**Title**

Use R to determine if a Tweet was negative, neutral, or positive. The tweets chosen were those about Donald Trump, Mother Teresa, and Barack Obama.

**Summary**

The solution analyzes the positivity of sentiments, then apply that positivity to tweets about popular persons. The aim was to use R to determine if a tweet was positive, neutral, or negative.

**Project Detail**

The original data (training.1600000.processed.noemoticon.csv), which was retrieved from <http://help.sentiment140.com/for-students/>, was used to train the model. It was renamed to TrainingData.csv, and was cleansed to remove invalid IDs.

Please read the Power Point file (Data Science with R.pptx) to understand the aim and process. Create the database and tables in SQL Server before running the code. The script to create the database is named CreateDatabase.sql. The script to create the tables in the database is named TablesToAnalyze.sql.

The data retrieval from Twitter (Donald Trump, Mother Teresa, and Barack Obama ) was inserted into the database. However, when the code is executed, new data will be retrieved. The new data can be inserted into the database. The new data must be inserted with only the following four columns; OpinionId (int) identity (1,1), userno varchar(50), diction (varchar151), ResponseDate (date). The extracted data from Twitter are in the csv files named; MotherTeresa.csv, BarackObama.csv, DonaldTrump.csv. The saved model after it was trained was named modelTrainingWithGlmnet.RDS.

**Project Tasks**

3,000 rows of data from Twitter for Mother Teresa, Donald Trump, and Barack Obama was retrieved. The retrieved tweets were inserted into the csv files, then inserted into the SQL Server database with SSIS.

The R program used the doc2vec deep learning algorithm. doc2vec draws context from phrases and is popular for movie reviews and sentiment classification. Text2vec will be used for sentiment analysis. Text2vec was written by Dmitry Selivanov.

709,376 tweets were collected in the file named TrainingData.csv. Those tweets were used to train a model, which was named modelTrainingWithGlmnet.RDS. The sentiment grading is from 0 to 1, therefore, .36 to .65 could be called neutral. The graph will use .5 for neutrality.

The data was split by using createDataPartition. A vocabulary-based vectorization (VBV) was done via preprocessing and tokenization, by using tolower, word\_tokenizer, and itoken. Also, the terms were collected and marked with unique IDs.

Document Term Matrix(DTM) was used to train the model, to find the frequency of terms. This was done by using create\_vocabulary, vocab\_vectorizer, and create\_dtm. The vectorizer function maps terms to indices, removes stop words and unusual terms.

A TF-IDF model was defined using tf-idf. TF-IDF is used to weigh the term frequency (word occurrence frequency) and inverse document frequency (idf decreases the weight for common words and increase the weight for uncommon words) of the data. To fit the model fit\_transform was used. The glmnet package was used for 5-fold cross-validation, using a L1 penalty and a small number of loops to reduce processing time. Prediction was done by the predict method.

The database connection was done by using the odbcDriverConnect method. The data was mapped to a data frame with dmat\_at.

The preprocessing, tokenization, vocabulary creation, DTM creation, data transformation with tf-idf, classification model loading, and prediction was done as before but on the new datasets. The same vectorizer that was used for the TrainingData.csv, was used for the new data sets. The probability that a randomly chosen positive case should outnumber a randomly chosen negative case, is shown by the area under the curve (AUC). The newly calculated weights were added to the new datasets.

**Challenges**

The data from <http://help.sentiment140.com/for-students/> had sentiment ratings, which were valuable to this analysis, therefore, the data was retrieved, cleansed, and used as the training data for this analysis. Loading the Twitter data into the database was a bit problematic. Therefore, to speed up development, SSIS was used to import the data from the Twitter csv files into SQL Server.

**Future Work**

Cleanse the data that was retrieved from Twitter, and modify the changing colors of the legend.

**References:**

Bryl, S. Twitter sentiment analysis with R. (2014). <http://analyzecore.com/2017/02/08/twitter-sentiment-analysis-doc2vec/>.

Bryl, S. Twitter sentiment analysis with Machine Learning in R using doc2vec approach (part 1). (2017). <http://analyzecore.com/2014/04/28/twitter-sentiment-analysis/>.