Notebook

April 29, 2020

0.0.1 Question 6c

Provide brief explanations of the results from 6a and 6b. Explain why the number of false positives, number of false negatives, accuracy, and recall all turned out the way they did.

the classifier never predicts positive, so the number of false positives is 0. False negatives mean the mails predicted ham mail, but originally those are spam mails, so I counted the number of spam mails in Y_train. Because the classifier predicts all negative, the accuracy is simply true negatives (ham mails) over the size of Y_train. The numerator of the equation for recall is True Positive, but there is no positives with the classifier, so recall would be 0.

0.0.2 Question 6e

Are there more false positives or false negatives when using the logistic regression classifier from Part A?

The number of false negatives is 1699, and the other is 122. There are more false negatives.

0.0.3 Question 6f

1. Our logistic regression classifier got 75.8% prediction accuracy (number of correct predictions / total). How does this compare with predicting 0 for every email?
2. Given the word features we gave you above, name one reason this classifier is performing poorly. Hint: Think about how prevalent these words are in the email set.
3. Which of these two classifiers would you prefer for a spam filter and why? Describe your reasoning and relate it to at least one of the evaluation metrics you have computed so far.

1. the 0 positive classifier has 74.5% prediction accuracy (True negative (ham mails)/ total size (ham mails + spam mails)). There is just 1% profit to use our logistic model.
2. I think the word 'memo' is one of the reasons that makes predictions less accurate. When I checked through a graph, there were more 'memo' words in ham mails than spam mails.
3. I prefer models with 'logistic regression'. As a result of replacing 'memo' with 'free' in 'some_words' above, I confirmed that the accuracy increased up to 76%. I think if we change some conditions, we can make a model with higher accuracy.

0.0.4 Question 7: Feature/Model Selection Process

In the following cell, describe the process of improving your model. You should use at least 2-3 sentences each to address the follow questions:

1. How did you find better features for your model?
2. What did you try that worked / didn't work?
3. What was surprising in your search for good features?

1. I wanted to improve the accuracy of predictions by changing 'some_words'. For that, I found the more frequent words from spam mails (like '', '</'). I used the '.head ()' to find those words, and I checked them using the graph in project 2a.
2. I had confirmed that the expression '!!!' comes out at a high frequency in spam mails. On the other hand, '!' Is used in ham mails as well as spam mails. So I added '!!!' to 'Some_words', but I got lower 'score' than putting '!'. Also, putting the number of characters in 'subject' into a variable did not significantly affect the predict.
3. I was amazed that the number of characters in the mail greatly influences the prediction. As a result of analyzing the contents of spam mails and ham mails, spam mails with content of less than 3000 characters were about 50% and ham mails were about 85%. I was interested that the model with this difference is about 7 percent more accurate than the previous model.

Generate your visualization in the cell below and provide your description in a comment.

In [12]:
```
# Write your description (2-3 sentences) as a comment here:
# When I put the number of characters in the email into the prediction model, it was quite int
# Since I scored about 7% higher than the previous model, I wanted to graph the difference of
# When the number of characters in the 'subject' ranged from 20 to 80, I could see spam mails

# Write the code to generate your visualization here:

new_dataq8 = train.copy();
new_dataq8['length_email'] = [len(x) for x in new_dataq8['email'].tolist()];
new_dataq8['length_subject'] = [len(str(x)) for x in new_dataq8['subject'].tolist()];
new_seriesq8 = new_dataq8.groupby('length_subject')['length_email'].mean();
new_seriesq8_1 = new_seriesq8.reset_index();

sns.set(rc={'figure.figsize':(12, 8)});
gh = sns.scatterplot(x = 'length_subject', y = 'length_email', hue = 'spam', size = 1, data = n
plt.plot(new_seriesq8_1['length_subject'], new_seriesq8_1['length_email'])
plt.xlim(0, 120)
plt.ylim(0, 50000)
# Note: if your plot doesn't appear in the PDF, you should try uncommenting the following line
# plt.show()
```

Out[12]: (0, 50000)