Assignment 2

Baixi (Patrick) Wu

Jiashu (Julie) Zhang

Project Overview

In this text-mining project, we select Twitter as our data sources. As Halloween Holiday is approaching, we use "Halloween" as the keyword to filter and select related tweets and to perform certain text analysis on them. The techniques we use are "Word Frequencies", "Natural Language Processing" (or "Sentiment Analysis"), and "Text Similarity." By analyzing the word frequencies and the similarity between posts, we expect to learn what people usually post for Halloween Holiday and whether people post similar information. By conducting sentiment analysis, we hope to further evaluate people's attitude toward this holiday.

Implementation

Before we start doing the analysis, we first utilize Twython to gather a hundred Twitter posts that meet our conditions into a text file as a list, with all special characters and slang excluded. In order to extract the frequencies of words from all the tweets, we split tweets into words and put them into a dictionary, containing each unique word and its frequency of showing up as the value. In addition, we create a "stopwords.txt" file that contains all the common words, such as "the", "and", "I", and etc. By running this file we can exclude common words from Twitter posts since we believe common words tend to have higher frequencies that will impact our analysis results. A new list is generated for the left words that relate to Halloween's topic. The last step is to use a function to print out each unique word and its corresponding frequency in descending order.

When performing the analysis on "Text Similarity," we initially want to employ the method of Cosine Similarity. However, when we research it online, we find it hard to understand and we use an alternative way, which is called Jaccard Similarity. In doing Jaccard Similarity, we count the number of same words and divide it by the number of all unique words in two tweets. The result is a ratio that implies the degree of similarity between two posts. The higher

ratio means a higher degree of similarity and vice versa. In order to make this analysis more meaningful, we randomly select two tweets to compare as one simulation, and we run total 1,000 simulations and calculate the average Jaccard Similarity ratio.

For sentiment analysis, we use SentimentIntensityAnalyzer to calculate sentimental scores for each tweet and then assign each sentimental score into a new list. We then calculate the average score for each sentiment using lists values and keep the average results in two decimal places.

Results

In our word frequency analysis, below is 10 most frequent words and their frequency:

- halloween, 90
- happy, 8
- candy, 7
- room, 6
- party, 6
- like, 6
- ideas, 6
- costume, 6
- costumes, 5
- video, 4

It is not surprising to see that “halloween” is the most frequent word. But it is interesting that the following words do not have that many frequencies (less than 10), which indicates that the topics people discuss during Haloween season are still relatively diverse. With that in mind, most words are quite generic, such as “happy,” “candy,” “party,” “like,” and “costume/costumes.” But words “room,” “ideas,” and “video” are slightly out of our expectation when we think of Halloween related words.

In addition, our similarity results approximately range from 0.03 to 0.05 due to random selection differences. The low similarity implies that people are tweeting more unique

topics/content instead of just saying “Happy Halloween.” We did not expect that because we already narrow down the keyword to “Halloween,” so we thought tweets should have high similarity in general. But at the same time, we think the result makes sense considering our low-frequency result in word frequency analysis.

In the sentimental analysis, average scores are displayed below:

- Negative score: 0.07
- Neutral score: 0.80
- Positive score: 0.13
- Compound score: 0.11

Since it is Halloween Holiday, not surprisingly, the average negative score is super low. However, the neutral score is much higher than the positive score. We think the reason might be that people might use tweets to post announcements such as party information or video-sharing instead of expressing personal feelings, thus resulting in the high neutral score. But still, we believe if the number of tweets analyzed becomes larger, the positive score should be slightly higher than the current score.

Reflection

The beginning of the project is the hardest. Harvesting text from Twitter and writing tweets into a new file is new to us. Moreover, cleaning special characters, emoji, and slangs takes us plenty of time. To do that, we perform some research and keep trying different codes until we get the ideal result. After mining text from Twitter, things are getting easier. The word frequencies part is similar to one of the exercises that we did for homework and therefore we quickly figure out the code to find the frequencies of each unique word. The only difficulty we meet in this part is to delete the stopwords from the original tweets because when we compiled stopwords from a text file into a list, .strip() did not succeed in getting rid of ' \n'. It takes a long time for us to figure out the appropriate code, .replace(). For the “Text Similarity” technique, the coding process goes smoothly since we choose to use the Jaccard Similarity method. The process

of doing sentiment analysis is relatively smooth. We easily find out how to extract each sentimental score and put them in each list.

When finalizing the code, we meet trouble. Our code is extremely disorganized with lots of useless and not functional code in comments. Therefore, it takes us some time to reread and retest our code in order to save the correct one. This is something we should pay more attention to in the future. Next time, after we finish coding one part, we should refine the code and make it clean and clear. In this way, we won't waste unnecessary time and effort in finalizing code at the end.

Julie and I work efficiently on this project. Before we get started to code, we discuss what data sources that we are going to use and what aspects of our content that we are going to analyze. On the part of mining text from Twitter posts and clean tweets, we work together, with one computer used to code and one computer used to do research. In the analysis part, Julie selects "Word Frequencies" and "Sentiment Analysis" to work on and I work on the "Text Similarity." Although we separate the task, we work in the library together. Therefore, we can help each other when either of us meets the problem. Lastly, we create a google doc to write the project report. Overall, we don't think there are any issues related to the team process.