National College of Ireland

Configuration Manual

Estimation of topics customers are interested in using tweets

MSc Research Project
Data Analytic

Mayur Kishor Mane
x15009009

School of Computing
National College of Ireland

Supervisor: Mr. Oisin Creaner

Contents

1 System Summary

The research study has been implemented using system with specific configuration which have discussed in this section. User has to configure system according to requirement mentioned below to run the python script for predictive topic modelling.

1.1 System Configuration

1. Operating System: Windows 10
2. RAM: 6 GB
3. Hard Disc: 500GB
4. Processor: Intel(R) Core(TM) i5 CPU M 480 @2.67GHz
5. System Type: 64-bit Operating System, x64-based processor

2 Getting Started

Getting started section explains how to download and configure the tools which are used in this research study. A brief explanation on tool setup and workflow have given below.

2.1 Tools Used

Tools used section shows the number of tools used in this research work for the predictive topic modelling. The tools used are listed as follows.

1. Microsoft Excel 2013
2. Python 3.4.4

2.2 Software Overview

This research project consists of programming environments and software tools. Python Version 3.4.4 is open source language used which is general purpose, high-level, interpreted, dynamic programming language. Below Python modules have used to achieve successful results in the research.

1. Natural Language Toolkit (NLTK)
2. Scikit Learn (sklearn)
3. BeutifulSoup
4. Pickle
5. Pandas
6. Numpy
7. Requests

These modules need to install prior running python script to find topic distribution. For installation please refer **section 4**.

2.3 Installation

The topic predictive model has implemented in Python version 3.x. and it does not give support to python version 2.x. User need to download python 3.x version from URL link. This should be installed on the device where research work is going to be used. Moreover, for installation guide please refer URL link.

2.4 System Menu

Twitter dataset used for this research study has been received from company Preceptiv, UK. This dataset is in CSV format and related to AdidasUK twitter account. The dataset was not useful due to unstructured format and hence feature engineering has been carried out as discussed in research paper. Below figure 1 helps to understand different variables present in the twitter dataset.

Topic	Subtopic	Tweet	Hashtag	Emojis
Sports	Sports/Cricket	RT @sachin_rt: Thanks for supporting		
Sports	Sports/Sports News	RT @BBCSport: It was #OnThisDay in	OnThisDay,bbcsnoo	
Sports	Sports/Soccer	RT @Cr7Prince4ever: This season:		
Sports	Sports/Soccer	RT @realmadriden: Tonight we face the	RMLiga	muscle,facepunch,
Sports	Sports/Soccer	@TwinB @1Xtra TRUST ME ... GOOD		
Sports	Sports/Soccer	RT @Arsenal: We are ready to go for the	AFCvWBA	
Sports	Sports/Formula 1	RT @svandoorne: Back in #Tokyo for a	Tokyo	jp
Sports	Sports/Soccer	RT @FIFAcom: Happy birthday to Edixon		

Figure 1: This is a sample CSV data of training set which have used to train the classifier. Mainly it contain 5 columns those are data variables used in data processing tasks.

These variables are discussed with domain experts and creator of this dataset to get insights. Below description is for data variables present in above figure.

1. Topics: It is label of tweet which describes the actual topic of tweet.
2. Subtopic: It is an additional label to get in depth information about a tweet.
3. Tweet: This data variable contains actual data which is unstructured with English and non-English words.
4. Hashtag: This variable created from Tweet variable which contains words those are started with symbol #.
5. Emojis: This data variable also created from Tweet variable where as it contains emotions which users have expressed in their tweet content.

All variables are processed and used in feature engineering to convert data into machine learning usable format. More specifically these variables converted into a large sparse matrix with dimensions of 1048576 columns and 1.2 million rows. Example of sparse matrix has shown below,

DTM	Term-1	Term-2	Term-3	Term-4	Term-5	Term-6
Tweet-1	0	1	1	0	1	1
Tweet-2	1	0	0	0	0	1
Tweet-3	0	1	1	0	0	1
Tweet-4	1	0	0	0	0	0
Tweet-5	1	1	0	0	0	1
Tweet-6	0	1	0	1	0	0

Figure 2: Tweet column of training set has been converted into numerical format as shown in this figure. This figure is known as document term matrix. If tweet-1 contain the term-1 (or word-1) in it then value 1 will be represented else if term-1 is not present then it will be represented as 0.

3 Design Workflow

AdidasUK twitter dataset divided into two parts viz. training set and test set. Training set has used to train the supervised machine learning classifier and test set has used to predict the topics on unknown tweets. Research design workflow has been shown in below figure,

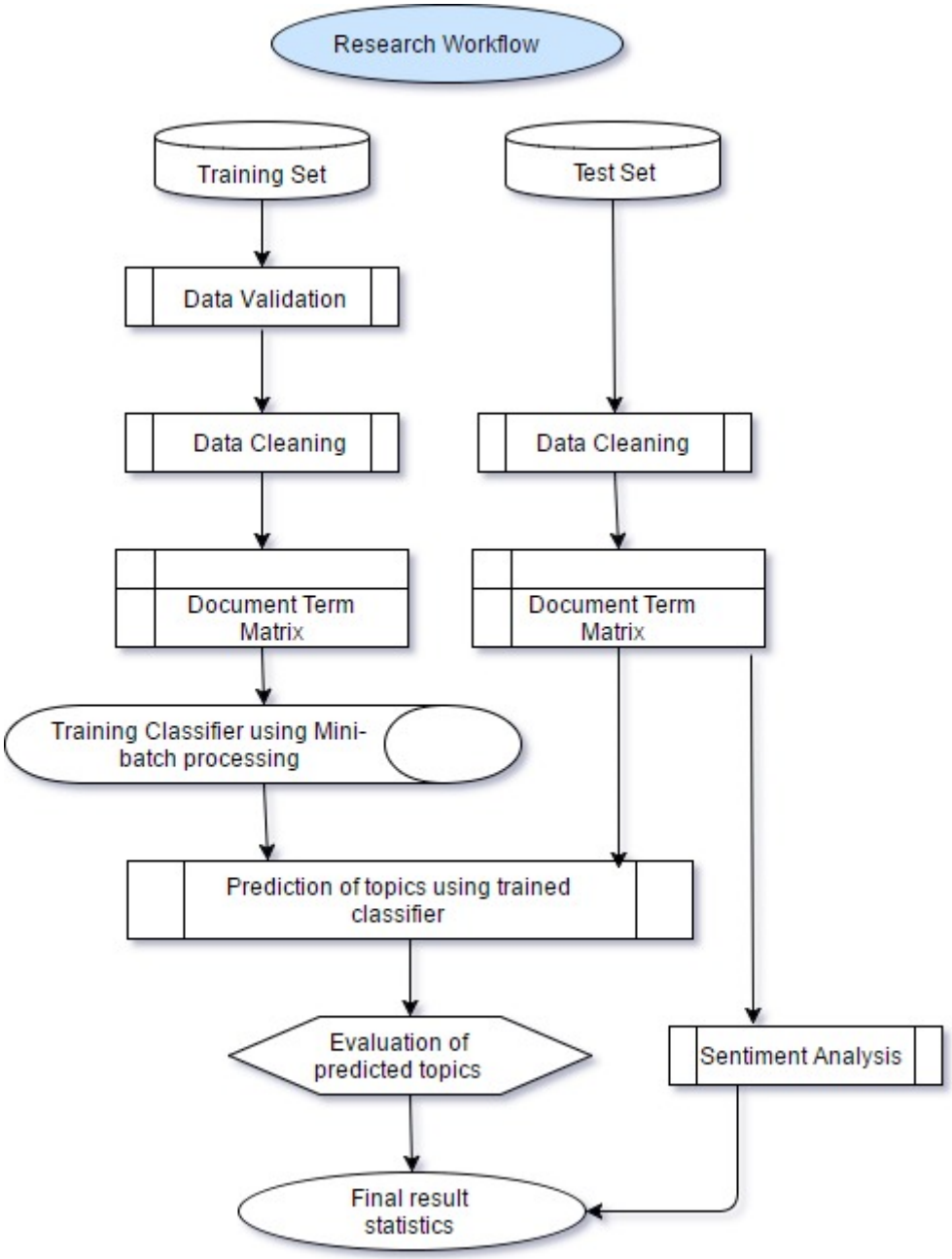

Figure 3: This figure represents research workflow. Two types of datasets used in the beginning viz. training set and test set. However, final results received in the end which present in CSV format.

Training set has been validated prior using in this research study. Validated data further cleaned and processed to get document term matrix (DTM). Document term matrix is a numerical representation of unstructured data. Similarly, we have cleaned and converted test set into DTM to get data into numerical format. DTM of training set has been used to train the supervised machine learning classifier and DTM of test set has been used to do prediction on unknown tweets. Also, we have carried out sentiment analysis on test set to understand user opinion. Finally, final result statistics received which gives topic distribution of tweets present in test set along with relative sentiments as shown in figure 4.

Subtopic	Percentage	Positive_sent	Negative_Sent	Neutral_Sent
Sports/Soccer	31.13	43.25	16.27	40.49
Art and Entertainment/Music	18.35	46.41	16.67	36.93
Technology/Internet Technology	10.32	47.84	10.32	41.84
Art and Entertainment/Social Media/YouTube	7.51	40.5	17.65	41.95
Art and Entertainment/Comedy	6.84	26.46	28.2	45.34

Figure 4: Trained SGDClassifier used against the test dataset to predict the subtopic labels. After predicting the subtopics, above final result has been calculated which shows that 31% tweets in test dataset are related to subtopic Sports/Soccer. Also, sentiment analysis of each topic has been calculated which have shown using percentage unit.

4 Using This System

To implement this research, different machine learning techniques have been used. These techniques consist of statistical computational methods to do predictive model building on training set and making predictions on test set. Advanced machine learning techniques used to make this research successful viz. SGDClassifier and MLPClassifier. These two techniques have explained below. Also, sentiment analysis carried out and implementation of this part has been explained below. Other than these machine learning techniques, NLTK python library used to carry out data pre-processing and cleaning.

1. NLTK provides interfaces to different corpora such as WordNet, stopwords along with text processing libraries. We have used stopwords corpora and for that we need to import stopwords from nltk.corpus submodule in python.

2. HashingVectorizer comes under Scikit Learn (sklearn), purpose to use this python module to convert unstructured data into structured format which is large sparse matrix as shown in figure 2. We have used below parameters to get DTM from HashingVectorizer.

   (a) analyzer = Word // To create features with each word
   (b) tokenizer = None // Words are already in tokenized form
   (c) preprocessor = None // Pre-processing done separately
   (d) stop words = None // Stop word list imported from NLTK
   (e) ngram_range = (1, 2) // To select ngram size 1 and 2
   (f) n_features = 2**20 // To select 1048576 features

   Other parameters are having default values.

3. SGDClassifier comes under Scikit Learn (sklearn) module, purpose to use this python module is to train classifier using Support Vector technique and mini-batch processing and to predict topics on test set. We have used below parameters to train the SGDClassifier.

   (a) loss = modified_huber // loss function for classification problem

(b) random_state = 42 // set seed as constant value. For repetitive process this value helps to get same result if the data processed is same for many iterations.

(c) n_iter = 1 // for partial_fit() method, number of iterations are one

(d) n_jobs = -1 // to use all cores of system and performing parallel processing

Other parameters are having default values.

4. MLPClassifier is again a submodule of python library sklearn. Purpose to use this python module is to train the classifier using neural network and mini-batch processing and to do prediction on test set. We have used below parameters to train the MLPClassifier.

   (a) activation: relu // it is a rectifier activation function which helps to pass non-negative value to the output.

   (b) solver: sgd // we have used sgd as a solver to work on same concept of stochastic gradient descent which helps to process large set of dataset.

   (c) learning_rate: adaptive // In neural network, we have to tune the learning rate so that classifier will correct the accuracy of a classifier automatically at highest possible value.

   (d) Hidden_layer: (200,) // We have used single hidden layer with 200 nodes to perform the computation for predicting different topics using neural network.

   (e) random_state: 42// set seed as a constant value. For repetitive process this value helps to get same result if the data processed is same for many iterations.

   Other parameters are having default values

5. SentimentAPI: We have used application programming interface (API) environment for performing sentiment analysis on test set. Mashap.com is an online developer community which contains enormous APIs for different software interfaces. We have used pre-built sentiment API from online developer community to carry out sentiment analysis. This is an open source API and we can send unlimited requests as per user requirement. To get the access to this API we need secret key and URL which we have already mentioned in the python script hence user does not need to do anything to carry out sentiment analysis part.

6. User need to install some python modules prior running python script and for that please refer below commands:

   (a) For windows users with python version 3.x
   pip install module_name

   (b) For Linux or Mac users with python version 3.x
   sudo pip install module_name

   For more details on installing python modules please go to URL link.

5 Executing the project

System is ready with above sections also it is understandable now that how algorithm is working with two classifiers. However, all files related to this research are uploaded to this URL link and explained in this section. User need to follow below steps to run the python script and perform predictive topic modelling using supervised machine learning technique.

Application execution procedure:

1. Keep all files starting with Adidas and python file SGDandPartialFit.py in one folder
2. Set this folder as a working directory
3. Verify all required python modules
4. Install all required modules before running the python script.
5. Open SGDandPartialFit.py file from directory in python Shell 3.x
6. Go to Code menu $\Rightarrow$ $Run \Rightarrow RunModule(F5)$
7. Wait till the end of execution
8. Open the working directory
9. Open final_result.csv which should be present in the working directory
10. Analyse the matrix containing topic and sentiment percentage distribution.
11. Finish