At time of submission, this was how I ranked. My username is AhmedA

47	**Ahmed**	**0.66260**	**4**	**Sat, 19 Sep 2015 00:47:47**

Now, obviously I failed to make significant progress achieving only 3-4% increase on the baseline. I tackled this assignment on two fronts, one was sanitizing and making sense of the data and the other theorizing what the best classifiers would be (making heavy use of the TfidVectorizer).

To sanitize the data, I noticed I was getting a lot of symbols and numbers in my top results so I wrote a function to pass into the preprocessor to remove these from the text. I also stripped the accents and smoothed the data with smooth_idf.
I also lemmatized using Wornet from the nltk library.

```
def removepunc(text):
punctuation = re.compile(r'[,.?!\':\)\(`";|0-9]')
text = punctuation.sub('', text)
return text
```

Now to pick the relevant data, I relied on the built-in english stopwords as a feature as well as using max_df=0.5 to ignore words that are used extremely frequently. I then had a bag of words setup, with a focus on groups of 1-3 words as features (my ngram).

Due to limited experience in Python and limited time, I hadn't been able to implement my attempts at further feature extraction. I attempted to add two classifiers to the data, the first was comparing the words with a list of dramatic words 'category drama dictionary' and weighing them more heavily (spoilers are usually suspenseful is the guiding assumption here). The second was incorporating the year and genre as features, due to repetitive nature of the tropes correlating with both year and genre.