Learning from Data
Assignment 1

Robin Schut
s2717166

Sharif Hamed
s1234567

September 14, 2020

Exercise 1.1 - Settings

Please refer to the comments in the code.

Exercise 1.2 - Binary vs Multi-class Classification

Using the sentiment labels (positive, negative), the classification problem becomes binary because there are two possible classes. A multi-class classification problem arises when we use the topic labels. There are six distinct topic labels for the problem. These labels construct the set {books, dvd, camera, health, music, software}. We can construct our data set using the function `read_corpus` in the *DataService* class. When setting the variable `use_sentiment` to True, the data set is constructed using the sentiment labels. The data set is constructed using topic labels when this variable is set to False.

Exercise 1.3 - Measures

In order to perform the experiments we have made a separate class `Experiments` that contains scripts to run the experiments. Please refer to the code. The method `experiment_binary` prints out the precision-, recall and f1-scores for the binary classification problem. The results are posted in table 1. We have rounded the numbers to four decimal precision.

Table 1: Scores binary classification

	Precision	Recall	F1-score
pos	0.9108	0.6372	0.7498
neg	0.71	0.9343	0.8069

The scores for the multi-class classification problem are reported in table 2.

Table 2: Scores multi-class classification

	Precision	Recall	F1-score
books	0.9425	0.9142	0.9281
dvd	0.88	0.9091	0.8943
camera	0.8259	0.938	0.8784
health	0.9746	0.7901	0.8727
music	0.9574	0.95	0.9537
software	0.8913	0.9318	0.9111

The precision score is defined as $Tp/(Tp + Fp)$. In words, it is the ratio of correctly classified classes C_j versus all classified output on the test set as C_j. The recall score is defined as $Tp/(Tp + Fn)$. In words, it is the ratio of the correctly classified C_j versus all output of the test set with label C_j. The F1-score is defined as $F_1 = 2 * (P_r + R_e)/(P_r * R_e)$, where P_r is the precision score and R_e is the recall score.

A good indicator for performance is the F1-score as this relates the output of the model to the precision and recall, which relate to all of the classes. Therefore, we can observe that the model performs marginally better on recognizing negative sentiments in the binary classification case. Furthermore, we observe that the category music performs well in case of the multi-class classification problem.

Exercise 1.4 - Probabilities

The Multinomial Naive Bayes classifier is used to output probabilities. When fitting the data it calculates the priors. These priors are the believes of the system before the novel data is seen. The way these priors are calculated is by word counting.

Table 3: Prior probabilities

	Books	Camara	DVD	Health	Music	Software	Total
Word count	760	730	770	743	767	730	4500
Prior	0.169	0.162	0.171	0.165	0.170	0.162	1

As we see in table 3, the priors are calculates by the following formula:

$$Prior_i = \frac{N_i}{N} \tag{1}$$

Where N_i is the number of words of class i and N the total number of words.

For the first test sentence the posterior probabilities can be seen in table 4. We see that the posterior are a lot different than the priors. This is because the

the priors are multiplied by the likelihood of the data and after that normalized again. This is the same for binary class inference.

Table 4: Posterior probabilities

	Books	Camara	DVD	Health	Music	Software
Posterior	0.545	0.041	0.242	0.037	0.088	0.046

Exercise 1.5 - Report

(a) Why should one not look at the test set?
answer: The test set is used to evaluate our model. If we look at the test set during training, the model will train a bias towards the test set and we can never get a proper evaluation score off of the test set.

(b) What happens with cross-validation?
answer: In cross-validation, we split the whole data set into k equal parts (instances and labels). We hold one instance out, iterating from $j == 1$ until $j == k$. For each instance j, we train our model on the rest of the instances and we evaluate on the held-out instance. Doing so, we can generalise our model properly, rather than having it biased towards instances it has seen over and over.

(c) What baseline could you use for the binary sentiment classification for the six-class topic classification? What should their performance be?
answer:

(d) Why is it useful to look at the confusion matrix?
answer: The confusion Matrix is the representation of the correctly and incorrectly classified test set output for a class. The confusion matrix is easiest understood in a binary classification problem (for instance, $C_j, \neg C_j$). The confusion matrix then looks like this:

TP	FP
FN	TN

Table 5: Caption

The entries of the matrix are abbreviations for True Positive, False Positive, False Negative, True Negative. The positive/negative indicate the output of our model $(C_j, \neg C_j)$ and true/false indicates whether the output of our model was correct or not. In other words, they are measurements to quantify the performance of our model.

(e) What features are used in the classification task you ran?
answer: The features in our classification task are a vectorizer object using a bad-of-words. These are vectors count the occurrences of each word with a label. These bag-of-words are used to calculate the prior probability of an occurrence of a word with a label. This information is then used to calculate the posterior probability. It seems as though every single word is used through the use of the `identity` function. However, in the lecture slides certain key-words are used.

(f) What is the difference between using accuracy and using F-score?
answer: The accuracy is the perunage of correctly classified class labels of test set (jaccard-score/ hamming distance). The F-score is the function The precision score is defined as $Tp/(Tp + Fp)$. In words, it is the ratio of correctly classified classes C_j versus all classified output on the test set as C_j. The recall score is defined as $Tp/(Tp + Fn)$. In words, it is the ratio of the correctly classified C_j versus all output of the test set with label C_j. Therefore, the F-score provides information on the performance of our model with respect to the test set, rather than only a measure of the performance of the model.

(g) What is the difference between macro F-score and micro F-score and what different functions and implications do they have?
answer: From the documentation: f1-score, we observe that micro-averaging calculates the metrics true positive, false negative and false positives globally and macro-averaging calculates the unweighted mean for each class label. The output of both averaging metrics is a single float rather than an output for each class label (hence the term 'average').