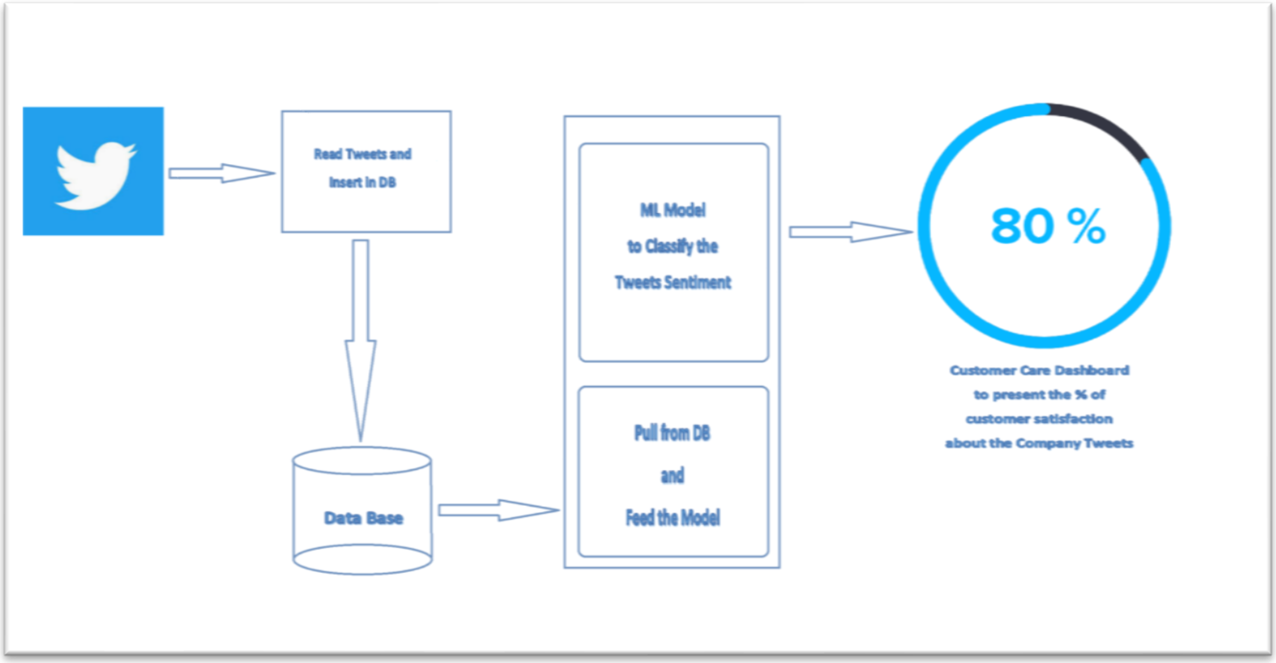
Project Documentation

Sentiment Analysis for Arabic Tweets

by Mohamed Alammary
moa2@Illinois.edu

1) An overview of the function of the code

The idea of the project was to build an application that measure customer satisfaction about the company's products by classifying the customers tweeter interactions in reply of the company's ads.



The model was planned to be implemented using scikit-learn and to be deployed on AWS SageMaker.

The delivery of the complete application end to end was not possible due to time constrains and the learning carve that was needed for AWS and scikit-learn. So I focused to deliver the code of the ML model to classify the tweets sentiment and decide if it is positive or negative.

2) Documentation of how the software is implemented

It was difficult to label data for training the model so I found a labeled data on UCI.

https://archive.ics.uci.edu/ml/datasets/Twitter+Data+set+for+Arabic+Sentiment+Analysis

The model was implemented using scikit-learn library on iPython Jupyter notebook. First, a split function is used to split the data into training and test splits using conventional 0.3 vs 0.7 ratio.

The next step was to convert the data to a BOW, I used scikit-learn CountVectorizer to create the bag of words representation.

Then I started to try many classification models. Also, I tried to improve the models by tuning the hyperparameters. First I tried LogisticRegression, and I got Training set score: 1.000, and Test set score: 0.688. Then I tried Multinomial Naive Bayes, I got Training set score: 0.98225 and Test set score: 0.73565. Lastly, I tried RandomForestClassifier, and I got Training set score: 0.99960 and Test set score: 0.67921.

Eventually, Multinomial Naive Bayes was the best model. I tried to improve the model by implementing TF-IDF and N-grams, using the scikit-learn TfidfVectorizer, but I did not notice much improvement, so I returned back to the normal CountVectorizer.

What I did NOT try yet is the use of the Stemming and lemmatization techniques as it was not supported in scikit-learn for Arabic. This is a potential area for improvement, but it requires much more time.

3) Documentation of the usage of the software (instructions on how to install and run a software).

1. Go to the URL: https://github.com/AlammaryMohamed/CS410-Text-Info-Systems on Github.
2. Download the repository on your local machine.
3. Install Anaconda python 3.7 version.
4. Open Anaconda console from the start window.
5. Go to the folder that you have downloaded the repository in.
6. Open Jupyter notebook by typing the command: Jupyter notebook
7. It will open Jupyter notebook on the web browser.

8. Open the code.ipynb script
9. Follow the steps.