Learning from data: Assignment 1
It is possible to automatically predict the sentiment and topic from only a reviews text?

Erik Bijl
s2581582
a.f.bijl@student.rug.nl

Abstract

Online reviews about several topics become more and more popular to read before purchasing or using something. To analyse a set of reviews it can be useful to train a classifier in order to extract certain facts from a review. In this experiment a classifier is trained by a Naive Bayes classification model to predict the sentiment and topic from only a reviews text. The results include that with a data split into a training and test set an average f-score of 0.81 for sentiment prediction and 0.92 for topic prediction can be achieved. Also cross validation is performed on the dataset which show comparable results to splitting the dataset into a training and test set.

1 Introduction

Nowadays it is common to provide your opinion on things you have purchased and used. A popular way to give your opinion is that of writing an online review about a certain product or service. As a consequence there is an increase in the amount of online reviews about certain topics. Analysing these reviews provides one with knowledge about a product but is a time-consuming task. Therefore a form of automatic scanning reviews is a time friendly manner to provide insights on a certain product or service.

In this first assignment is was asked to analyse a set of reviews in order to predict several aspects of additional reviews. In particular the experiment predicted the sentiment and topic from a reviews text. In order to predict a Naive Bayes classification model was used to train a classifier on a provided dataset. Below the theory and findings of these experiment are explained and discussed.

2 Data

In this assignment a corpus of reviews is provided as the dataset. The corpus contains 6000 text reviews on different topics and with varying lengths. The corpus contains on line reviews consisting of:

- the topic discussed in the review. The topics existing in this dataset are: books, camera, DVD, health, music and software.
- the mood expressed in the review. The possible moods that can be expressed are positive (pos) or negative (neg).
- an unique id for each review. During this experiment the id of a review is not used.
- the text of the review. The actual sentences the reviewer wrote down in the review. As a pre-processing step this text is already tokenised such that the appearance of each individual word can be an estimator in this experiment.

The distributions of both the sentiments and all the topics are approximately equal in the dataset. Meaning that approximately half of the dataset is positive (2958) where as the other half is negative (3032) and that the count of each different topic has approximates $\frac{1}{6}$ of the reviews.

In order to extract features from the reviews text a method called term frequency-inverse document frequency (tf-idf) is used. This method assigns a value for each word in a document representing the importance of that word. In this way the importance of words that frequently appear in each review is decreased where as the importance of unique words in a review is increased.

3 Method/Approach

During the experiment a classifier is trained on a subset of the original dataset, this part is called the training set. In our experiment we start with a balance of 75% of the examples belonging to the training set. The remaining 25% is used as a test set in order to test the performance of the classifier. A strong separation between the training and test set is needed because evaluating examples where a classifier is already trained on does not show that the classifier really learned something. When a classifier can correctly predict new examples it shows that it has effectively learned to link the appearance of words to the sentiment or topic. An example would be that a trained classifier could 'learn' to link the appearance of the word 'read' to the topic of a book or the appearance of the word 'boring' to a negative review. Applying the classifier on the test set where also the labels are known enables one to determine whether the classifier predicted the correct label based on given text.

The classifier uses the multinomial Naive Bayes model (Manning et al., 2008) to predict the class of the examples. In this model each word in a to be predicted review is split and the probability of every single word occurring in a certain class are multiplied. The class with the highest total probability is then assigned as the classification of this example. The actual formula in a naive bayes model is given by:

$$c_{map} = argmax \prod_{j=1}^{n} p(i_j|c) \cdot p(c)$$

where $p(i_j|c)$ is the probability that given class c the review has i_j and $p(c)$ the so-called prior probability of happening c.

In order to test the correctness of a classification four different scenarios can be distinguished. Consider an example x, the possible label of that example y and a classification of that example $C(x)$ to a class. We consider in this case that the label y can belong to either class α or to a different class. The classification of this example can have four different outcomes:

- A True positive (TP), the case that the classifier predicts example x belongs to class α and this example also has label α.

$$TP : C(x) = \alpha \wedge y = \alpha$$

- A True Negative (TN), the case that the classifier predicts example x belongs not to class α and this example also has another label than α.

$$TN : C(x) \neq \alpha \wedge y \neq \alpha$$

- A False Positive (FP), the case that the classifier predicts example x belongs to class α but this example has a different label than α.

$$FP : C(x) = \alpha \wedge y \neq \alpha$$

- A False Negative (FN), the case that the classifier predicts example x belongs not to class α but this example has label α.

$$FN : C(x) \neq \alpha \wedge y = \alpha$$

To evaluate the performance several measures can be calculated from these notations. Note that above one case was considered but below the total count of TP, TN, FP and FN are considered. In this experiment the following measures are used:

- The accuracy is the measure of all instances that are correctly categorised over all instances. The accuracy is given by:

$$A = \frac{TP + FP}{TP + FP + TN + FN}$$

- The precision is the measure of all instances that are correctly classified to a class over all examples that are classified to that class. The precision is given by:

$$P = \frac{TP}{TP + FP}$$

- The recall is the measure of all instances that are correctly categorised to a class over all examples of that class. The recall is given by:

$$R = \frac{TP}{TP + FN}$$

- The F-score is a combination of the precision and recall and can be seen as the overall performance of a classification. The F-score is given by:

$$F = 2 \cdot \frac{P \cdot R}{P + R}$$

Also the confusion matrix is used as a visualisation of the performance of an algorithm. The confusion matrix consists of two axis, it shows the predicted versus the true classes. This enables one with an overview of the performance of the trained classifier on the test set. From a confusion matrix it is easy to see how certain classes were predicted, for example which classes were mistaken for a another class. Therefore we know which classes were difficult or easy to predict.

In the experiment also a different construction of test and training sets is used called k-fold cross validation. In this method the dataset is split in k equal parts where one of the k parts is used to test the classifier. The classifier is trained on the remaining k-1 parts. By using k-fold cross validation average and total measures can be used to indicate the performance with as advantage that each k parts is used for both training and testing.

4 Results

As described above two experiments are performed. The first one with simply splitting the dataset into a training and test set and the second one with n-fold cross validation.

4.1 Training and test split

In the first experiment the dataset is split into a training and test set. The training set consists of 75% of the whole dataset and contains 4500 reviews. The test set contains the remaining which are 1500 reviews. Tables 1 and 2 show several measures obtained from testing the trained classifier on predicting respectively the sentiment and topic.

The tables show us also a different f-score for certain classes. For example when we predict the sentiment of the reviews the f-score for positive is 0.81 where the f-score for negative is 0.75. This difference means that the classifier is better in predicting positive reviews than negative. When the classifier predicts topics we also see that certain topics could be better predicted than others. The topics books and camera are best predicted.

An interesting observation can be made by changing the amount of of reviews in both sets and compare the performance measures of both splits. The amount of reviews in the training set is increased to 90% of the whole dataset where the training set contains the remaining 10%. The performance measures of both class predictions are shown in 3 and 4. Comparing these tables to those of the smaller training set indicates a better performance on the second split. The average scores on all measures are increased which corresponds to a better classification. Also different topics were best predicted compared to the smaller dataset.

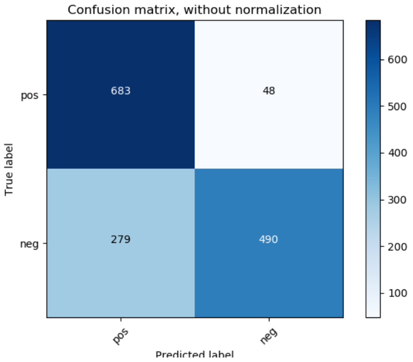

Figure 1: The confusion matrix of predicting the sentiment of a review

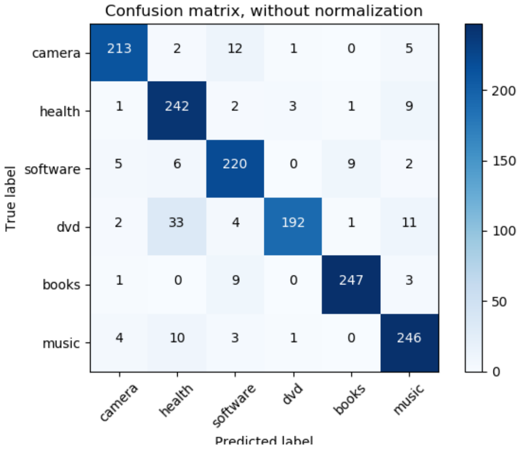

Figure 2: The confusion matrix of predicting the topic of a review

class	acc	pre	rec	f-scr
pos	0.93	0.71	0.93	0.81
neg	0.64	0.91	0.64	0.75
average	0.78	0.81	0.78	0.78

Table 1: Performance measure on predicting sentiments

class	acc	pre	rec	f-scr
camera	0.91	0.94	0.91	0.93
health	0.94	0.83	0.94	0.88
software	0.91	0.88	0.91	0.89
dvd	0.79	0.97	0.79	0.87
books	0.95	0.96	0.95	0.95
music	0.93	0.89	0.93	0.91
average	0.91	0.91	0.91	0.91

Table 2: Performance measure on predicting topics

class	acc	pre	rec	f-scr
pos	0.92	0.74	0.92	0.82
neg	0.71	0.91	0.71	0.80
average	0.81	0.83	0.81	0.81

Table 3: Performance measure on predicting sentiments with a large training set (90%)

4.2 k-fold cross validation

As described above another method to evaluate the performance of a clustering is k-fold cross validation. In this case the amount of $k = 5$ was used to indicate a performance. In this experiment cross validation is applied to predict the topic of reviews. The confusion matrices for each of these folds is shown in figure 3 for the topics. It can be seen that the performance on each fold is approximately similar. In cross validation each example is classified so if all folds are summed up we have a performance over all examples. From this overall performance the measures as shown before can be extracted and are shown in table 5. One can see that the highest f-score is obtained for the topic books and lowest for health. Still the differences are not considered high between the topics.

class	acc	pre	rec	f-scr
dvd	0.94	0.98	0.94	0.96
books	0.96	0.84	0.96	0.90
music	0.88	0.91	0.88	0.90
camera	0.80	1.00	0.80	0.89
health	0.95	0.94	0.95	0.95
software	0.96	0.88	0.96	0.92
average	0.92	0.92	0.92	0.92

Table 4: Performance measure on predicting topics with a large training set (90%)

class	acc	pre	rec	f-scr
books	0.87	0.94	0.94	0.96
camera	0.94	0.83	0.96	0.90
dvd	0.91	0.86	0.88	0.90
health	0.76	0.97	0.80	0.89
music	0.93	0.93	0.95	0.95
software	0.93	0.88	0.96	0.92
average	0.89	0.92	0.92	0.92

Table 5: Performance measure on predicting topics with cross validation

5 Discussion/Conclusion

In the experiment sentiment and topic prediction is performed using a Naive Bayes classification. The first experiment used a split of 75% for the training set and 25% for the test set. The results showed certain measures for the classification where it can be concluded that in this experiment the topic prediction had a better performance than predicting the sentiment. This was also the case when we used a split of 90% for the training set and 10% for the test set. A difference was that when using more training examples also the performance improved. This performance increase supports that using more training examples results in a higher f-score. Therefore an possible improvement could be to incorporate more examples to achieve an even better classification.

Also an experiment was performed using 5-fold cross validation. The results were shown in the same manner as the first experiments but did not show an increase in f-score. The reason for this could be that 5-fold cross validation is almost identical with a 75% split. In both cases the Naive Bayes classification model was used to train the classifier. Therefore an potential improvement could be to use a higher number of k in the k-fold cross validation.

References

Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. *Introduction to Information Retrieval*. Cambridge University Press, New York, NY, USA.

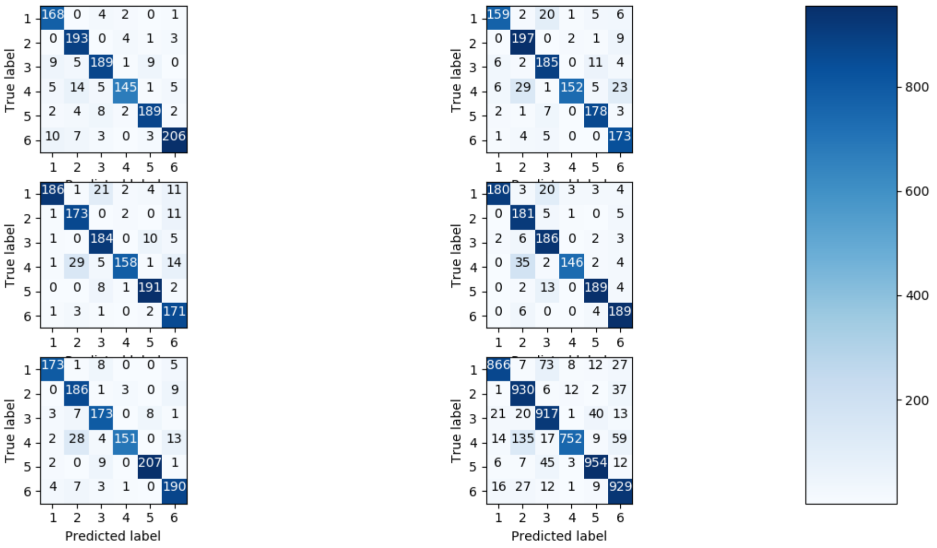

Figure 3: The confusion matrices with cross validation. The performance on each fold is shown. The last plot is the the total count of all tested folds