Movie reviews

Machine Learning

Hanna Nurska, 499866

May, 2022

I affirm that this report is the result of my own work and that I did not share any part of it with anyone else except the teacher.

1 Introduction

The main goal of this activity was to analyze the movie reviews written by the users of IMDB website and try to predict if the review is positive or negative on the base of its content. For this task, multinomial Naive Bayes classifier was used.

2 Organization of data

The dataset includes 25 000 movie reviews with label *positive* and 25 000 movie reviews with label *negative* which results in 50 000 reviews in common. 25 000 reviews will be used as a training set, 12 5000 as a validation set and the remaining 12 5000 reviews as a test set. Also, there is a subset of a training set that can be used to check if the program is running correctly.

First of all, the n most popular words were drawn out of all of the reviews and saved into `vocabulary.txt` file. In the beginning the number of 1000 was taken as n value. Some of the most popular words were: *the, and, this, that, was, for, with, movie, but, film, you, not, ...*. An observation might be that over 10 most popular words have neutral overtone which means that it is not possible to determine if the word imply positive or negative nature of the review. Also, some of the words are, so called, *stop words* what will be described in the further part of the report.

Second of all, the movie reviews were converted into *Bag of Words* representation which is the vector of the counters of the occurances of most popular words (from `vocabulary.txt` file) in the text. The *BoW* represantation of all of the reviews was later saved in a compressed file `train.txt.gz`.

3 Naive Bayes

As mentioned before, binary Naive Bayes was implemented and used for the training of the model and after for making predictions for test data. Is is a model that fits well when it comes to text processing. *BoWs* were used as an input feature vector to the model. The formula of binary Naive Bayes model is following:

$$\hat{y} = 1 \iff \sum_{j=0}^{n-1} x_j(\log \pi_{1,j} - \log \pi_{0,j}) + (\log P(1) - logP(0)) > 0), \tag{1}$$

which in other way might be also represented as:

$$\hat{y} = 1 \iff \boldsymbol{w}^T\boldsymbol{x} + b > 0, w_j = \log \pi_{1,j} - \log \pi_{0,j}, b = \log P(1) - logP(0). \tag{2}$$

The main objective of the training is to find the probabilities $\pi_{1,j}$ and $\pi_{0,j}$ which are the probability of classification to the class 1 (positive review) and the probability of classification to the class 0 (negative review), respectively. In other words, the vector w must have been found. The b value is equal to 0, because the probability of $\log P(1) == \log P(0) = \frac{1}{2}$, since the number of positive and negative reviews was the same.

The accuracy of the training was 82%. Here are some words with the biggest weights for the negative reviews: *waste, worst, awful, poorly, lame, horrible, crap, wasted, badly, worse, ...*, and for the positive reviews: *fantastic, wonderful, excellent, superb, amazing, powerful, perfect, unique, highly, masterpiece ...*. It is noticible that the most influential words are not those most popular that were mentioned in section 3. Further experiments with excluding *stop words* and with the use of stemming were conducted and described in section 4.

The validation accuracy was also 82%. The worst (with highest confidence) errors of classification were reviews: `7063_3.txt` classified as positive, `10377_8.txt` classified as negative, `12493_4.txt` classified as positive, `5521_4.txt` classified as positive and `1329_4.txt` classified as positive. I read those reviews and my suspicions about why they were wrongly classified are that those reviews are not as emotional as the reviews that were classified correctly (with the highest confidance), but they are rather more descriptive. Also they may be a little bit ironic which explains the use of postively charged words in a negative review. Interesting observation confirming the second thesis is that there was a higher rate of misclassification of negative reviews as positive.

4 Improvements

Firs of the improvements of the model performance could be, as mentioned before, getting rid of the stop words. After ignoring the stop words while searching the most common words, the most common would be: *movie, film, like, just, good, time, story, really, bad, people, great, ...*. It can be observed that in the opposite to the previous try of creating vocabulary, the positively or negatively charged words occur already in the first ten most common words (ex. *good, bad, great*). With the size of vocabulary of 1000 words, the training and validation accuracies are equal to 83%. It can be deducted that similar result of 82-83% of training and validation accuracy should be obtained with smaller size of the vocabulary. Decreasing the size of vocabulary to 200 seemed too be to extreme because it resulted in 76% training accuracy and 76% validation accuracy. The vocabulary size of 500 words gives 81% accuracy (both training and validation) and the size of 700 resulted in 82% accuracy ((both training and validation)) which is satisfactory considering that the vocabulary is smaller by nearly $\frac{1}{3}$.

Another improvement that can be made is the application of stemming which means that the ending od the word is cut off and only its root is left. It may give more reliable estimation of popularity of the specific word. For example, the words *connection, connections, connective, connected, connecting* will be considered as one word *connect*. After the applications of Porter's stemming with the vocabulary size of 700, the most common words were: *movi, film, like, just, time, good, make, charact, watch, stori, realli* and training and validation accuracies were both 82% which is a slight improvement by 1%. After increasing the size of vocabulary to 1000, 2000 and 5000, the training and validation accuracy were respectively: 83% and 82.5%, 84% and 83%, 85% and 83%. It can be deducted that with the size of vicabulary over 2000 the model might be overtrained because the train accuracy is increasing but validation accuracy stays the same. When I took a look at the most influential words they were: *boll, uw, thunderbird, beowulf, seagal, mst3k, dreck, ...* for the negative reviews and *antwon, gunga, goldsworthi, yokai, gypo, ...* for the positive reviews and some of those words don't make sense comparing to the most common words from the smaller vocabulary. There is a chance that some of these words are slang, they may be less popular but may be strongly charged positively or negatively. Those are nevertheless only my suspicions and they may be wrong.

Considering the model with the vocabulary of size 2000, which in my opinion was the best choice (the highest validation accuracy), ignoring stop words and application of stemming, the achieved test accuracy was equal to 84%.