Davide Lagano
Georgios Patrikis

Lab Group 24

REPORT
Laboratory Exercise 3: Opinion Mining

Questions:

Question 1: Does this look like texts containing opinions? Are they correctly classified as positive and negative? Why can it sometimes be difficult to determine if a review should be classified as positive or negative?

If we open the two files ‘positive.review’ and ‘negative.review’ with a text editor, we can see that it's not really simple to go throw the opinions and of course doesn't look like a usual document with opinions because it has the format of HTML documents. It seems that in the documents the opinions are classified in the correct field (positive or negative), but of course it is not optimized to read every single opinion in this format, so I decided to go throw the document and pick random review.
It is difficult for a computer always classified a review as positive or negative because as we know, it isn't able to elaborate a semiotic of information as the human do. For example, it is not able to understand the sarcasm, for this reason, a phrase like "this book is simple to read as the 'The Brothers Karamazov'", that it is, of course, sarcastic, would have been classified as a 'simple to read book'.

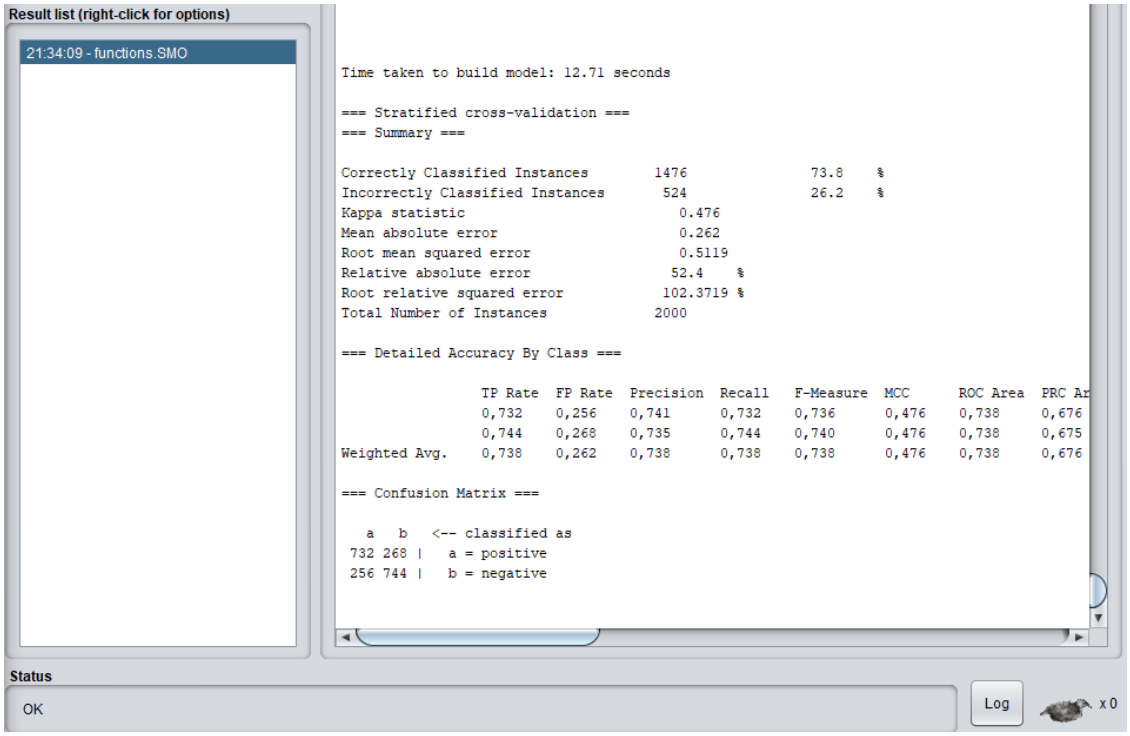
After executing the command ‘after executing the command’, will be available a file that take the 1000 reviews, convert them into ARFF-format and merge them into the file book_review.arff, which is readable by WEKA. We open the file in WEKA, and we convert the file into word vector format.

Question 2: Check the file book_review_vector.arff, what does it contain?

In the current relation, we have the first characteristics of the document: we can see that there are 2000 instances and 1251 attributes.In the attributes, we find the class and the other features. In the right part of WEKA tool, we can also visualize the statistics for each attribute, with min, max, mean, StdDev. Moreover, we can see a chart that reports the numbers of times that that word appear in the reviews.

Question 3: What percentage of correctly classified instances did you obtain?

After the 10-Folds Cross-validation, and after running the Support Vector Machine, as we can see from the picture below, we obtain 73.8% of instanced correctly classified.



Question 4: How do the performance results of the classifier change with the size of training set? Why do you think that is? Motivate and discuss.

The next step will be to increase the size of the training set, starting at 10% and going up to 90% to investigate how the performance changes as the size of the training set increases.

Results:
Training size -> Accuracy

10% -> 67.1667
20% -> 69.375
30% -> 69.2143
40% -> 70.0833
50% -> 71.2
60% -> 72.25
70% -> 71
80% -> 71.25
90% -> 74

As we can see, as much we increase the training set, as the accuracy grow. We think that was an expected result, because more we increase the training set, more the model is able to train with more data.

Question 5: How many correctly classified instances in percentage did you obtain?

After the 10th step, the instances correctly classified are 92,86%.

Question 6: How does this result compare to previous results? What are the drawbacks of this evaluation?

It's a very good result, but also the reviews that I add were really simple:

```
<review_text>
beautiful book
</review_text>

<review_text>
terrible book
</review_text>

<review_text>
amazing book
</review_text>

<review_text>
book really bad
</review_text>

<review_text>
Was simple to read it
</review_text>

<review_text>
Was difficult to read it
</review_text>

<review_text>
I will recommend this book at my friend
</review_text>

<review_text>
I will never recommend this book at my friend
</review_text>
```

We add this 8 review: the model misclassified the review: 'I will recommend this book at my friend' as negative.

It's also important to notice that the test set was extremely little, just 14 reviews.

Question 7: There are many ways to try to improve classification results. Discuss and motivate the methods you would do to try to improve the results you got during the lab so far.

Of course, the principal way to improve our classification result could be adding more train data. Because as we sow in the 3rd question, more data the model have, better are the results.

One other way could be to remove the stop words because they add just noise in our training set, and they don't have any value.

Question 8: Did the stop word filtering improve or impair the results after 10-fold cross-validation? How much? What size does the stop word filtered and the not stop word filtered file have respectively? Would you recommend using stop word filtering?

Now we are going to implement a new model that not consider the step words.
The accuracy improved, but just 0.25% more. The initial dataset was 2MB, the new one it's 400kb, this is the real achievement. I recommend using stop word filtering because allow us to have better accuracy with a reduction of data.

Question 9: Did stemming improve or impair the results after 10-fold cross-validation? How much? Does the stemmer work properly (HINT check the attribute list)? Which of the stemmers did you try?

In this last step, we are going to use stemming.The accuracy is 73.65%, which is worse than the previous one.I don't think that the stemmer properly worked because it generated strange words without any meaning.