NLP with South Park

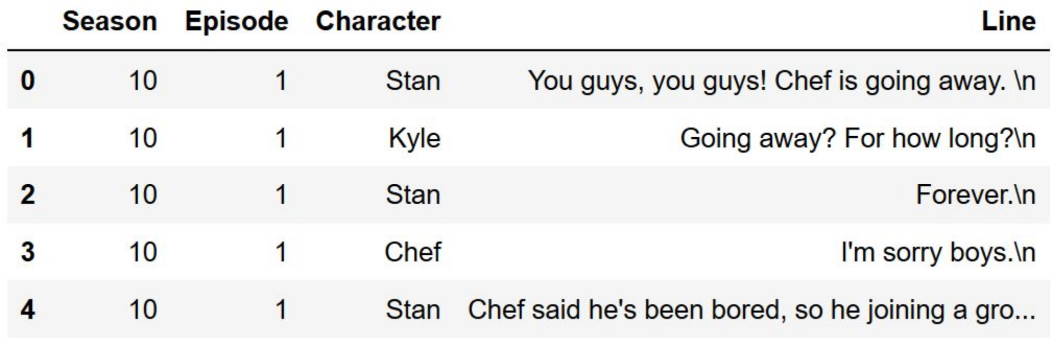
Taylor Willingham

Objective and Overview

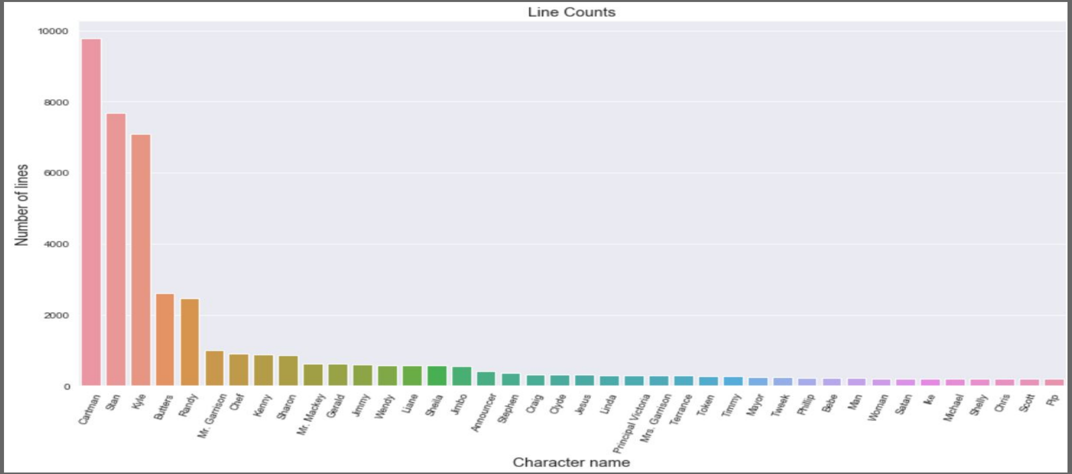
Can we combine Natural Language Processing (NLP) techniques and supervised machine learning to build a classifier to predict which lines of dialogue belong to Cartman?

The Data

- Each observation is one line of dialogue, identified by the character speaking.
- Cartman has the most lines, as illustrated in the bar chart to the right.



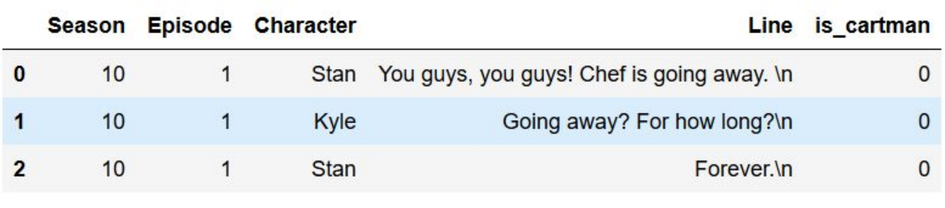
	Season	Episode	Character	Line
0	10	1	Stan	You guys, you guys! Chef is going away. \n
1	10	1	Kyle	Going away? For how long?\n
2	10	1	Stan	Forever.\n
3	10	1	Chef	I'm sorry boys.\n
4	10	1	Stan	Chef said he's been bored, so he joining a gro...



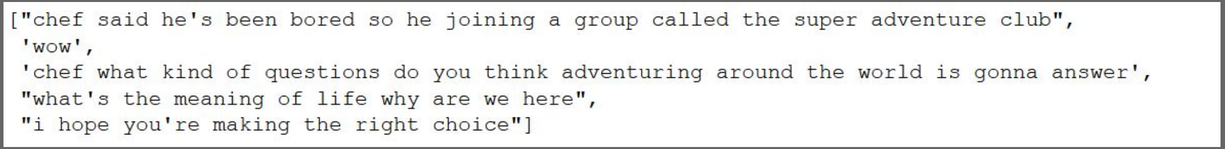
Preprocessing the text

Here are the steps for preprocessing:

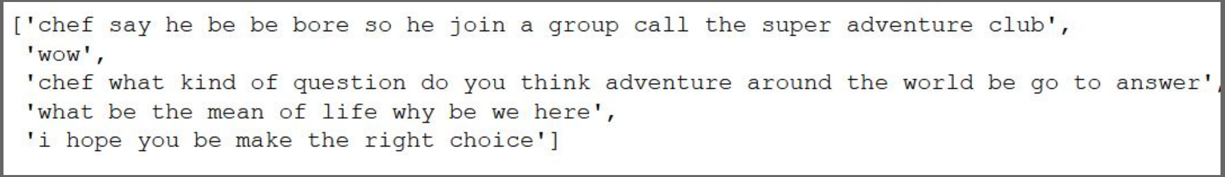
- Compile all strings, or lines of dialogue into one list
- Remove the new-line figure (\n) from each line
- Convert each word to lowercase and remove punctuation
- Expand all contractions to their whole words
- Lemmatize all nouns and verbs so that only root words are present



	Season	Episode	Character	Line	is_cartman
0	10	1	Stan	You guys, you guys! Chef is going away. \n	0
1	10	1	Kyle	Going away? For how long?\n	0
2	10	1	Stan	Forever.\n	0



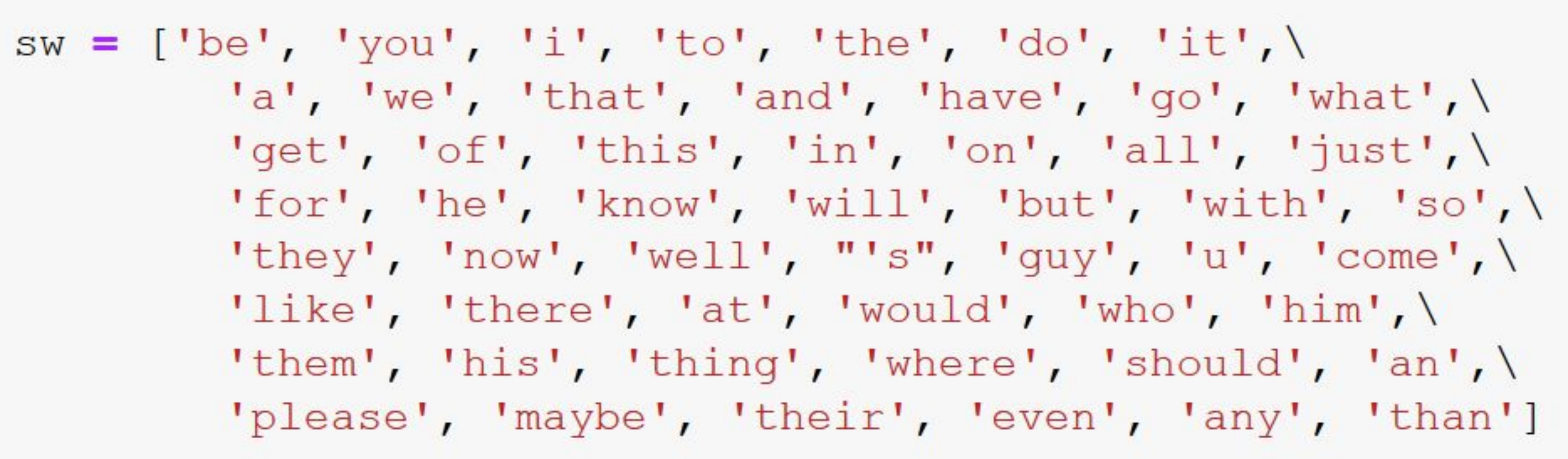
```
["chef said he's been bored so he joining a group called the super adventure club",
 'wow',
 'chef what kind of questions do you think adventuring around the world is gonna answer',
 "what's the meaning of life why are we here",
 "i hope you're making the right choice"]
```



```
['chef say he be be bore so he join a group call the super adventure club',
 'wow',
 'chef what kind of question do you think adventure around the world be go to answer',
 'what be the mean of life why be we here',
 'i hope you be make the right choice']
```

Stop Words

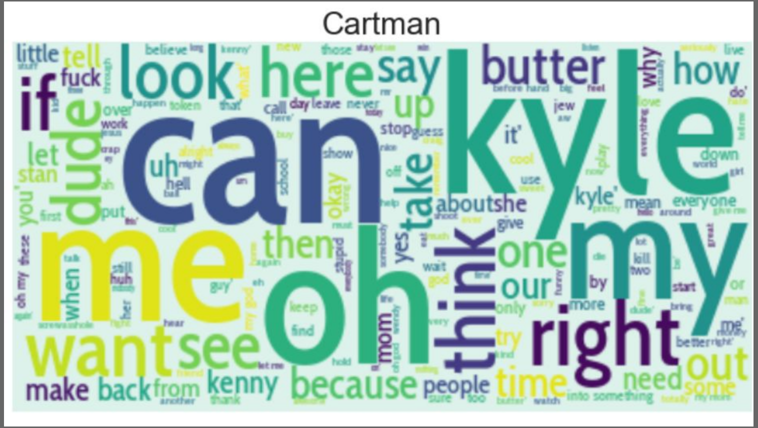
Examining words that occur frequently to assess potential stop words.

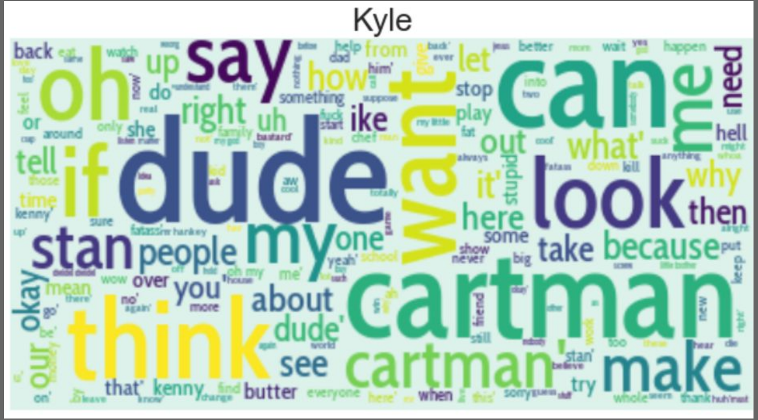


```
sw = ['be', 'you', 'i', 'to', 'the', 'do', 'it',\
        'a', 'we', 'that', 'and', 'have', 'go', 'what',\
        'get', 'of', 'this', 'in', 'on', 'all', 'just',\
        'for', 'he', 'know', 'will', 'but', 'with', 'so',\
        'they', 'now', 'well', "'s", 'guy', 'u', 'come',\
        'like', 'there', 'at', 'would', 'who', 'him',\
        'them', 'his', 'thing', 'where', 'should', 'an',\
        'please', 'maybe', 'their', 'even', 'any', 'than']
```

Word Clouds

Word clouds can be used to visualize common words.


Cartman


Kyle

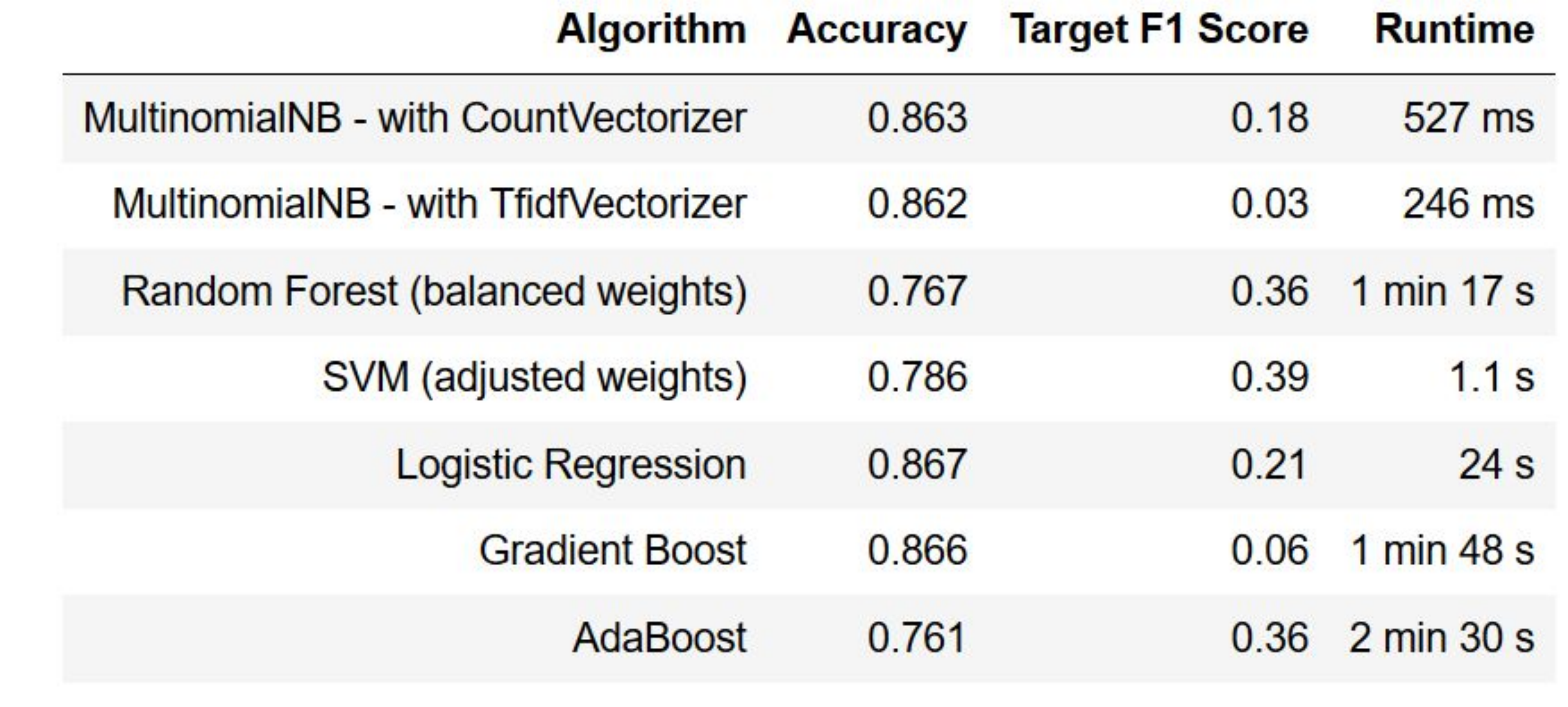
Vectorizers

- To convert the word corpus into mathematical vectors, I used CountVectorizer and TfidfVectorizer from Scikit Learn.
- CountVectorizer is essentially a bag-of-words, where documents are assessed on simple word frequency.
- Tf-idf creates vectors of word weights based on term frequency and document frequency.
- Using MultinomialNB algorithm, the accuracy with CountVectorizer was 0.863, and was 0.862 with tf-idf.

Other Algorithms

- Besides MultinomialNB, I also tried Random Forest, SVM, Logistic Regression and variants of boosting.
- The accuracy scores were similar, but the f1-scores varied quite a bit.
- After adjusting for class weights, SVM seemed to perform the best based on a combination of precision, recall and computational expense.

Final Results



Algorithm	Accuracy	Target F1 Score	Runtime
MultinomialNB - with CountVectorizer	0.863	0.18	527 ms
MultinomialNB - with TfidfVectorizer	0.862	0.03	246 ms
Random Forest (balanced weights)	0.767	0.36	1 min 17 s
SVM (adjusted weights)	0.786	0.39	1.1 s
Logistic Regression	0.867	0.21	24 s
Gradient Boost	0.866	0.06	1 min 48 s
AdaBoost	0.761	0.36	2 min 30 s

Further Thoughts

- As for why the models weren't successful, I think it likely has to do with:
  - The nature of the data - Only two people, Trey Parker and Matt Stone, provide voices for most characters, which might create hidden similarities.
  - The structure of the data - The documents are short, making them hard to differentiate and predict.
- To try and improve the results, there are a few next steps we might attempt:
  - Test thresholds for document length
  - Use a more complex model such as deep learning
  - Narrow the focus by omitting unimportant characters that might dilute the data