Module 6: Text Mining

Case Study

Things you will learn in this case-study:

1. Text Mining using Bag of Words Approach
2. Word Stemming
3. Text Visualization using Word clouds
4. Visualizing Positive and Negative Tweets

Back Ground:

Text Analytics has become imperative in any analytics driven organization. Companies are using social media analytics to understand the voice of their customers. One of the most influential such sources come from Twitter, which is one of the Top 10 most visited sites on the internet.
In this case study, we'll be trying to understand sentiment of tweets about one such company Apple. This is where they require your support.

Overview of the problem:

In this project, you will play the role of Data Scientist for Apple, and you have been given the task to carry out text processing and use visualization to understand the sentiments about the company, as expressed in twitter. Post this, the company will make changes in its social media strategy, if required.
The Data you will be dealing contains tweets about apple products and sentiment rating. The data is in data frame format which has been stored in the file named 'tweets.csv'

Data set description:

Tweet: tweet about apple as expressed on twitter
Avg: average sentiment rating given to the tweet ranging from -2(Strongly negative) to +2 (strongly positive)

Objective:

Based on the knowledge you acquired in Module, you are expected to do text analytics using Bag of Words Approach and carry out Visualization as the output.

You should do the following to create the above shiny output:

1. Load the required libraries and the data and call the dataframe 'tweets'.

2. Understand the data structure and provide concise summary on the following –
   - no of observations
   - total number of variables
   - structure of the variables
3. Setting up before text processing. Run the following lines of codes
   r1 = as.character(tweets$Tweet)
   set.seed(100)
   sample = sample(r1, (length(r1)))
4. Data Preprocessing using Bag of Words Technique
   - Create a Corpus - which, in simple terms, is nothing but a collection of text documents.
   - Now, remove punctuations
   - Next, change the case of the word to lowercase so that same words are not counted as different because of lower or upper case.
   - Next, remove numbers
   - Next, remove whitespaces
   - Now, remove unhelpful terms, also referred as stopwords
   - Now, please carry out the process of stemming, motivated by the desire to represent words with different endings as the same word.
   - create a document term matrix from the corpus
   - now create the data frame from the output of the above line

5. Now, create three different wordclouds using the following arguments
   - Create a word cloud and set random.order = TRUE:
   - Create a word cloud and set random.order = FALSE:
   - In the above word cloud, adjust the frequency level with min.freq parameter set at 5

Text Mining and Visualization for negative Tweets

1. Create a new dataframe from the original data 'tweets' which only includes negative tweets, where the Avg Value is less than zero. Name this dataframe as 'negativeTweets'

2. Now, Run the follwoing commands in your R Studio

   - n1 = as.character(negativeTweets$Tweet)
   - set.seed(100)

- sample2 = sample(n1, (length(n1)))

3. Next, run the following Data Preprocessing tasks
   - Create a Corpus
   - Remove punctuations
   - Convert to lowercase
   - Remove Numbers
   - Remove whitespaces
   - Remove stopwords
   - Perform Stemming
   - Create a document term matrix from the corpus
   - Now, create the data frame from the output of the above line

4. Now, create three different wordclouds using the following arguments
   - Create a word cloud and set random.order = TRUE:
   - Create a word cloud and set random.order = FALSE:
   - In the above word cloud, adjust the frequency level with min.freq parameter set at 5

Text Mining and Visualization for Positive Tweets

5. Create a new dataframe from the original data 'tweets' which only includes positive tweets, where the Avg Value is greater than zero. Name this dataframe as 'positiveTweets'

6. Now, Run the following commands in your R Studio
   - p1 = as.character(positiveTweets$Tweet)
   - set.seed(100)
   - sample3 = sample(p1, (length(p1)))

7. Next, run the following Data Preprocessing tasks
   - Create a Corpus
   - Remove punctuations
   - Convert to lowercase
   - Remove Numbers
   - Remove whitespaces
   - Remove stopwords
   - Perform Stemming
   - Create a document term matrix from the corpus

- Now, create the data frame from the output of the above line

8. Now, create three different wordclouds using the following arguments
   - Create a word cloud and set random.order = TRUE:
   - Create a word cloud and set random.order = FALSE:
   - In the above word cloud, adjust the frequency level with min.freq parameter set at 5

Submission should include the following:

1. Please follow the instructions and report all the outputs.
2. Summary on the approach should be documented.
3. R Code File.