Sentiment Analysis for Sarcasm Detection on Streaming Short Text Data

1- Over View:

The Proposed Model is made to detect sarcasm. The proposed system consists of three main levels which are:

A- Data Pre-Processing

B- Data Preparation

C- Sarcasm Detection

2- Proposed Model:

First the Pre-Processing Stage. The Pre-Processing stage consists of 3 Main parts which are , Hashtag identification, Emoji Replacement, Slang Dictionary Mapping. Starting with Hashtag identification in this stage they are identifying whether the words after the hash symbol is important and handy or not like #Sarcasm And #entertainment the first one helps the system that there could be sarcasm in this part by the second one isn’t helping too much. Moving on to the Emoji Replacement in this part they search for the emoji in the text and they replace it with weight Whether it is positive Weight of negative one. Moving on to Slang Dictionary in this phase they are searching for the slang and match it with slang from their dictionary and then they try replace the slang in the document with the phrase found in the dictionary.

Second the Data Preparation Stage. the Data preparation stage consists of 6 layers which are, Word Tokenization , Part of Speech , Stemming and lemmitization, Feature identification, New Representation Creation. Starting with the Word Tokenization Stage in this part they are tokenizing the phrase to words , moving on to the part of speech they are searching in the adjectives if they found a lot of adjectives in the text then there will be most probable a sarcasm in the document as a lot of adjectives means that they are used to describe something. Stemming and lemmitization is used to make the word go back to its original form. Feature identification consists of multiple layers which are Blob polarity , Blob subjectivity, Capitalization, Positive sentiment , Negative Sentiment. After The Feature Selection Layer is done a new Representation is given by using all the features and their respective weights.

Third Sarcasm Detection stage. They used classifiers : Decision Tree , Random Forest , Gradient Boosting , Adaptive Boosting , Logistic Regression , Guassien Naive Bayes.

3- Data sets:

They got Twitter data set with manually labeling it.

4- Results:

With emoji and Slang Dictionary:

Random Forest: 79.44 , 80.93 , 77.49

Naive Bayes: 74.56 , 73.9, 75.1

5-Draw Backs:

The main Draw back I have seen in this document is that they remove the stop words which could lead to wrong classification because taking a message like it is silly, when they remove the stop words the only word is going to remain is silly which is going to be classified as sarcasm.