Greg Matthews
CS 510
11/20/17

CS 510: Naïve Bayes Classification

Reflection Questions (3 Classifier Failures):

1.) “If only it had been the final movie as well.”

- The naïve Bayes classifier output a false positive for this review, which I believe is attributed to the classifier not being able to identify the context of the sentence. The use of “final” sounds like a positive comment under the right context, however because the naïve Bayes classifier is examining words individually, it loses the context of the message and makes if difficult for the naïve Bayes approach to classify correctly. The word “final” will more prominently fall under positive review words as well, and therefore the weight of such a positive word made it hard for the naïve Bayes classifier to classify correctly.

2.) “Adventure, Religion, Nazis being vaporized, Serpants, cryptology...This movie inspired me on SO many levels.”

- The naïve Bayes classifier output a false negative for this review, and I believe this is because of the amount of “bad” words in the sentence, such as Nazis, vaporized, and serpent. You normally wouldn’t use Nazis, vaporized, or serpent to tell someone whether you liked a movie or not, those are filler words that are used to describe the movie but shouldn’t be used as a basis for identifying if someone likes a movie or not by using those words. The naïve Bayes approach failed because it weighed the probability of someone liking a movie using descriptive words in context with the movie, but not the overall review.

3.) “I didn't think this movie was very good because the ending ruined the whole movie. The special effects were good but the way that Kevin Bacon dies like three times isn't very pleasing.”

- The naïve Bayes classifier output a false positive for this review, and this really has to do with understanding the context of the review. Since the naïve Bayes classifier is only using unigram classification, it will see words such as “good”, and “pleasing” and register it as a positive review as these words are more predominately found in positive reviews. Using sentences like “not very good”, or “isn’t very pleasing” will dramatically make a unigram classifier produce an erroneous result because you are making a review by using negated phrases. Unigram classifiers can’t handle negated phrases because they take 2+ words to describe something, like “not good” or “isn’t amazing”, and will label all words as separate classifications. Therefore, a bigram classifier at this point would be necessary to solve the problem.

Explanation of features (bayesbest.py model):

The biggest problem with the naïve Bayes classifier was not being able to handle negated phrases like "not good" or "isn't terrible" because the program would individually classify the words as separate features, even though they need to be linked to together to understand the context of the sentence. Therefore, instead of only using unigram features, I implemented bigram features as well to allow the classifier to understand the context of 2-word structures.

This was accomplished by adding additional 2-word features of all related words that appear next to each other. This allows for both 1 word and 2-word classification, solving the issues with negated phrases. For example, say we have the review: "I really didn't like the movie, it wasn't good." We can see that words such as "like" and "good" will predominately be weighed as a positive word, with less negative word weight as "didn't" and "wasn't" could be both positive or negative. Using bigrams, well also be able to take into account "didn't like" and "wasn't good" which will both predominately be weighed as negative words, and thus helping the classifier output more accurately.

Evaluation of Original Bayes vs. Optimized Bayes Classifier:

For testing both the original and optimized Bayes classifiers, I used 100 positive reviews and 100 negative reviews for testing, and used the rest of the files (11,129 positive and 2,735 negative reviews) for training the classifier.

Note that for classifying a neutral review, I used a threshold value of 0.0025, whereby if the positive and negative review posterior probabilities are within 2*alpha of each other, it will output neutral.

Table 1: Results of Original vs Optimized Bayes Classifier

	Positive	**Neutral**	**Negative**
Original Bayes Classifier	97	6	97
Optimized Bayes Classifier	94	6	100

Table 2: Evaluation of Original vs Optimized Bayes Classifier

	Precision	**Recall**	**F1 Measure**
Original Bayes Classifier	0.927	0.937	0.932
Optimized Bayes Classifier	0.946	0.937	0.942

From the gathered data, we can immediately see that the precision has improved from 0.927 to 0.946 in correctly identifying positive reviews given the features, while the recall hasn't benefitted much from the addition of bigrams. We can see that the precision has increased because we have reduced the number of false positives in our results by including bigram features. This is because the bigrams have been able to add some context to negated phrases, labeling a phrase like "not good" as a more negatively weighed word, whereas before using unigram we would have “not” and “good”, where good would be a heavily weighed positive word.

There isn’t much change shown for recall because the false negatives shouldn’t have been affected much by the inclusion of bigrams, which were predominately used to reduce the number of false positives. We see more use in negated phrases such as “wasn’t good” vs. “wasn’t bad”, therefore the use of bigrams wouldn’t help much in reducing false negatives.

Because of the precision increasing for the optimized Bayes classifier, we directly see an improvement in the F1 measure, where the original Bayes classifier had an F1 measure of 0.932, and now the Bayes Classifier including bigrams has an F1 measure of 0.942.

In conclusion, we can see that the original Bayes classifier performed poorly due to the classifier individually labeling each word as its own feature, which causes problems for many different negated phrases, which is used considerably in American culture. As a result, I have included bigram features to the new Bayes classifier model, so it will take into account negated phrases of 2-word length, which has shown to reduce the number of false positives from happening and directly increasing the precision of the classifier.

Future Extension of Bayes Classifier:

Including bigrams as features is just one of many other optimizations that can be used to increase the performance of the classifier. Some other ways to improve performance include:

1.) Eliminating low frequency features
    - Because the classifier has to weigh every single word, or combination of words if using bigrams, the low frequency features can lead to negative performance if there’s a considerably amount of low frequency features. A better approach is to select the top n features for the classifier to choose, that way the higher frequency features will be weighed more as well as increase the run time of the system due to less low frequency features.
2.) Eliminate words that don’t provide any information gain
    - Common words such as “and, the, I, if” don’t give you a lot of information about whether a review is positive or negative, if will frequently be found in both types of reviews. Other words such as names of actors, factual descriptions of movies, and locations also don’t provide insight on whether a review if positive or negative. All these non-informative words can be recursively grown and stored into a list that can be continuously checked whether it is a member, and if it is, don’t count it as a feature.
3.) Go beyond bigrams and use trigrams, or n-grams.
    - Using bigrams solved some of the negated phrase issues, and allowed for more negated phrases to be classified correctly. However, there are phrases that could take 3 or more words to get the context across, such as “not very good”, “isn’t really interesting”, and “would never really recommend.” The optimal Bayes classifier should also experiment with trigrams and beyond to increase correct classification of words that take multiple words to comprehend a positive or negative review. This can be done much the same ways as I did with bigrams, but instead of separating a sentence into coupled two-word pairs of nearest words, I would do this for 3 words, or 4 words, etc.