Abstract: This project involved development of a supervised classification model to predict on the test dataset, which contains 20k comments from different communities on Reddit platform.
The goal was to provide the most accurate predictions of the communities for each test comment. There should be text pre-processing to clean the comments. There are many characters and symbols in the text of comments (especially social media comments).
The classifiers accept numbers and matrices, therefore the cleaned text must be processed and transformed into shapes that classifiers accept.
Bernoulli Naive bayes is the classifiers mentioned by instruction to be used. Also two more classifiers to be added in order to compare the results. As a competition, the best achieved result should be submitted in Kaggle platform to be compared with other students.

Introduction: The training data contained 2 main rows (comments, subreddit) and 60k columns. each column provided a sample comment of the related subreddit.
There was no null value and the considered rows have object data types. The comments row of the train data set has passed to the cleaning function which process the text by different functions.
After converting the text to lower case, we provide a pattern to combine a regular expression pattern into pattern objects. Then we replace the passed pattern with space (" ") in the text. In the next step we replace some of the most misspelled words with the correct in order to provide better vocabulary.
After the cleaning process we pass the text to the tokenizer to prepare a table for the stripping the vocabulary. The strip function eleminates some alphabets (up to 3) from the end of each word in order to bring the vocabulary close to their root. Then by using the stop-words function we eliminate all words which are possible not in English dictionary.
Finally we append all the cleaned words in the same order in the empty list which we prepared before, and the train data-set has been cleaned and prepared.
As a comparison with the strip function, we use the lemmatizer and we look at the most frequent used words in our text.
There are some types of vectorizers available, DicVectorizer, CountVectorizer, but for the case of this project the TfidfVectorizer provided the best result after trying the other possible vectorizers. This vectorizer convert the

words to feature vectors in a matrix. For the next step we apply the values of the parameters on the actual data and gives the normalized value by transforming the text, and here the data is ready for prediction.
By using the train-test-split function from sklearn, we split the data into train and test data-sets.
Bernoulli Naive Bayes classifiers is mentioned to be used in this project.

Related work:

Dataset and setup:

Proposed approach:

Results:

Discussion and Conclusion:

Statement of Contributions: