Data Science - Homework 8 (Text mining)

- Introduction of datasets

1. 60K Stack Overflow Questions with Quality Rating (from Kaggle)

Stack Overflow is the principal site for professional programmers and enthusiasts to prompt and discuss questions regarding computer science. A remarkable feature of Stack Overflow is their user reputation award process. This feature acts as the primary governing platform for evaluating quality of questions and answers, allowing the site to be self-moderating by its community.

This is a dataset containing 60,000 Stack Overflow questions from 2016-2020. Questions are classified into three categories: HQ (high quality), LQ_EDIT (low quality but still open), LQ_CLOSE (low quality — close by the community). After importing this dataset, I found that it has 45000 records in training set, and validation set has 15000 records, both of them has 6 attributes. Then, I check the information of the training set and validation set, there are not any missing data.

	Id	Title	Body	Tags	CreationDate	Y
0	34552656	Java: Repeat Task Every Random Seconds	<p>I'm already familiar with repeating tasks e...	<java><repeat>	2016-01-01 00:21:59	LQ_CLOSE
1	34553034	Why are Java Optionals immutable?	<p>I'd like to understand why Java 8 Optionals...	<java><optional>	2016-01-01 02:03:20	HQ
2	34553174	Text Overlay Image with Darkened Opacity React...	<p>I am attempting to overlay a title over an ...	<javascript><image><overlay><react-native><opa...	2016-01-01 02:48:24	HQ
3	34553318	Why ternary operator in swift is so picky?	<p>The question is very simple, but I just cou...	<swift><operators><whitespace><ternary-operato...	2016-01-01 03:30:17	HQ
4	34553755	hide/show fab with scale animation	<p>I'm using custom floatingactionmenu. I need...	<android><material-design><floating-action-but...	2016-01-01 05:21:48	HQ

2. 200K SHORT TEXTS FOR HUMOR DETECTION (from Kaggle)

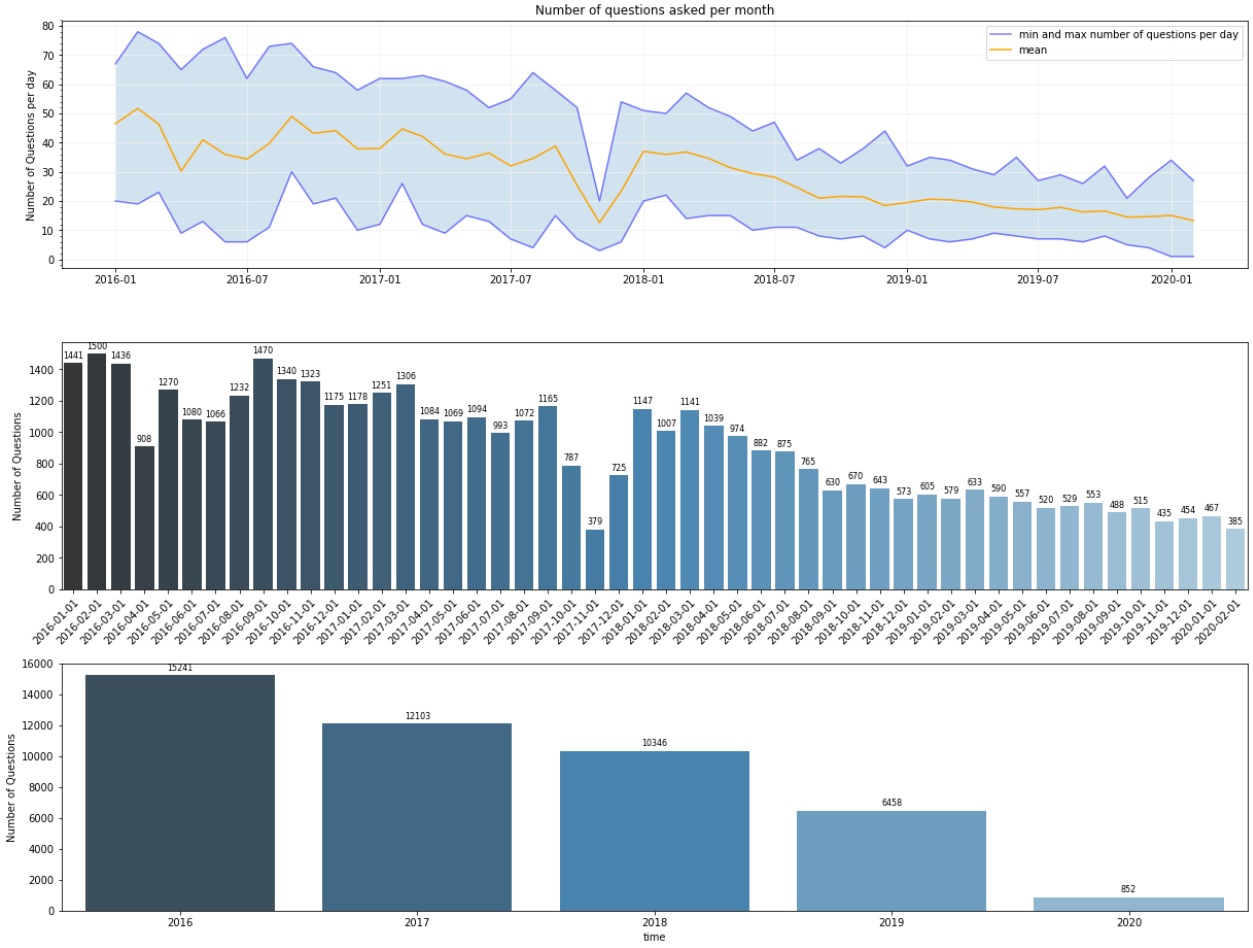
Automatic humor detection has interesting use cases in modern technologies, such as chatbots and virtual assistants. Existing humor detection datasets usually combined formal non-humorous texts and informal jokes with incompatible statistics. This makes it more likely to detect humor with simple analytical models and without understanding the underlying latent lingual features and structures.

I found a dataset for the task of humor detection, which contains 200k labeled short texts, equally distributed between humor and non-humor. It has no any missing data that is necessary to preprocess.

- Conduct text mining with data preprocessing and frequency analysis.

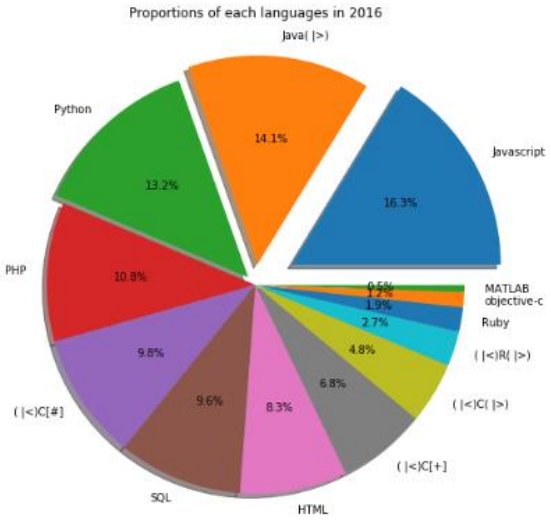
1. 60K Stack Overflow Questions with Quality Rating

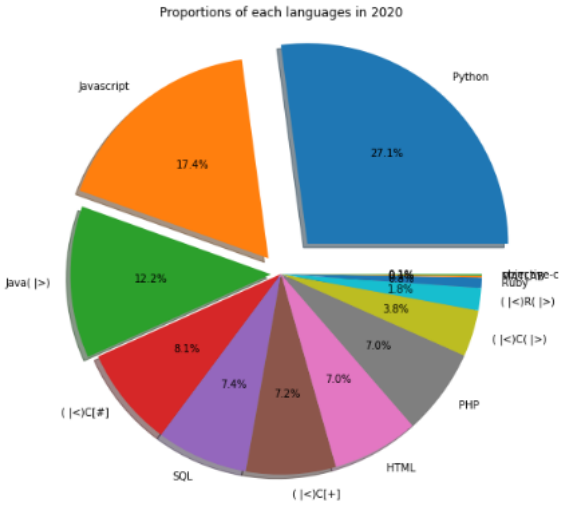
The first task I think that it's interesting to know the number of question according to time. It is a necessary step to group by object created to calculate the number of questions per day, month and year, respectively. It would be a little bit messy to show the number of questions per day, instead of that I showed the mean, min and max number of questions asked per days over months.



From the analysis above, we have more questions from 2016 than 2020. The algorithm could be influenced by old questions, we need to be careful about the influence of the time on our predictions.

The second task I conducted is to know which language have the more questions on Stack Overflow. I only showed the result of 2016 and 2020 and gave some my observations for this part.



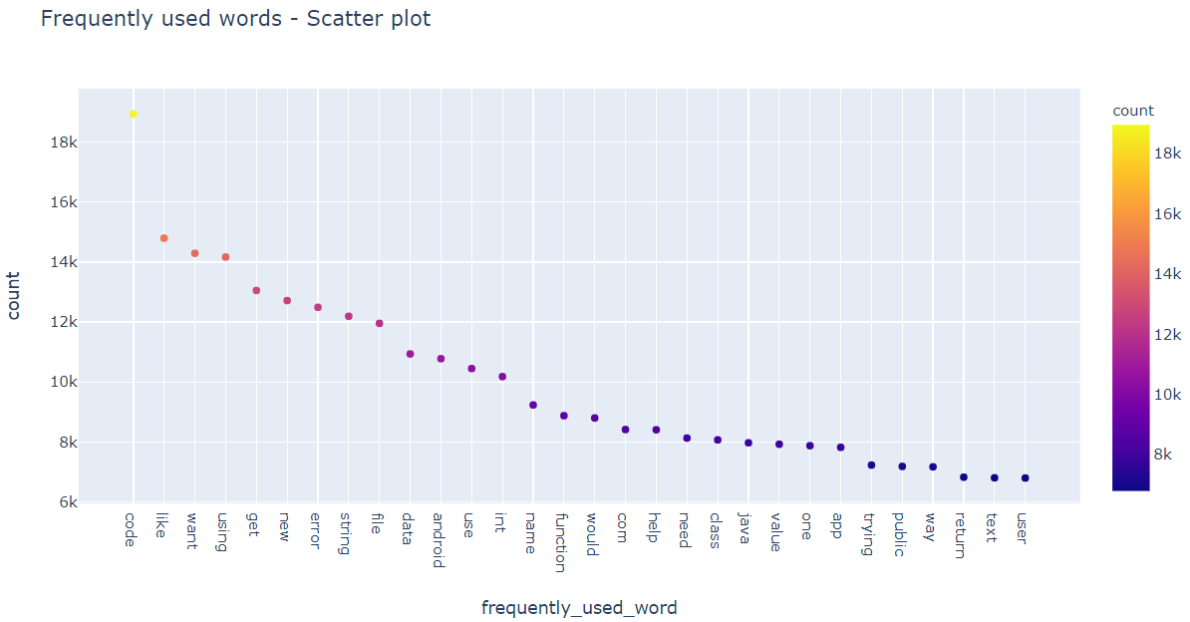


We can obviously see that python is one of the most popular languages in recent years, and it has the largest change between 2016 and 2020. The other interesting thing is that javascript and java still maintain a high number of questions.

From the figure on the right, we can observe that the most popular technologies regardless of the year, and the words C and java appear the most times, so we can see that the fonts of these two words in the word cloud are the largest. The second largest is Name and python, but I don't know what is the name of this technology XD. I believe that python is well known, as long as a person studied in the field of computer science.

Finally, I want to do the frequency analysis for used words in the questions, and I choose the scatter plot to present. As the Body column contains both code and content, I will have to remove code from the content. Also, I will do a little cleaning on the content by removing stop words and less than 3 characters.

	Id	Title	Body	Tags	CreationDate	Y	Body_code	Body_content	content_words
0	34552656	Java: Repeat Task Every Random Seconds	<p>I'm already familiar with repeating tasks e...	java,repeat,	2016-01-01 00:21:59	LQ_CLOSE	None	i m already familiar with repeating tasks ever...	[already, familiar, with, repeating, tasks, ev...
1	34553034	Why are Java Optionals immutable?	<p>I'd like to understand why Java 8 Optionals...	java,optional,	2016-01-01 02:03:20	HQ	None	i d like to understand why java optionals wer...	[like, understand, why, java, optionals, were,...
2	34553174	Text Overlay Image with Darkened Opacity React	<p>I am attempting to overlay a title over an ...	javascript,image,overlay,react-native,opacity,	2016-01-01 02:48:24	HQ	<code>//component for article preview touchabl...	i am attempting to overlay a title over an ima...	[attempting, overlay, title, over, image, with...



Classification

1. Logistic Regression

The classification performance does not seem to be ideal by using the logistic regression. The test accuracy is less than 70%, train accuracy is only 81.4%. I will try to use the other models like BERT or LSTM later to improve the prediction.

```
Confusion Matrix =
[[4128  547  325]
 [ 716 2872 1412]
 [ 496 1467 3037]]

Classification Report =
              precision    recall  f1-score   support

          HQ       0.77      0.83      0.80      5000
    LQ_CLOSE       0.59      0.57      0.58      5000
     LQ_EDIT       0.64      0.61      0.62      5000

    accuracy                           0.67     15000
   macro avg       0.67      0.67      0.67     15000
weighted avg       0.67      0.67      0.67     15000
```

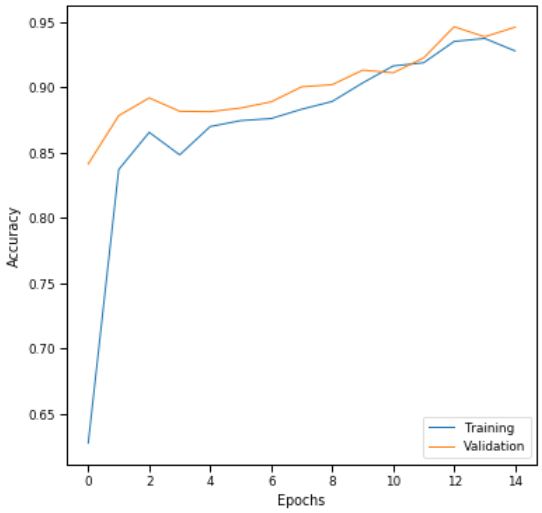
```
Train accuracy: 0.814
Test Accuracy: 0.669
Misclassified examples: 4963
```

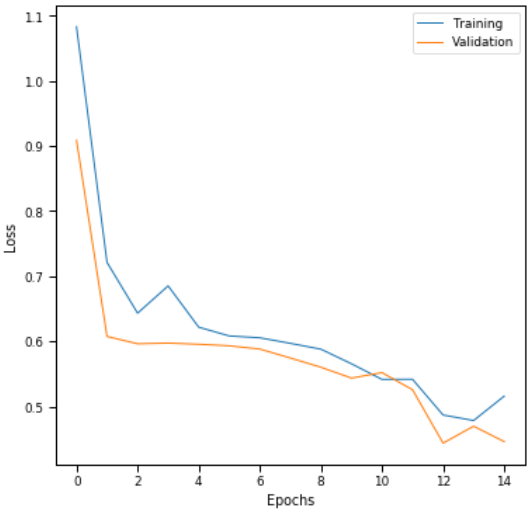
2. LSTM

I construct a simple LSTM architecture to classify the quality of this dataset and compare the performance between LSTM and logistic regression model. Obviously, we get a better result from LSTM model, it is the result I expected. The accuracy and loss curves of training phase are showed in the next page.

```
Model: "model"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
input_2 (InputLayer)         [(None, None)]            0
_________________________________________________________________
embedding (Embedding)        (None, None, 128)         2560000
_________________________________________________________________
bidirectional (Bidirectional (None, None, 128)         98816
_________________________________________________________________
bidirectional_1 (Bidirection (None, 128)               98816
_________________________________________________________________
dense (Dense)                (None, 32)                4128
_________________________________________________________________
dropout (Dropout)            (None, 32)                0
_________________________________________________________________
dense_1 (Dense)              (None, 3)                 99
=================================================================
Total params: 2,761,859
Trainable params: 2,761,859
Non-trainable params: 0
_________________________________________________________________
```

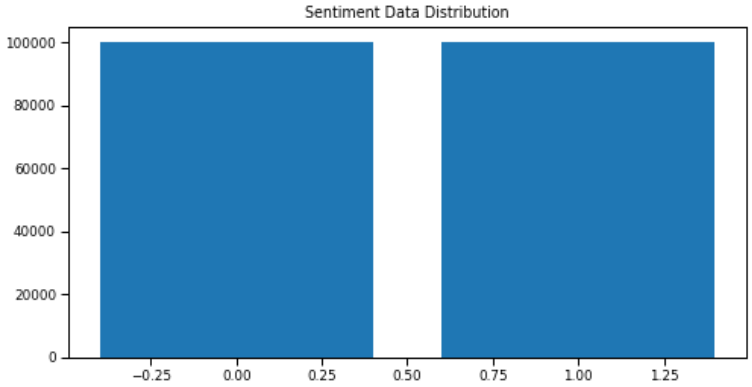
The train accuracy and test accuracy are 92.8% and 94.6%, respectively. The loss curves of both training and validation seem to be still declining. I think a longer training time may get the better results.



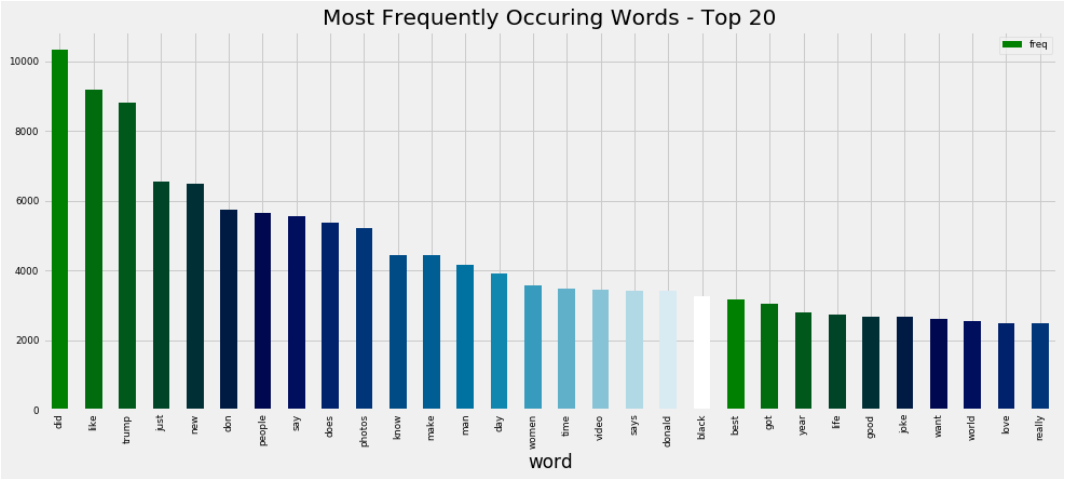


2. 200K SHORT TEXTS FOR HUMOR DETECTION (from Kaggle)

For this dataset, I want to know the distribution of the categories, I learn that this dataset is balanced with 100000 records in each of the two categories.



Then, I use the other presentation of frequency analysis – bar plot, and list the top 20 most frequently occurring words. Compared with scatter plot, I think that the bar plot is a better method for me, because it is more clear at a glance, if it is a scatter plot, there may be a problem that the point corresponds to the wrong word.



Classification

This part I will use the same model with the previous dataset, it is convenient for comparison.

1. Logistic regression

The test accuracy is up to 87%, and the train accuracy is 87.6%. This result is much better than the previous dataset, so I think logistic regression model is more suitable for classifying this kind of task, maybe we can say that it is a relatively simple task. The following diagram is the ROC curve for logistic regression classifier.

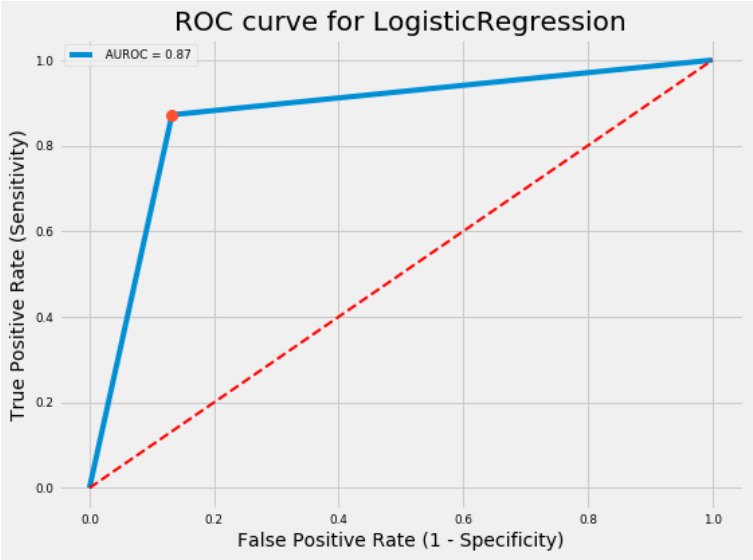
```
Confusion Matrix =
[[21612  3300]
 [ 3209 21879]]

Classification Report =
              precision    recall  f1-score   support

       False       0.87      0.87      0.87     24912
        True       0.87      0.87      0.87     25088

    accuracy                           0.87     50000
   macro avg       0.87      0.87      0.87     50000
weighted avg       0.87      0.87      0.87     50000

Train accuracy: 0.876
Test Accuracy: 0.870
Misclassified examples: 6509
```



2. LSTM

I modified the training epoch this time, it seems to get a good performance.