COL774 Assignment 2

Aniruddha Deb
2020CS10869

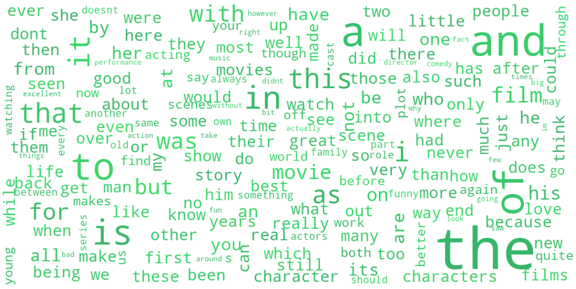
October 2022

1 Naive Bayes

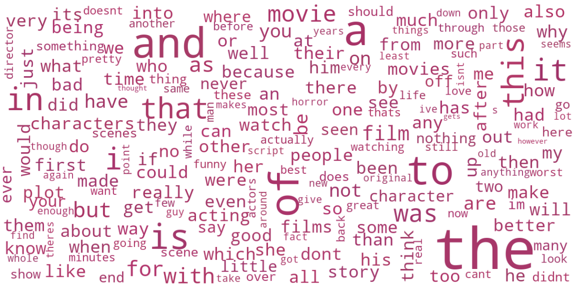
(a) Naive Bayes was implemented using the **Multinomial Event Model** as discussed in class, which uses the frequency of the words rather than just their presence. The vocabulary was learnt from the training dataset, and any words not in the vocabulary in the test dataset were ignored. With this implementation, the following results were obtained:

(a) An accuracy of 79.313% on the test dataset, with 7501/10000 positive examples and 4396/5000 negative examples correctly classified

(b) The following word clouds were obtained. Note that there are a lot of stopwords, which we remove in part (d)



Positive reviews



Negative reviews

(b) With Random Guessing, we'd obtain a test set accuracy of approximately 50%. By simply predicting each sample as positive, this would jump to 66.6%, as the number of positive reviews in the test set is twice the number of negative reviews.

Our Algorithm gives a 30% increase in accuracy over random guessing and a 14% increase in accuracy over simply predicting each review as positive.

(c) The confusion matrices are as follows:

NaiveBayes

	AP	AN
PP	7501	604
PN	2499	4396

Random

	AP	AN
PP	5000	2500
PN	5000	2500

AllPositive

	AP	AN
PP	10000	5000
PN	0	0

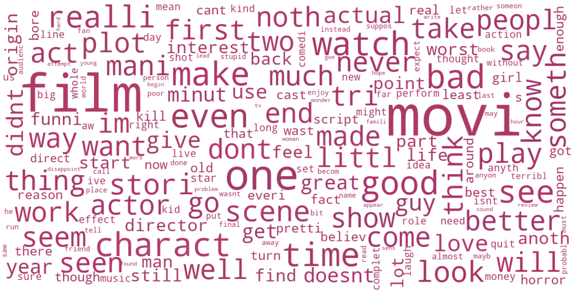
TODO part 2 of this: wdym "highest value of diagonal entry"?

The pattern is that the column sums for the actuals always add up to the number of examples of that category in the training set. This is because the actual column measures the number of actual examples in the dataset, and across all predictions, would sum to the total number of that category in the dataset.

(d) Removing stopwords and stemming gives us an accuracy of **80.4%**, with 7678/10000 positive reviews and 4388/5000 negative reviews correctly classified.

movi
film
one
good
stori
watch
time
charact
love
perform
great
scene
show
realli
see
make
think

Positive reviews



Negative reviews

The word clouds show the frequency of the various word stems rather than the frequency of the words themselves.

(e) Using Bigrams (in addition to removing stopwords and stemming) gives us an accuracy of **84.4 %**, with 8209/10000 positive reviews and 4447/5000 negative reviews correctly classified.

For the additional feature, we use Trigrams. Using trigrams, in addition to bigrams and removing stopwords + stemming gives us an accuracy of **85.2 %**, with 8352/10000 positive reviews and 4423/5000 negative reviews correctly classified.

The additional sets of features do help us improve our accuracy (explain why)

(f) The best-performing (most accurate) model is the Trigram model, which has a precision of $8352/(8352+577) = 0.935$, a recall of $8352/(8352+1648) = 0.835$ and an F1 score of $2(0.935 \times 0.835)/(0.935 + 0.835) = 0.882$

The F1 score is a better metric for this dataset, as there are twice as many positive movie reviews in the test dataset compared to negative reviews. In this case, we can see significant accuracy gains by simply declaring everything as positive instead of proceeding class by class. The F1 score helps to prevent this from happening, and gives a more realistic representation of how accurate our model is, across both classes irrespective of how many samples there are in each class.

2 Binary SVM

3 Multiclass SVM