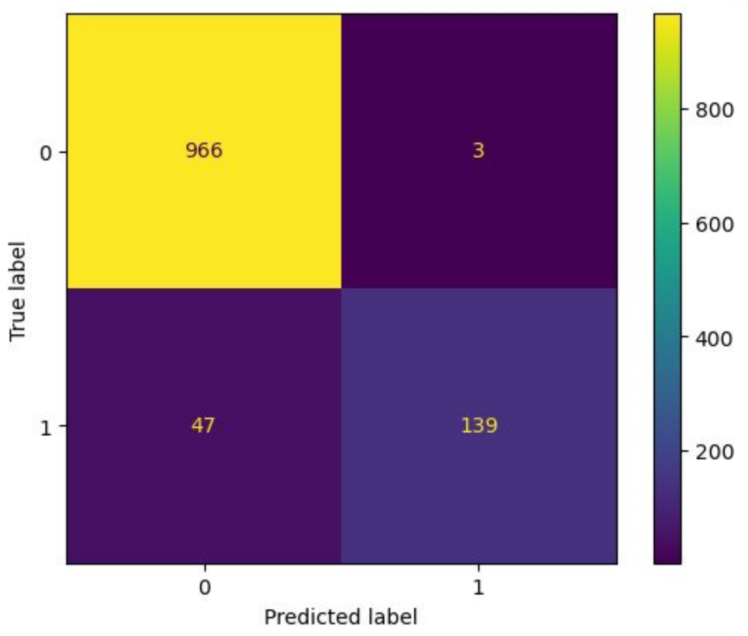
Baseline Model

Created a Logistic Regression model with no modification to the data

Precision Score = 0.98

Recall Score = 0.75

F1 Score = 0.84



Logistic Regression

Improvements from Baseline

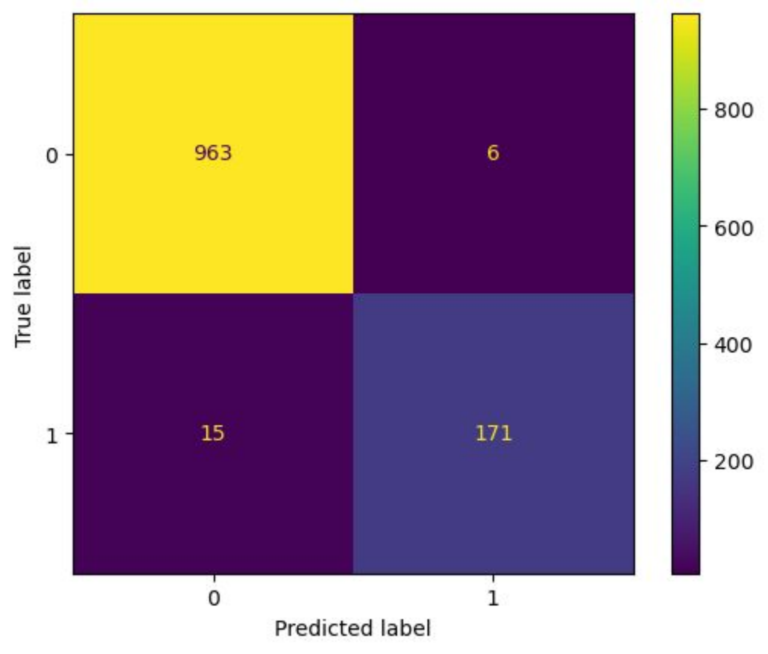
- Implementing oversampling using SMOTE
- Adjusting the C-value for the model
- Adjusting the parameters of the TFIDF Vectorizer

Results

Precision Score = 0.97

Recall Score = 0.92

F1 Score = 0.94



Multinominal Naive Bayes

Baseline

Precision (Spam):

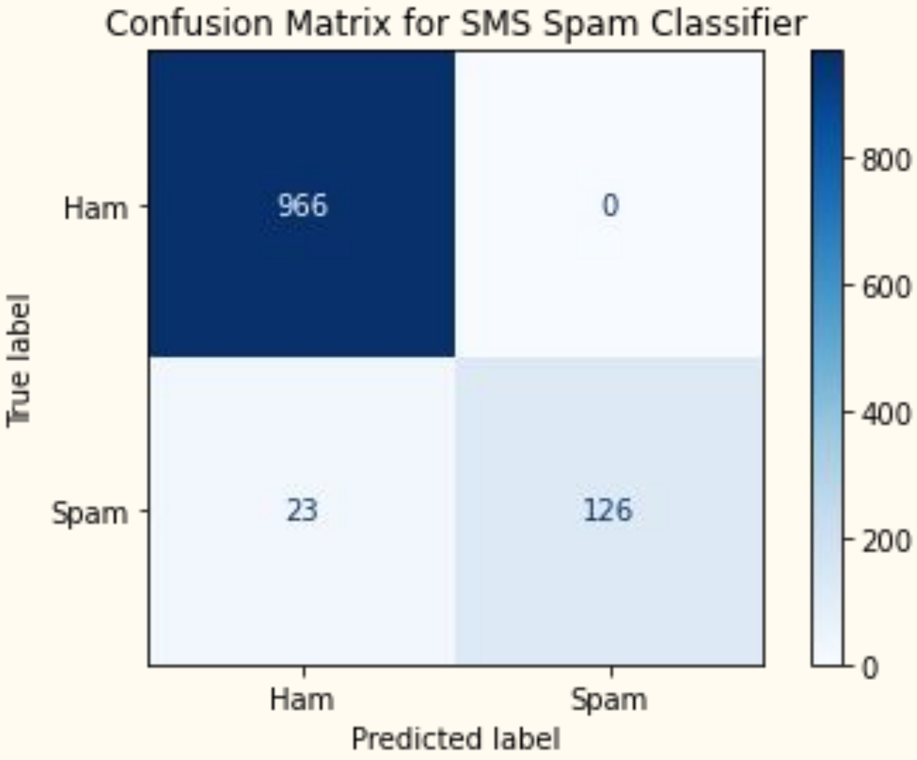
- 100%

Recall (Spam):

- 84.56%

F1 Score (Spam):

- 91.6%



Threshold Adjustment

Precision (Spam):

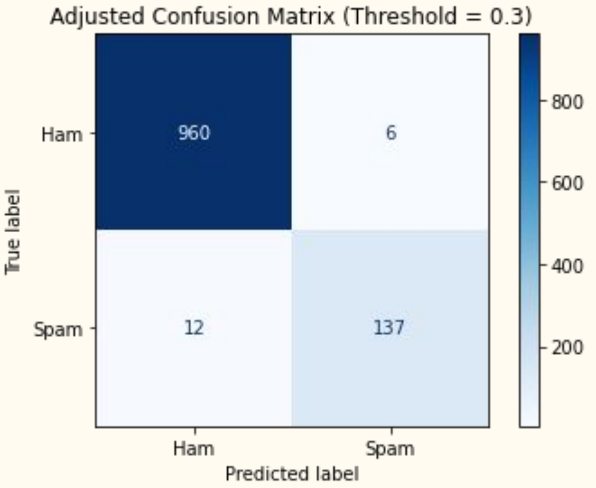
- 100%

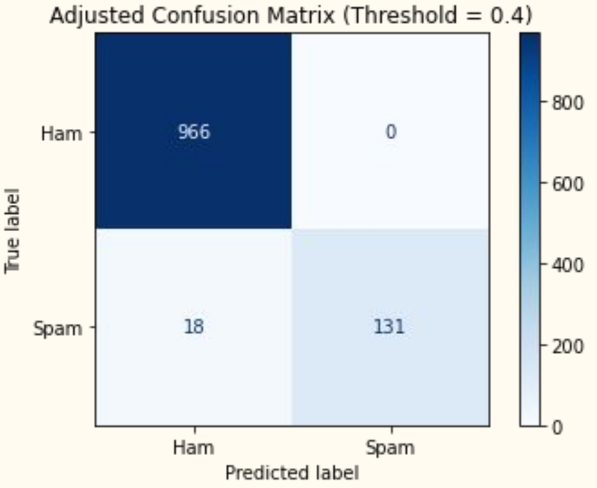
Recall (Spam):

- 87.9%

F1 Score (Spam):

- 93.8%





Enhanced Naive Bayes Code with N-grams

Precision (Spam):

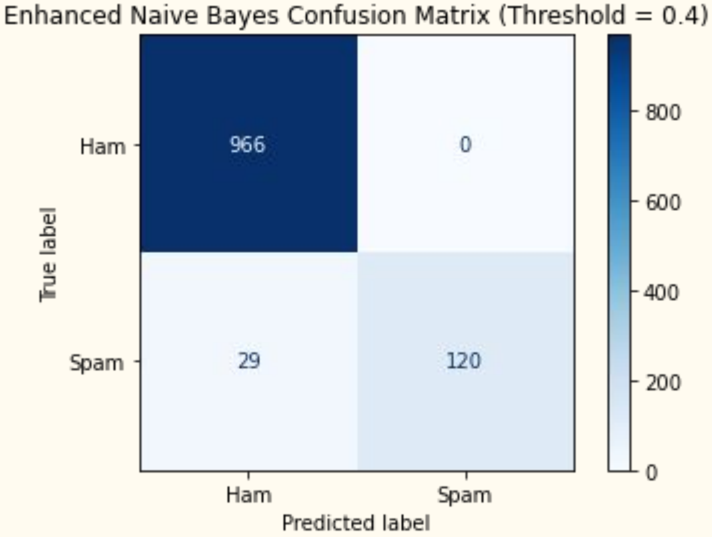
- 100%

Recall (Spam):

- 80.6%

F1 Score (Spam):

- 89.3%



Regex Features

Use Regex to detect phone numbers and URLS

Precision (Spam):

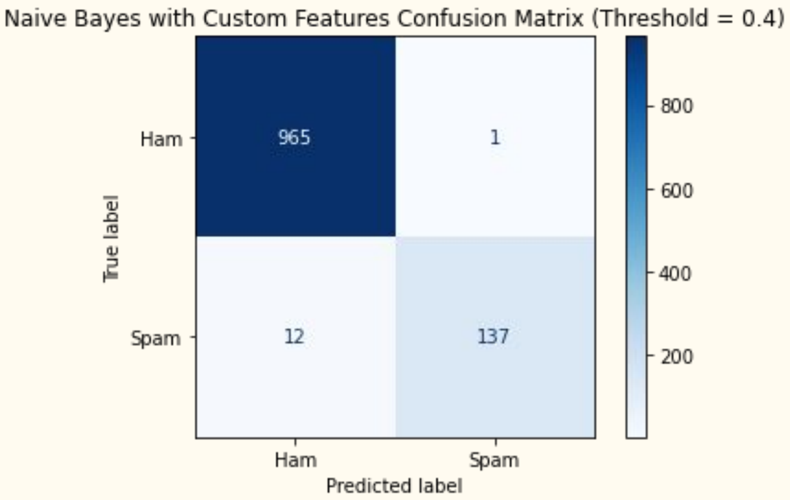
99.3%

Recall (Spam):

91.9%

F1 Score (Spam):

95.5%



Use Regex to detect pricing cues and short codes

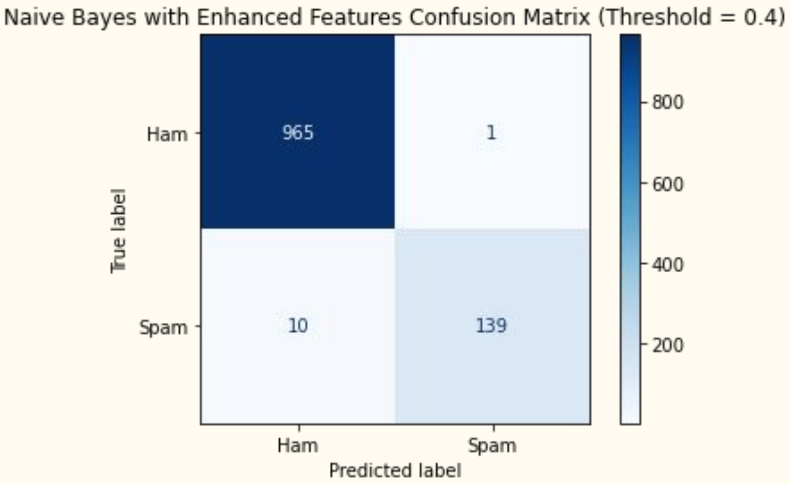
Precision (Spam):

99.3%

Recall (Spam):

93.3%

F1 Score (Spam):
96.2%



Business Recommendation

Use the tuned Multinominal Naive Bayes model to detect spam with an F1 Score of 96.2%

Next Steps

- Gather modern SMS data and retrain the model to keep the model accurate as spam patterns change
- Train the model for other messaging services, like email and IM