

Understanding the Political Landscape in US

Data mining in Reddit posts



Content



- A. Overview and Agenda
- B. Exploratory Data Analysis
 - a. Feature Importance
 - b. Word Cloud
- C. Model Selection and Analysis
- D. Model Evaluation
- E. Conclusion

Overview



Given the popularity of social media, it is becoming a standard operating procedure for major political and business groups to extract insights of the public information.

In this project, we have performed a test to classify whether the posts are from the Conservatives or Democrats subreddits, using the Natural Language Processing (NLP) algorithms.

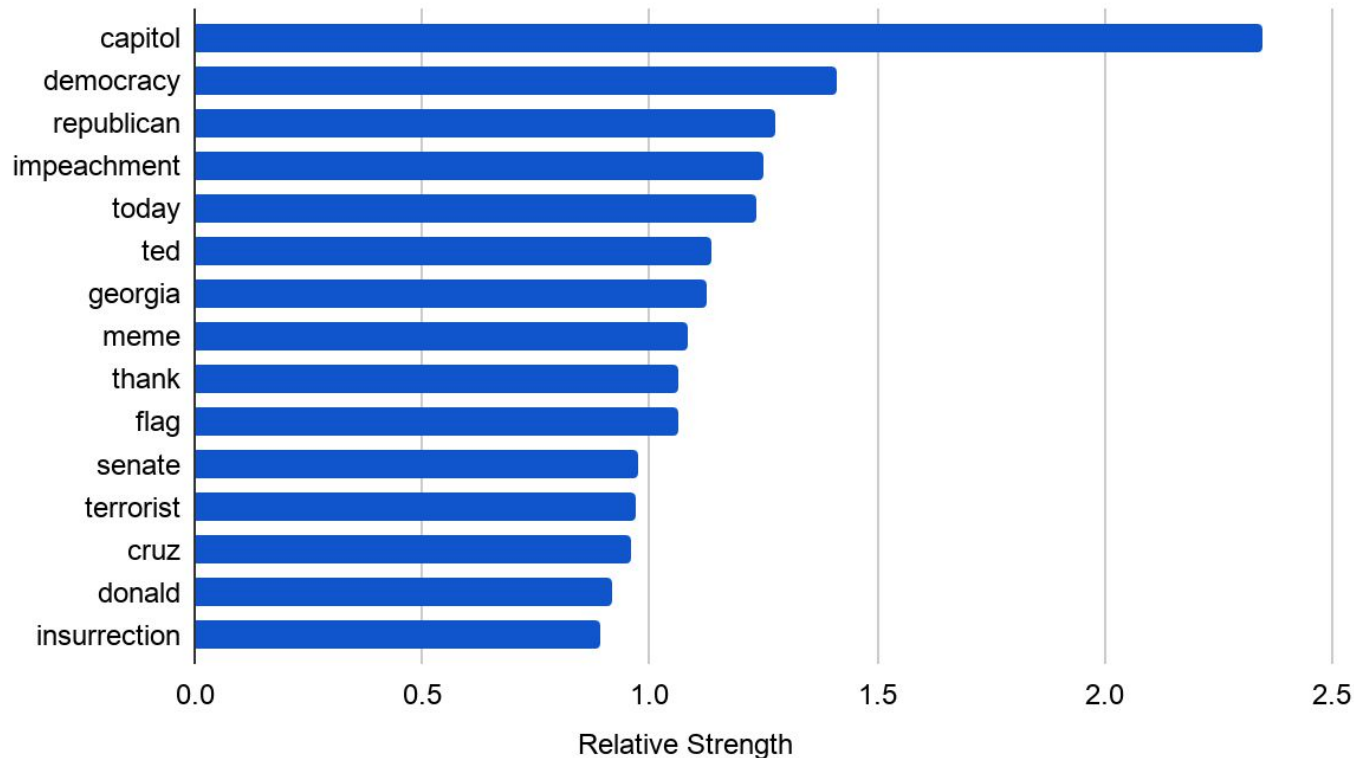
Out of a sample of ~2000 posts, we have built a classifier at an accuracy of [75%], versus a baseline accuracy of [50%], ie equally split between Democrats and Conservatives.

Based on the weightings of key words in a model, some insights are derived as to what could be viewed as important issues. This will be useful in driving the marketing strategy of political and business groups.

Exploratory Data Analysis - Feature Importance



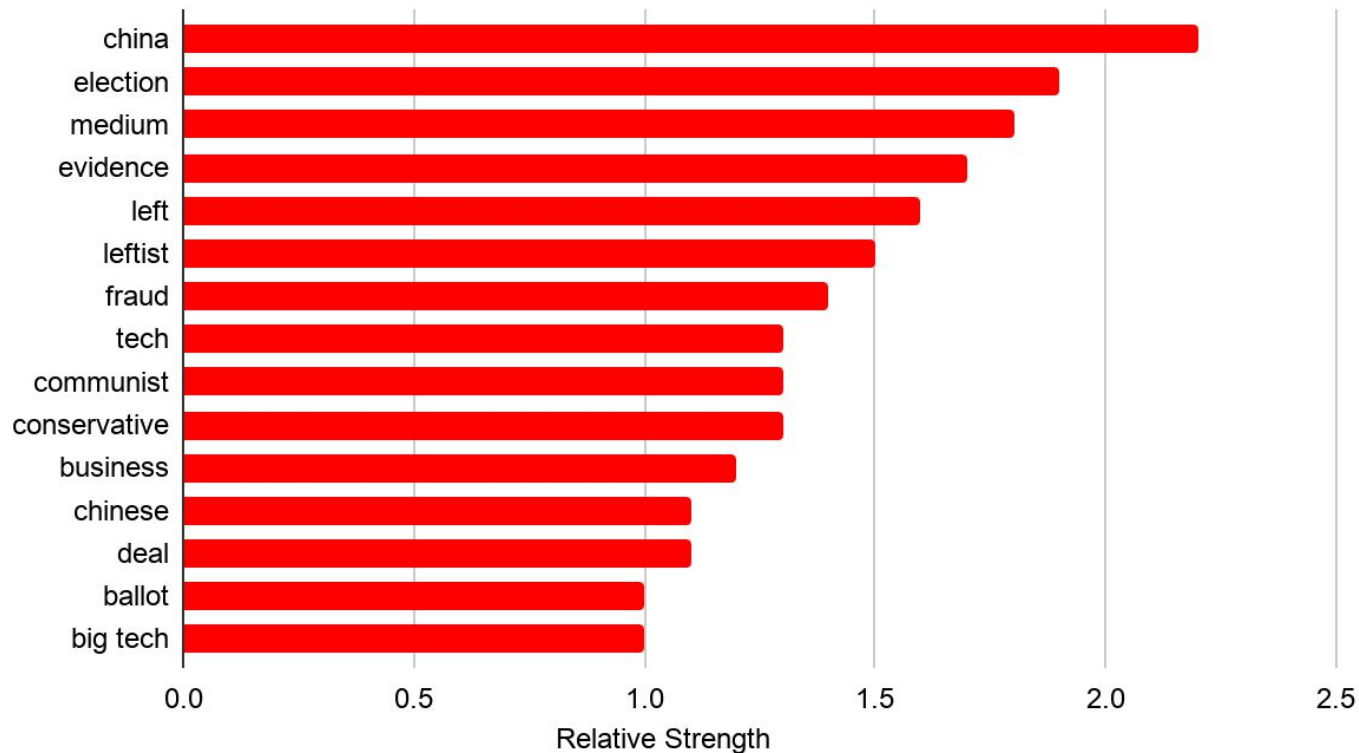
Importance of key words - Democrats



Exploratory Data Analysis - Feature Importance

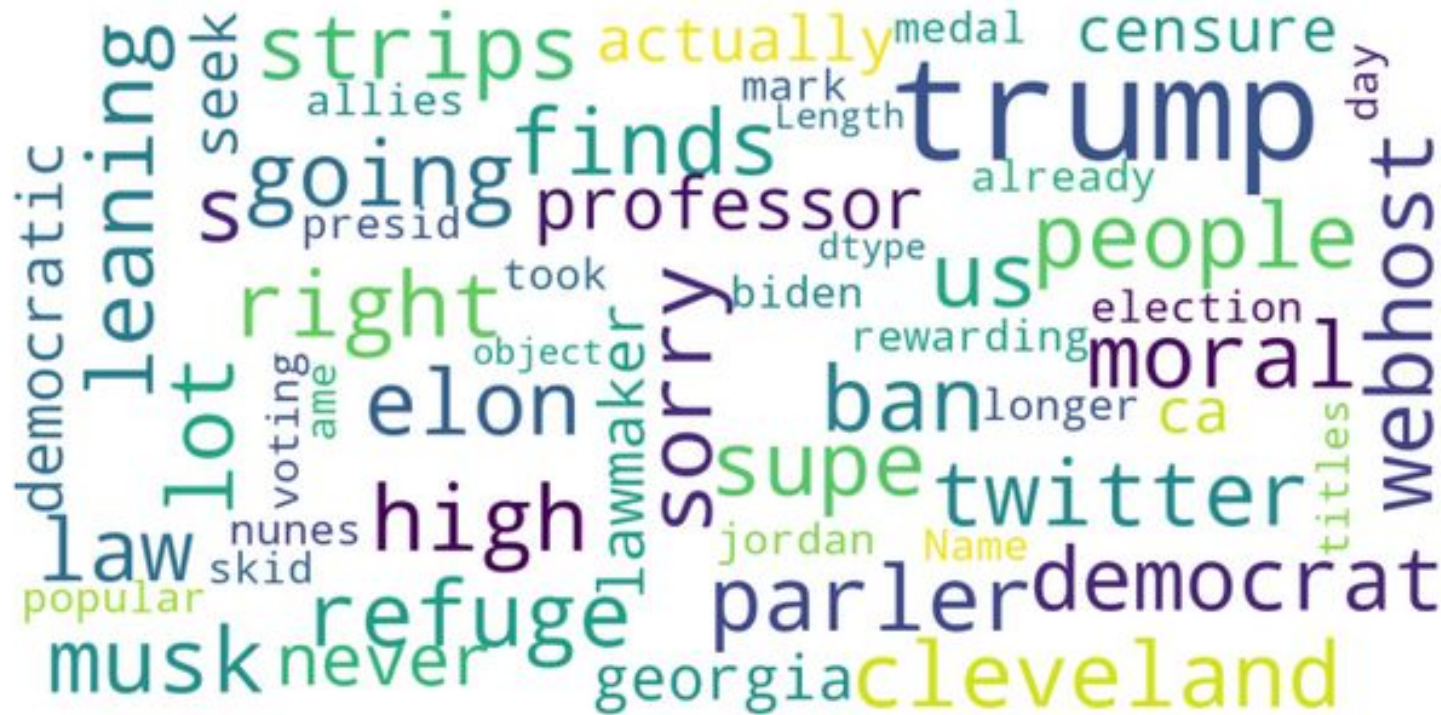


Importance of key words - Conservatives



Exploratory Data Analysis - Word Cloud

Frequency of words used in titles of both *r/Conservative* and *r/democrats*



Democrats Camp

Frequency of words used in titles of *r/democrats*



Model Analysis and Selection

Results of Pipeline 1 (Count Vectorizer, Logistic Regression)				
	Titles_Lemm	Titles_Stemm	Comments_Lemm	Comments_Stemm
Best accuracy score across folds	0.813	0.806	0.916	0.914
Accuracy score on Train Set	0.994	0.993	0.985	0.98
Accuracy score on Validation Set	0.842	0.813	0.926	0.927
Results of Pipeline 2 (Tf-idf Vectorizer, Multinomial Naive Bayes)				
	Titles_Lemm	Titles_Stemm	Comments_Lemm	Comments_Stemm
Best accuracy score across folds	0.806	0.813	0.893	0.887
Accuracy score on Train Set	0.961	0.955	0.955	0.952
Accuracy score on Validation Set	0.832	0.802	0.917	0.897
Results of Pipeline 3 (Count Vectorizer, Decision Tree Classifier)				
	Titles_Lemm	Titles_Stemm	Comments_Lemm	Comments_Stemm
Best accuracy score across folds	0.776	0.756	0.886	0.876
Accuracy score on Train Set	1.0	1.0	0.994	0.993
Accuracy score on Validation Set	0.813	0.775	0.876	0.893
Results of Pipeline 4 (Count Vectorizer, Random Forest Classifier)				
	Titles_Lemm	Titles_Stemm	Comments_Lemm	Comments_Stemm
Best accuracy score across folds	0.826	0.81	0.954	0.952
Accuracy score on Train Set	1.0	1.0	0.994	0.993
Accuracy score on Validation Set	0.816	0.797	0.972	0.964
Results of Pipeline 5 (Count Vectorizer, Support Vector Classifier)				
	Titles_Lemm	Titles_Stemm	Comments_Lemm	Comments_Stemm
Best accuracy score across folds	0.801	0.798	0.856	0.855
Accuracy score on Train Set	0.984	0.987	0.929	0.933
Accuracy score on Validation Set	0.821	0.797	0.889	0.887

- 5 different classification algorithms considered :
 - Logistic Regression,
 - Multinomial Naive Bayes,
 - Decision Tree Classifier,
 - Random Forest Classifier, and
 - Support Vector Classifier
- Evaluation metric:
 - Accuracy score
 - No imbalanced classes
 - Objective of the project makes us impartial towards either class
- Baseline Model
 - Default parameters

Model Analysis and Selection

Results of Pipeline 1 (Count Vectorizer, Logistic Regression)				
	Titles_Lemm	Titles_Stemm	Comments_Lemm	Comments_Stemm
Best accuracy score across folds	0.813	0.806	0.916	0.914
Accuracy score on Train Set	0.994	0.993	0.985	0.98
Accuracy score on Validation Set	0.842	0.813	0.926	0.927
Results of Pipeline 2 (Tf-idf Vectorizer, Multinomial Naive Bayes)				
	Titles_Lemm	Titles_Stemm	Comments_Lemm	Comments_Stemm
Best accuracy score across folds	0.806	0.813	0.893	0.887
Accuracy score on Train Set	0.961	0.955	0.955	0.952
Accuracy score on Validation Set	0.832	0.802	0.917	0.897
Results of Pipeline 3 (Count Vectorizer, Decision Tree Classifier)				
	Titles_Lemm	Titles_Stemm	Comments_Lemm	Comments_Stemm
Best accuracy score across folds	0.776	0.756	0.886	0.876
Accuracy score on Train Set	1.0	1.0	0.994	0.993
Accuracy score on Validation Set	0.813	0.775	0.876	0.893
Results of Pipeline 4 (Count Vectorizer, Random Forest Classifier)				
	Titles_Lemm	Titles_Stemm	Comments_Lemm	Comments_Stemm
Best accuracy score across folds	0.826	0.81	0.954	0.952
Accuracy score on Train Set	1.0	1.0	0.994	0.993
Accuracy score on Validation Set	0.816	0.797	0.972	0.964
Results of Pipeline 5 (Count Vectorizer, Support Vector Classifier)				
	Titles_Lemm	Titles_Stemm	Comments_Lemm	Comments_Stemm
Best accuracy score across folds	0.801	0.798	0.856	0.855
Accuracy score on Train Set	0.984	0.987	0.929	0.933
Accuracy score on Validation Set	0.821	0.797	0.889	0.887

- Generally across all models, training the model on comments did better than titles
 - Much more words in comments than in titles
 - For a post, there is a title but there could be >1,000 comments in the comment thread
- Between stemming and lemmatizing the words, the models are mixed in their results
 - Impartial toward either
 - Move forward with stemmed comments

Model Analysis and Selection

Results of Pipeline 1 (Count Vectorizer, Logistic Regression)				
	Titles_Lemm	Titles_Stemm	Comments_Lemm	Comments_Stemm
Best accuracy score across folds	0.813	0.806	0.916	0.914
Accuracy score on Train Set	0.994	0.993	0.985	0.98
Accuracy score on Validation Set	0.842	0.813	0.926	0.927
Results of Pipeline 2 (Tf-idf Vectorizer, Multinomial Naive Bayes)				
	Titles_Lemm	Titles_Stemm	Comments_Lemm	Comments_Stemm
Best accuracy score across folds	0.806	0.813	0.893	0.887
Accuracy score on Train Set	0.961	0.955	0.955	0.952
Accuracy score on Validation Set	0.832	0.802	0.917	0.897
Results of Pipeline 3 (Count Vectorizer, Decision Tree Classifier)				
	Titles_Lemm	Titles_Stemm	Comments_Lemm	Comments_Stemm
Best accuracy score across folds	0.776	0.756	0.886	0.876
Accuracy score on Train Set	1.0	1.0	0.994	0.993
Accuracy score on Validation Set	0.813	0.775	0.876	0.893
Results of Pipeline 4 (Count Vectorizer, Random Forest Classifier)				
	Titles_Lemm	Titles_Stemm	Comments_Lemm	Comments_Stemm
Best accuracy score across folds	0.826	0.81	0.954	0.952
Accuracy score on Train Set	1.0	1.0	0.994	0.993
Accuracy score on Validation Set	0.816	0.797	0.972	0.964
Results of Pipeline 5 (Count Vectorizer, Support Vector Classifier)				
	Titles_Lemm	Titles_Stemm	Comments_Lemm	Comments_Stemm
Best accuracy score across folds	0.801	0.798	0.856	0.855
Accuracy score on Train Set	0.984	0.987	0.929	0.933
Accuracy score on Validation Set	0.821	0.797	0.889	0.887

- Based on accuracy score on train set, Decision Tree Classifier and Random Forest Classifier did the best
- Compared with score on validation set, the Random Forest Classifier generalized better on unseen data.

Selected Model:

Random Forest Classifier

Model Evaluation



- Performed a grid search to find best parameters

The best parameters are : {'cvec__min_df': 2, 'cvec__ngram_range': (1, 2), 'cvec__stop_words': None, 'rf__max_depth': None}

- Removing stop words did not improve the model, likely because stop words help with the context of content in each subreddit
- Unigrams and bigrams perform better than using just unigrams for a similar reason

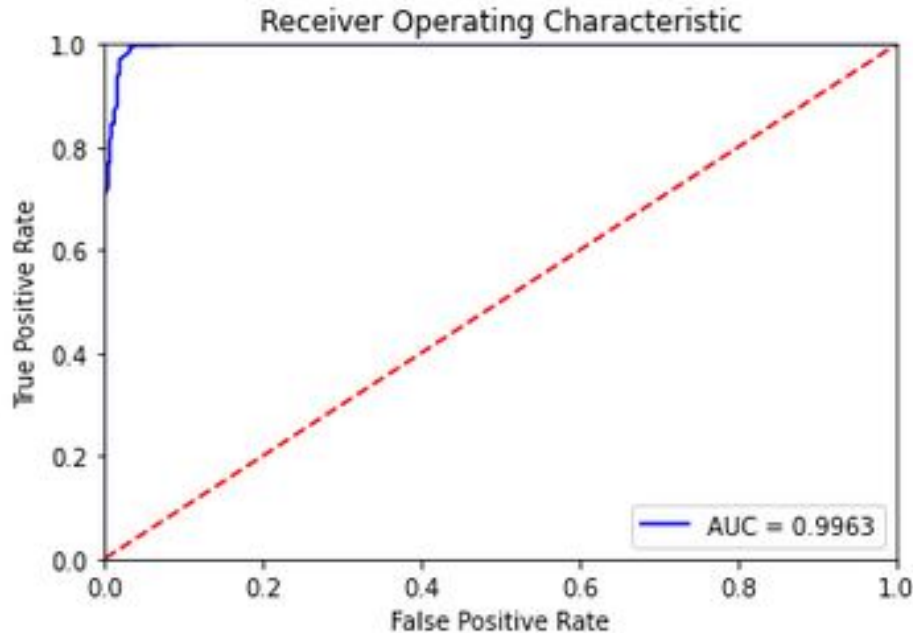
Accuracy Score with Best Parameters on Training, Validation and Test Set	
Dataset	Accuracy Score
Accuracy score on Training Set	0.994
Accuracy score on Validation Set	0.985
Accuracy score on Test Set	0.979

- Performed better than the baseline model
- Model is able to account for **97.9%** of variability of data

Model Evaluation



ROC-AUC curve



- Represents degree or measure of separability
- With a relatively high performing model, an AUC close to 1 is expected

Conclusion



- Production model did very well to classify a post into *r/Conservative* and *r/democrats*.
- For the same topic, content that is discussed in each subreddit goes in very different directions (ie. redditors use very distinct words for each subreddit)
- From wordcloud and feature importance, it is clear that there are no common repeats (except “trump”) between both subreddits.

Thank You

Questions