Samuel Nordale

Hafsa Khalid

Nicholas Paun

COMP 551 – Miniproject 2

**Abstract:**

The goal of the project was to develop a data model that could predict which subreddits a Reddit comment came from, after having learned on a dataset of Reddit comments and the subreddits they came from. Our most interesting findings were x, y, z.

**Introduction:**

There were three core tasks assigned in the project. First, we were required to implement a Bernoulli Naïve Bayes model from scratch. Second, we were to run experiments on the performance of several classification models from Sci-Kit Learn. Lastly, we needed to create a model validation pipeline and report on the differences between models.

The Reddit comment dataset that we received included 70,000 comments from 20 different subreddits, and the only fields defined in the table were the comments’ text, subreddits, and ID number.

To implement Bernoulli Naïve Bayes, we

**Related Work:**

**Dataset and Setup:**

The biggest factor when processing the dataset was the fact that the comments did not conform to a common standard for language, format, or punctuation. Some comments from have unique formatting for different types of information due to the standard of the specific subreddit. Some comments use links, emoticons, and emojis, which need to be accounted for.

**Proposed Approach:**

We implemented our own Multinomial Naïve Bayes model, and to compare our results with different classification models from Sci-Kit Learn, we chose to compare Logistic Regression, Multinomial Naïve Bayes, and Random Forest. Our feature design focused on creating arrays of keywords for subreddits. Then, by comparing each comment to the keywords, we could signal how likely it is that an individual comment is from a specific subreddit. Two other features we implemented were comment length, word count, and average sentence length. We hoped that this would help the model identify subreddits where users engaged in more long-form discourse.

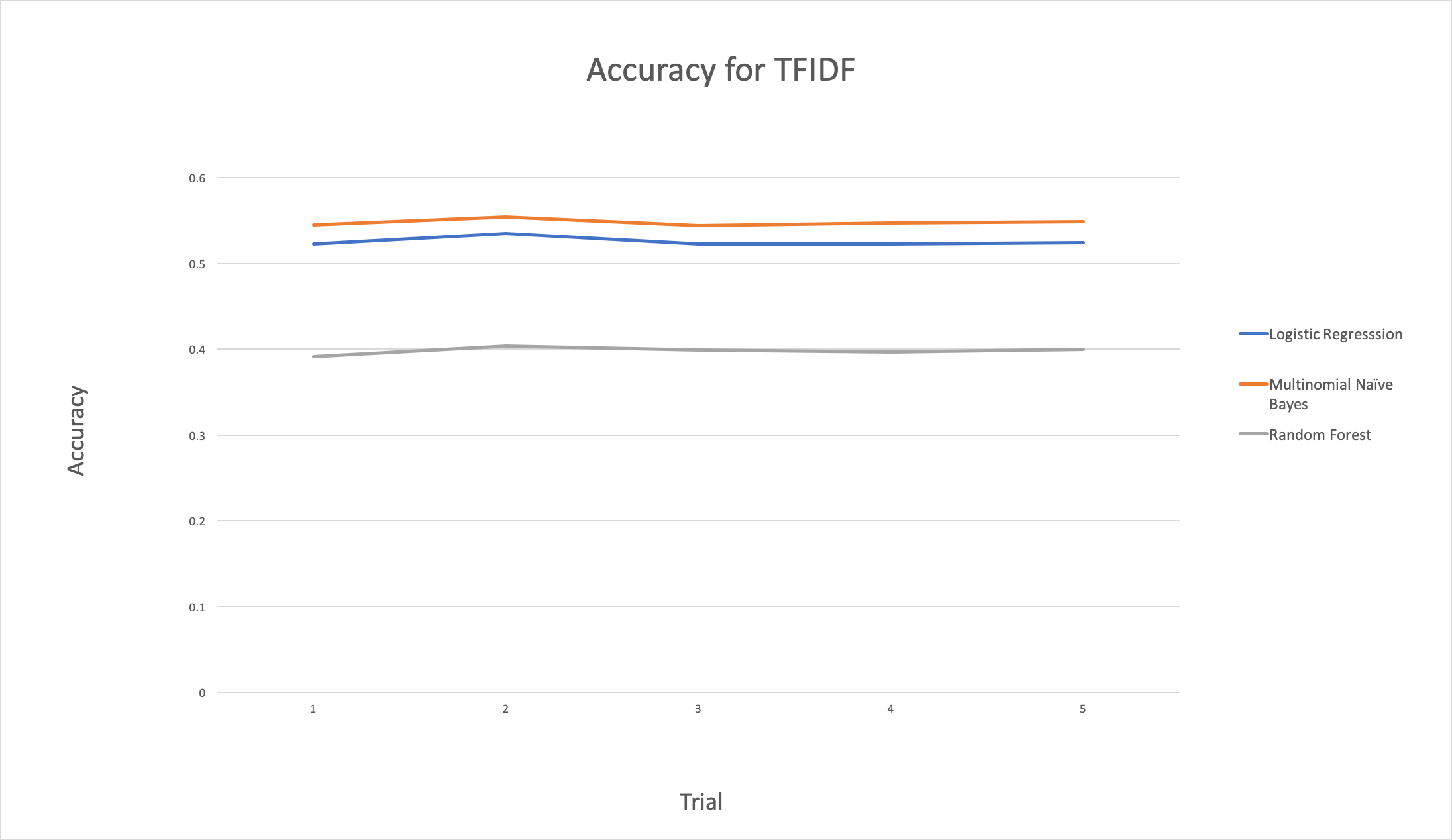
To implement Bernoulli Naïve Bayes,

**Results:**

As our baseline, we calculated the accuracies for Sci-Kit Learn’s TF-IDF and Count vectorizers. Using just the feature matrix given by the vectorizers, we calculated these accuracies:

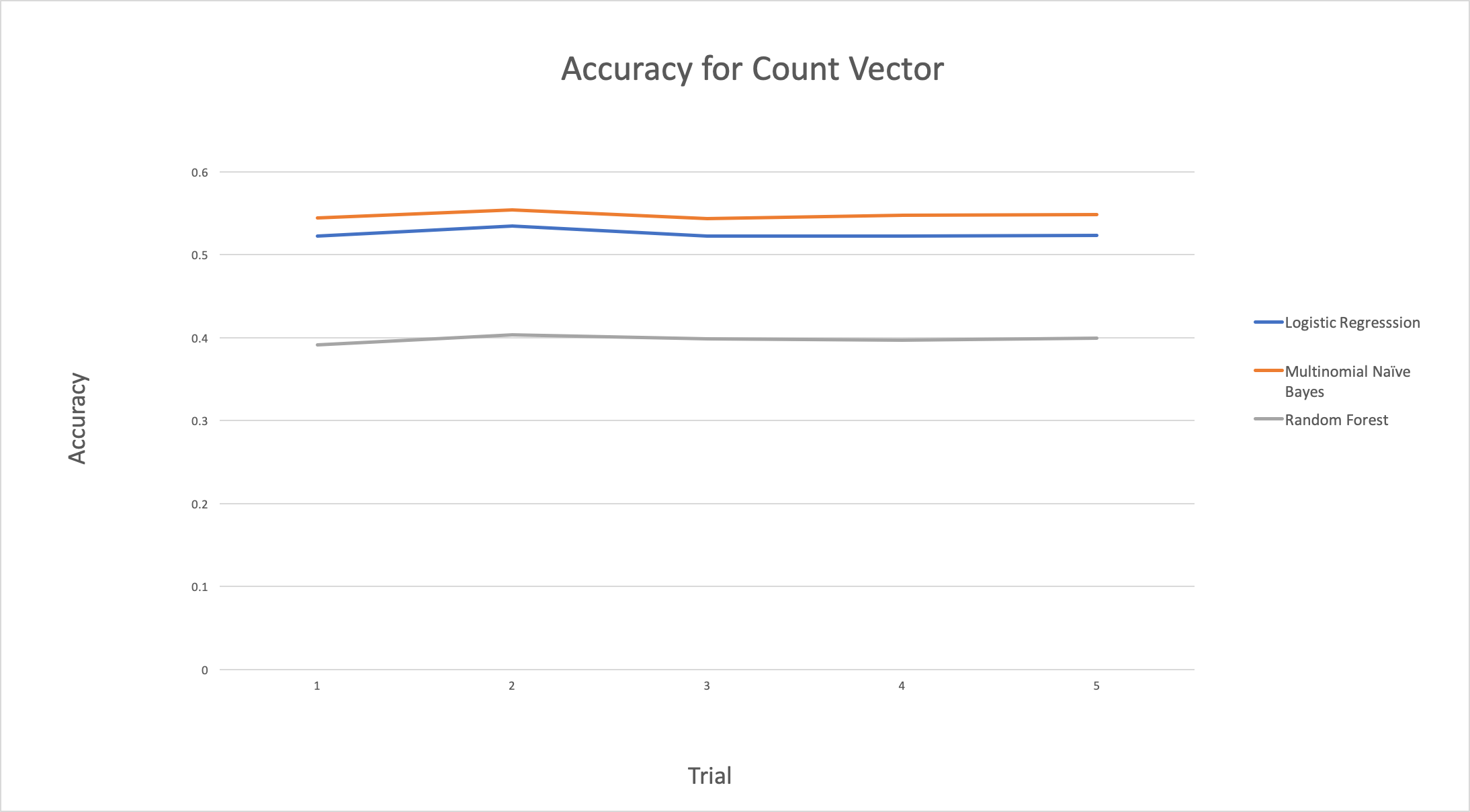
**Baseline Accuracies for Sci-Kit Learn Models**

|  |  |  |
| --- | --- | --- |
|  | TF-IDF | Count Vector |
| Logistic Regression | 54.80% | 52.53% |
| Mulitnomial Naïve Bayes | 56.18% | 54.77% |
| Random Forest | 39.95% | 39.80% |

****

*Accuracies for TFIDF Vectorizer from Sci-Kit Learn over 5 trials. This graph illustrates*

*how Logistic Regression and Multinomial NB outperformed Random Forest every trial.*

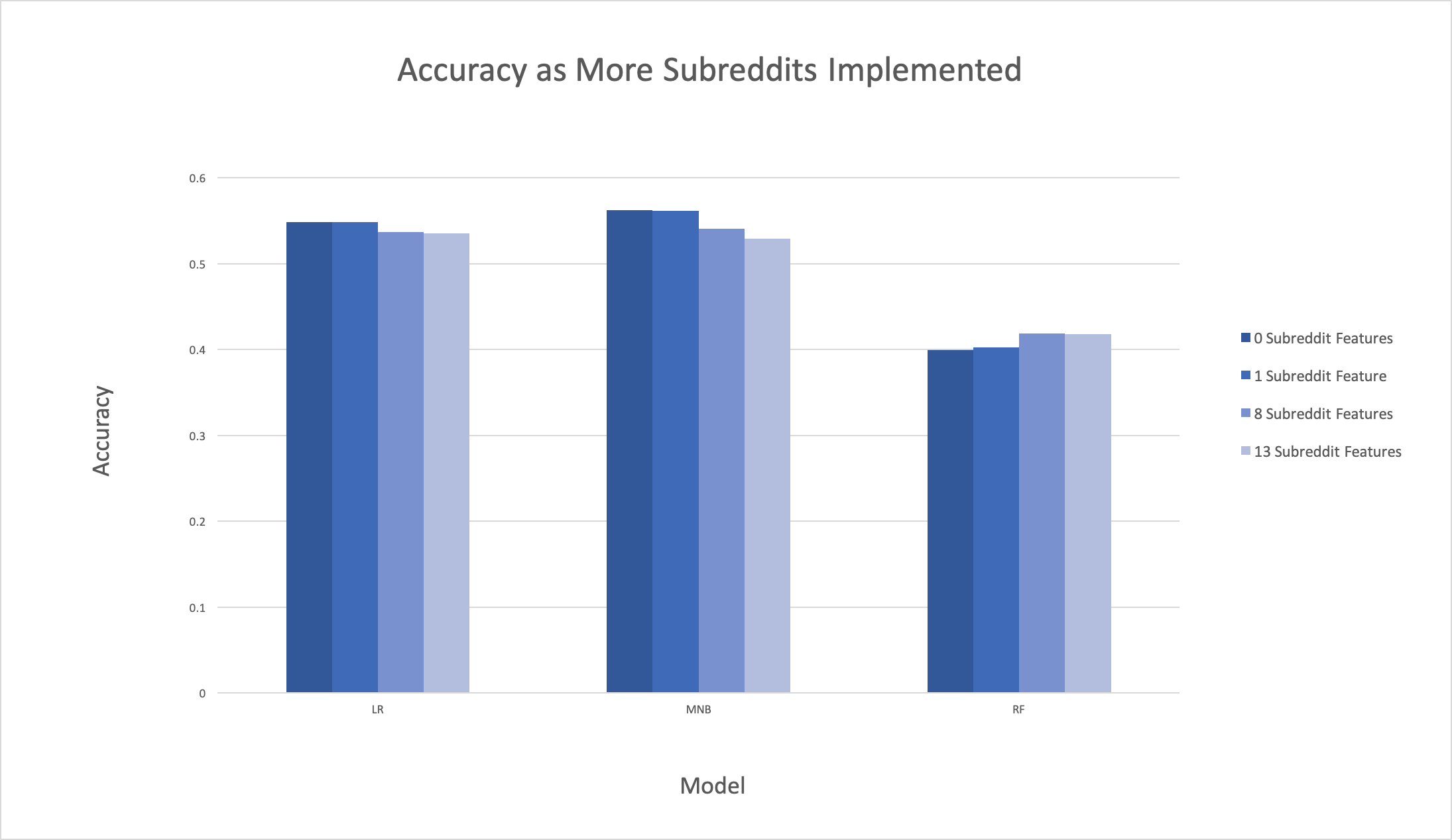
****

*Accuracies for Count Vector Vectorizer from Sci-Kit Learn over 5 trials. This graph illustrates*

*how Logistic Regression and Multinomial NB outperformed Random Forest every trial.*

As our data illustrates, Multinomial Naïve Bayes performed the best and Random Forest performed the worst on both features matrices. The TF-IDF feature matrix also had better accuracy for all three Sci-Kit Learn models.

As we implemented our keyword vocabularies, we measured the accuracy trend as we added vocabularies for more subreddits:



*Accuracies for each classification model when 0, 1, 8, and 13 subreddit specific vocabulary features are*

*implemented. Logistic Regression and Multinomial Naïve Bayes decreased as we implemented more features,*

*while Random Forest increased.*

**Discussion and Conclusion:**

**Statements of Contribution:**