David Schulz
CS 4980 001
4/4/2022

Homework 3

1. How accurate was the classifier? What was the Precision and Recall? The F-measure?

		Actual		
		Positive	Neutral	Negative
Predicted	Positive	1	0	1
	Neutral	1	1	0
	Negative	0	2	2

a. Accuracy: $\frac{Trues}{Total} = \frac{4}{8} = 0.5$

b. Precision

i. Positive: $\frac{TP}{TP+FP} = \frac{1}{2} = 0.5$

ii. Neutral: $\frac{TP}{TP+FP} = \frac{1}{2} = 0.5$

iii. Negative: $\frac{TP}{TP+FP} = \frac{2}{4} = 0.5$

c. Recall

i. Positive: $\frac{TP}{TP+FN} = \frac{1}{2} = 0.5$

ii. Neutral: $\frac{TP}{TP+FN} = \frac{1}{3} = 0.33$

iii. Negative: $\frac{TP}{TP+FN} = \frac{2}{3} = 0.67$

d. F-measure

i. Positive: $2 * \frac{precision * recall}{precision + recall} = 2 * \frac{0.25}{1} = 0.5$

ii. Neutral: $2 * \frac{precision * recall}{precision + recall} = 2 * \frac{0.165}{0.83} = 0.398$

iii. Negative: $2 * \frac{precision * recall}{precision + recall} = 2 * \frac{0.335}{1.17} = 0.573$

2. Choose one incorrectly classified document.

a. Manually calculate the sentiment probabilities for the document (you can use your classifier to generate the likelihoods and prior probabilities, but do the classifying on paper)

i. Incorrectly classified document: “The program sucks.”

ii. Logprior[“POS”] = -1.099

iii. Logprior[“NEG”] = -1.099

iv. Logprior[“NEU”] = -1.099

v. Loglikelihood[“the”] = [-3.076, -3.541, -3.512] # [POS, NEG, NEU]

vi. Loglikelihood[“program”] = [-3.769, -4.234, -4.205] # [POS, NEG, NEU]

vii. Loglikelihood[“sucks”] does not exist

viii. $\hat{P}(positive) = logprior[\text{"POS"}] + \sum_{w\ in\ V} loglikelihood[w, \text{"POS"}] = (-1.099) + (-3.076) + (-3.769) = -7.944$

        ix. $\hat{P}(negative) = logprior[\text{"NEG"}] + \sum_{w\ in\ V} loglikelihood[w, \text{"NEG"}] = (-1.099) + (-3.541) + (-4.234) = -8.874$

        x. $\hat{P}(neutral) = logprior[\text{"NEU"}] + \sum_{w\ in\ V} loglikelihood[w, \text{"NEU"}] = (-1.099) + (-3.512) + (-4.205) = -8.816$

        xi. $Argmax_c = Positive$

b. What is the difference of the probability sums of the correct class and the class assigned by the system?

    i. $\hat{P}(positive) - \hat{P}(negative) = -7.944 - (-8.874) = 0.93$

c. Identify the term or terms that caused the system to misclassify the document.

    i. “Sucks”, which indicates strong negative sentiment, doesn’t exist in the training data. “Program” was also slightly more common in positive reviews, skewing the word to indicate positive even though the word should be neutral.

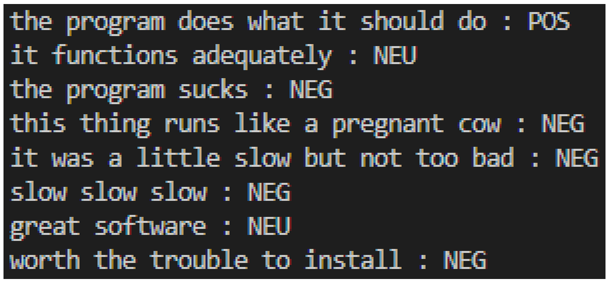
d. Build a document (or documents) to add to the training set that would allow the system to correctly classify the document.

    i. Show the mathematical reasoning for your choice of words in the document.

        1. New document: “The thing sucks!”

        2. Adding this new negative document with “sucks” makes loglikelihood[“sucks”] = [-4.883, -4.270, -4.913] with add-1 smoothing. The loglikelihood for negative is noticeably higher than the other two, so adding that value will increase the probability of a document containing the word “sucks” being negative.

    ii. Rerun the tests with the additional information.



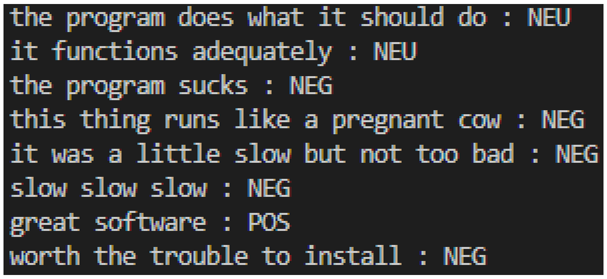
```
the program does what it should do : POS
it functions adequately : NEU
the program sucks : NEG
this thing runs like a pregnant cow : NEG
it was a little slow but not too bad : NEG
slow slow slow : NEG
great software : NEU
worth the trouble to install : NEG
```

    iii. Did adding the additional information change any other document classification? If so, how? Did it improve the overall accuracy of your system or make it worse?

        1. No, it didn’t change any other document classification. It improved the overall accuracy of the system to 5/8 = 0.625.

e. Add the MPQA Subjectivity Cues Lexicon to your system and run the tests again and report the results.

i. Results



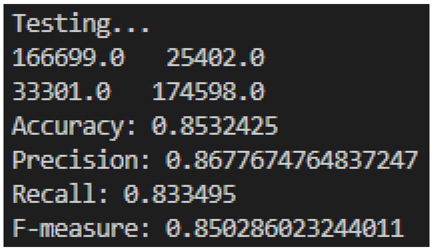
```
the program does what it should do : NEU
it functions adequately : NEU
the program sucks : NEG
this thing runs like a pregnant cow : NEG
it was a little slow but not too bad : NEG
slow slow slow : NEG
great software : POS
worth the trouble to install : NEG
```

ii. Choose a document that was classified differently after adding the lexicon. Was it correctly or incorrectly classified? Discuss why.

1. “Great software” is now correctly classified because the word “great” wasn’t in the training set before adding the lexicon, but now with the lexicon saying that “great” is a positive word, it is taken into consideration and greatly contributes to the probability of the document being positive sentiment.

f. Finally, use the provided collection of Amazon reviews from 2007 to train your classifier. Run the associated tests and report the Precision, Recall, and F-measure.

i. Output



```
Testing...
166699.0   25402.0
33301.0   174598.0
Accuracy: 0.8532425
Precision: 0.8677674764837247
Recall: 0.833495
F-measure: 0.850286023244011
```

ii. Confusion Matrix

	Positive	Negative
Positive	166699	25402
Negative	33301	174598

iii. Accuracy: 0.8532425

iv. Precision: 0.8677674764837247

v. Recall: 0.833495

vi. F-measure: 0.850286023244011

g. Briefly discuss what you learned from this assignment, what you liked or disliked about the assignment and, optionally, anything you would like to see changed or added to improve the assignment.

i. I learned how important performing good pre-processing on your dataset before training with it really is. I liked training with a big dataset from a real source, but working with the Amazon reviews dataset took a lot longer than I thought it would. In the future, I suggest using a smaller dataset for the homework.