INFORMATION RETRIEVAL PROJECT

FINAL REPORT

Group Members:

Name	Email Address	Enrolment Number
Mano Vishnu A	am.vishnu@st.niituniversity.in	U101115FEC012
Harshitha Julakanti	Julakanti.harshita@st.niituniversity.in	U101115FEC017

Project Title: SENTIMENT ANALYSIS – TWITTER DATA

Project Definition:

Given a collection of tweets, classify them into four classes namely: positive, negative, neutral and mixed. The tweets used here are pertaining to two US presidential candidates namely: Barack Obama and Mitt Romney. By classifying the tweets into the mentioned classes we would be capable of predicting the opinion of the public and get a sense of the outcome of the election.

Techniques applied and Solution Methodology

1. **Approach:**

   Initially we have a number of tweets (approximately 7000 for each Obama and Romney) and their classes, in an Excel spreadsheet.
   These steps we take for classifying the tweets are as follows:
   1. Data Pre-Processing
   2. Feature Extraction
   3. Training
   4. Classification

2. **Tool Used:**
   Programming Languages used for the implementation is python. The NTLK libraries of Python provide a rich source of classification techniques which simplifies the task. We used the following modules of NTLK.
   1. nltk.stem for Lemmatization
   2. nltk.tokenize for tokenization
   3. nltk.classify for Naïve Bayes Classifier
   4. nltk.metrics for generating confusion matrix
   5. nltk.corpus for stop words

3. Algorithm Description:

A. Pre-Processing

Pre-processing of tweets involves the following steps:

- Removal of Hyperlinks: Used of Regular expressions to remove links as they do not provide essential information for classification.
- Removal of usernames: these are the items in the tweet that begin with ‘@’ character they are removed because we don’t consider opinion holders important for the classification of tweets ex: @narendramodi.
- Removal of hash character in hashtags: these are the items in the tweet that begin with ‘#’ character. They represent the topic of the tweet which in our case tends to be important. Hence we drop the ‘#’ character and preserve rest of the word. Ex: #voteformodi becomes voteformodi.
- Split camel case words: camel casing means two or more words merged into one such that the first word starts with a lowercase and subsequent words starts with a uppercase letter. Ex: voteformodi becomes vote For Modi.
- Removal of Annotations: they do not contribute towards classification and are hence removed ex: ‘<a>’ and ‘<e>’.
- Removal of special characters: All punctuations are removed. Except the single quote. The single quote is utilized later on while expanding worlds like “don’t” to “do not”. Ex: punctuations like! <>,%.
- Removal of digits: Digits are unimportant during the classification and are removed. Ex: Mano4456.
- Stripping off white spaces: additional white spaces are stripped off from the tweet.
- Fixing repeated characters: ex: “Gooood” to “Good”
- Conversion to lowercase: this helps in maintaining uniformity.
- Tokenizing tweets: the space separated tweets are tokenized to obtain tweets as a list of words.
- Expanding abbreviated words: Abbreviations are expanded to make words look more meaningful, ex: “lol” becomes laugh out loud, and “ppl” becomes people.
- Lemmatizing the tweet: each word undergoes lemmatization in order to obtain the root word. Ex: “winning” becomes “win”.
- Removal of stop words: These do not contribute towards the classification
- Removing duplicate words: as the frequency of the word in a tweet does not increase the class of the tweet it shall be removed.

B. Features:

The Individual features extracted from each tweet are tuples of the form: (tweet, sentiment). For each tweet extracted from the training set, the words in the tweet were assigned to the corresponding sentiment, thus resulting in 4 list of words – one for each sentiment. These features are fed to the train function to prepare to classify the training model. The test set is later fed to the classify function to classify the tweets and generate the accuracy, precision, recall, and F-score for each class / sentiment. The features used are unigrams thus ignoring the position of the words in the tweet. The tweet is instead considered as a “bag of words”.

C. **Learning Algorithm** : Naïve Bayes Classifier

Results and Conclusions:

- **Description of data**: The data considered for this project is a set of tweets about Obama and Romney and their corresponding sentiments. The set of tweets and their sentiments is provided in an excel file.
- **Precision, Recall and F-Score for the positive and negative classes:**

OBAMA:

Accuracy: 40.39%

	Positive	**Negative**	**Neutral**	**Mixed**
Precision	55.47%	46.61%	48.52%	28.40%
Recall	26.97%	48.11%	28.92%	63.39%
F-Score	36.30%	17.35%	36.24%	39.22%

Confusion Matrix

	Negative	**Neutral**	**Positive**	**Mixed**
Negative	331	93	38	226
Neutral	173	197	67	244
Positive	105	74	157	246
Mixed	101	42	21	226

ROMNEY:

Accuracy: 47.66%

	Positive	**Negative**	**Neutral**	**Mixed**
Precision	50.00%	56.18%	43.26%	37.94%
Recall	23.63%	63.43%	21.98%	63.88%
F-Score	32.09%	59.58%	29.15%	47.61%

CONFUSION MATIRX

	Negative	**Neutral**	**Mixed**	**Positive**
Negative	<609>	92	235	34
Neutral	235	<122>	164	24
Mixed	113	40	<329>	33
Positive	127	28	139	<91>

Issues faced in Model Solution:

- Removal of hash character in hashtags
- Removal of Hyperlinks
- Stripping off white spaces

Issues Faced in Development:

- Both team members are new to python and its libraries – have to learn python
- Installation and Import issues are faced with python modules
- Instead of using Dataset from kaggle tried to pull tweets from twitter in real time
- Twitter API Connection timed out Errors

Conclusion:

The classification was done on the provided test set with reasonable accuracy. We found that Naïve Bayes Classifier works well with the feature set that we generated from the tweets. In order to improve the results, we could have experimented with bigrams and collocations in feature sets.

References

- http://nltk.org/ - for documentation about NLTK libraries
- Bing Liu. "Sentiment Analysis and pinion Mining" ,May 2012. eBook: ISBN 9781608458851
- http://streamhacker.com/2010/10/25/training-binary-text-classifiers-nltk-trainer/ - for info on text classification methods
- http://www.ravikiranj.net/drupal/201205/code/machine-learning/howbuildtwittersentiment-analyzer - for info on tweet classification