Using Some Machine Learning Algorithms for Emotion Tagging and Sentiment Analysis

Partha De

Abstract*— *The present task involves the machine learning based approaches to emotion tagging for Bengali Documents and Sentiment Analysis for English Documents. For the Bengali documents, all the unigrams and bigrams are considered as features for emotion tagging. The feature selection is done using point wise mutual information technique. To prepare training data, all the sentences of the documents are tagged manually with one of the Ekman's six basic universal emotion label (Happy=1, Sad=2, Anger=3, Disgust=4, fear=5, Surprise=6, other emotion=7). Point wise mutual information of all the features are calculated by calculating the number of occurrences in a particular emotion category. The unigrams and bigrams that have point wise mutual information greater than a certain threshold value are considered as features. The feature matrix for the sentences with their emotion labels is calculated to prepare the training data. For emotion tagging or sentiment analysis, we train a number of machine learning algorithms chosen from WEKA, which provides a collection of machine learning tools. For performance evaluation, 10 fold cross validation is done and the final accuracy is calculated after averaging the results over all 10 folds. The average best accuracy obtained for emotion tagging is 55.89%. For sentiment analysis, we have used the bench mark datasets for experiments. Mutual information has also been used for feature selection for sentiment analysis. For sentiment analysis on the bench mark datasets, the average best accuracy obtained is 89%.

Keywords*—*point wise mutual information, naïve bayes multinomial, weka, emotion tagging, sentiment analysis

I. INTRODUCTION

In a Bengali or English sentence of a document, there may contain some emotions. Each sentence may express one of the Ekman's six basic emotions such as Happy, Sad, Anger, Disgust, Fear, and Surprise. The emotion of a sentence depends on the words of the sentence. From the previous works of emotion tagging on Bengali blog data [1], we get that the emotion of a sentence depends on whether the words are colloquial words or foreign words or reduplication words or emotion words. From [1] we also get that the emotion of a sentence also depends on whether the sentence has quoted symbols or special punctuation symbols or question words. From [1] we also get that the emotion of a sentence depends on the parts of speech information of the words. But as we are working on Bengali sentences of a document taken from newspaper and a Bengali story of "Rabindra Nath Tagore" the above mentioned features gives low accuracy. So, we use unigrams and bigrams of the words as features. The aim of our thesis is to design a system that can classify the emotion of a sentence depending on the above features of the sentence.

All the sentences of the training data are tagged manually with one of the Ekman's six emotion tags or other emotions. Then, all the unigrams and consecutive forward bigrams of the training data are taken. Here, these are called features. Next, point wise mutual information is used to reduce features. With a computer program the feature matrix is calculated which calculates the number of occurrences of the reduced features for every sentence with the corresponding emotion tags. Next, this feature matrix is converted to Weka (http://www.cs.waikato.ac.nz/ml/weka) format and then is run in Weka. Weka is a machine learning tool. Here, the accuracies of the proposed method can be seen by the correctly classified instances. In the present task, Bengali documents are collected from the well known news paper "Ananda Bazar Patrika" and a short story named "Dena Paona" of world famous Bengali poet "Rabindra Nath Tagore". For sentiment analysis of the English datasets, same procedure is followed and the datasets are taken from the website (http://www.cs.jhu.edu/~mdredze/datasets/sentiment).

II. RELATED WORK

In this approach, a corpus is prepared for emotion tagging and sentiment analysis. Now, machine learning algorithm is applied in emotion tagging and among the feature there is a feature named Senti Word Net emotion word. Here, every words are checked whether they are present in Senti Word Net[2]. Words that are present in Senti Word Net (Bengali) are supposed to contain emotions. It is important to differentiate between emotion words and non-emotion words. Word Net Affect Lists (Bengali) is prepared from English Word Net Affect Lists using English to Bengali bilingual dictionary [1].

Dipankar Das and Sivaji Bandyopadhyay (2011)[3] has shown that the Conditional Random Field (CRF)- based classifier performs better than the SVM(Support Vector Machine) classifier in the case of the document level emotion tagging of Bengali document and in word level emotion tagging SVM works better than CRF. Alm et.al [4] has shown that emotion tagging from text can be done by a method that learns from the feature. Here, a machine learning approach is followed where some features are used to classify the emotion of a sentence. Dipankar Das and Sivaji Bandyopadhyay(2010) [5] has shown that identifying emotional expressions, intensities and sentence level emotion tagging can be done using a Support Vector Machine (SVM) based supervised framework. Das, D. & Bandyopadhyay, S. [1] have used Conditional Random Field (CRF) for emotion tagging in Blog and News data at Word and Sentence Level. Xu et. al [6] (2007) has shown that feature selection for text categorization can be done by document Frequency thresholding (DF), information gain

Revised Version Manuscript Received on May 03, 2016.

Partha De, Department of Computer Science & Engineering, Birbhum Institute of Engineering & Technology, Suri, Birbhum, West Bengal, India.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication Pvt. Ltd.

(IG), mutual information (MI) and IG is best methods whereas MI has lesser efficiencies.

III. PROPOSED METHODOLOGY

Our proposed system has several components:

A. Emotion Tagging

1) Tagged Sentences:

We prepared training data using 29 Bengali documents from the newspaper named "AnandaBazar Patrika" and one Bengali short story by "Rabindranath Tagore" from the website www.rabindra-rachanabali.nltr.org (a Bengal Engineering & Science University, Indian Institute of Technology –Kharagpur initiative). Then, these sentences were tagged manually with emotion tag (Happy, Sad, Anger, Disgust, Fear, Surprise, other emotions). Next, stemming of each word of the training data is performed using Indian Statistical Institute, Kolkata stemmer (yass-yet another stemming software).

2) Feature Extraction: All unigrams and all consecutive forward bigrams of the words of training data are generated.

Algorithm for Feature Extraction:

```
Feature_List = Empty
For each tagged sentences in the training data
    Step 1: Generate unigrams. The words of the sentences
    are the unigrams of that sentence.
    Step 2: Add these unigrams to the Feature_List
    Step 3: Generate consecutive forward bigrams. The
    consecutive words of the sentences are concatenated to
    generate the bigrams of that sentence.
    Step 4: Add these bigrams to the Feature_List
    Step 5: Loop until the training data complete
    End.
    Feature_List is the total number of unigrams and
    consecutive forward bigrams.
```

3) Feature selection: Feature selection is done using point wise mutual information (PMI) which is explained below:

Point wise mutual information of all unigrams and all bigrams are calculated which are considered as features for our tasks.

a) Point wise Mutual Information:

Text categorization (TC) is the process of grouping texts into one or more predefined categories based on their content.

Given a category c and a term t, let A denote the number of times c and t co-occur, B denotes the number of times t occurs without c, C denotes the number of times c occur without t, and N denotes the total number of documents in c(Here, N is the number of sentence in a category say 'Happy'). The point wise mutual information criterion between t and c is defined as:

$$PI (t, c)= \log [p(t \wedge c) / (p(t) * p(c))]$$

and is estimated using:

$$PI (t, c) \approx \log [(A * N) / ((A+B) * (A+C))]$$

This PI (t, c) for each word is calculated for each category of emotions (Happy, Sad, Anger, Disgust, fear, Surprise, other emotions).

b) Add-one Smoothing:

The smoothing of (m / n), if m, n ≈ 0, is equal to ((m + 1) / (n + c * 1)), where c is the number of category. As we know that the value of m, n may be zero, one is added to m and c * 1 is added to n to avoid multiplication and division by zero.

Therefore, $PI (t, c) = \log [((A + 1) * N) / ((A+B + 7) * (A+C))]$

Because, the number of emotion category is 7.

After the computation of these criteria, thresholding is performed to achieve the desired degree of feature elimination from the full vocabulary of a document corpus.

In a general way, point wise mutual information as defined above compares the probability of observing t and c together (the joint probability) with the probabilities of observing t and c independently (chance). If there is a genuine association between t and c, then the joint probability P (t,c) will be much larger than chance P(t) P(c), and consequently PI(t,c) >> 0. If there is no significant relationship between t and c, then P(t,c) ≈P(t)P(c), and thus, PI(t,c) ≈0. If t and c are in complementary distribution, then P(t,c) will be much less than P(t) P(c), forcing PI(t,c) << O. That is, point wise mutual information as defined above can be negative.

Next, point wise mutual information of all unigrams and all bigrams (features) for all emotions categories are calculated. Then, we fix a threshold value and the features whose point wise mutual information value is greater than the threshold value are taken as current features set.

4) List of Features: Thus, with the help of point wise mutual information, the number of unigrams and bigrams (features) are reduced. These reduced features are then used to generate feature matrix.

5) Feature matrix Calculation: With this reduced feature set we calculate a feature matrix whose column are the words that have point wise mutual value greater than a certain threshold value. The rows of the feature matrix are the sentences of the training data. Here, for a particular row, the number of occurrences of the words of features set in that sentence is taken as value for that position in the feature matrix.

6) Vector Labeler: Here, emotion tags of all the sentences are placed in the Feature matrix and labeled vector is generated. In the labeled vector, for a particular row, the number of occurrences of the words of the reduced features along with the corresponding emotion label is present.

7) Create File in WEKA format:

From this feature matrix, a weka(3.6.11) file (.arff) file is generated. Weka is a machine learning tool of Department of Computer Science, University of Waikato, New Zealand.

8) Learning Algorithm Chosen from WEKA: The classifiers used in weka are Naïve Bayes, Naïve Bayes Multinomial, and Naïve Bayes Multinomial Updateable (bayes). Here, as the number of features are large in number (or the feature matrix is a sparse matrix), Naïve Bayes Multinomial, and Naïve Bayes Multinomial Updateable are chosen [8].

9) Model: Now, the feature matrix for training portion is used for training and feature matrix for test portion is used for test in model. From here, performances of the proposed system are the correctly classified instances.

10) Labeling untagged sentences: When weka format file is run in weka, weka splits this file into two portions- training and test. From the training portion, weka is being trained and the test portion is used for testing. Now, logically let us assume that the training portion is unlabeled. Following the previous procedure, feature matrix is calculated for the test portion also.

B. Sentiment Analysis:

For this, datasets have been taken from the website http://www.cs.jhu.edu/~mdredze/datasets/sentiment. There are 4 types of datasets named BOOKS, DVD, KITCHEN and ELECTRONICS. The number of sentences in that datasets is shown in table - 1.

Table - 1: Number of sentences of the training and test data of Sentiment Analysis.

Data	Training	Test
BOOKS	4465	2000
DVD	3586	2000
KITCHEN	5941	2000
ELECTRONICS	5943	2000

The size of training data and test data (file type = Unicode, plain text) in that datasets is shown in Table - 2.

Table - 2: Size of the training data and test data of Sentiment Analysis.

Data	Training (in MB)	Test (in MB)
BOOKS	23.3	11.0
DVD	18.9	10.5
KITCHEN	16.8	5.68
ELECTRONICS	18.8	6.72

In this datasets every sentence has one of the two sentiment label (negative=1, positive=2). Similar procedure is followed here.

IV. RESULTS

A. Results on Emotion tagging

1) Experiment No. 1(point wise mutual information (PMI) of unigrams and bigrams + no stemming is applied): Here, at the beginning all unigrams and all consecutive forward bigrams are taken as initial features set. Next, point wise mutual information is performed on these features. The threshold value of the point wise mutual information is -1.25. The results is (Bayes.NaiveBayesMultinomial + (PMI is used)) = 50.99%.

2) Experiment No. 1A (Unigrams + Bigrams + No point wise mutual information is used + no stemming is applied): The results is (Bayes.Naive Bayes Multinomial + (PMI is not used))= 45.56%.

3) Experiment No. 2 (point wise mutual information of unigrams + no stemming is applied): Here, at the beginning all unigrams are taken as initial features set. Next, point wise mutual information is performed on these features. The threshold value of the point wise mutual information is -1.2525. The results is (Bayes.NaiveBayesMultinomial)= 49.32%.

4) Experiment No. 3(stemming + point wise mutual information): Here, at the beginning all unigrams and all consecutive forward bigrams are taken as initial features set. And, stemming is performed by Indian Statistical Institute, Kolkata for each word of the training data. Next, point wise mutual information is performed on these features. Then, different threshold value is used to get different type of feature size. The Table shows the results (Bayes.NaiveBayesMultinomial) for different threshold value.

Table: 3 Results of Emotion Tagging for different threshold value.

Threshold value	Results
-1.36	52.76%
-1.2525	47.34%
-1.25	54.32%
-1.12	55.26%
-1	55.68%
-0.92	55.89%
-0.88	55.37%
-0.84	49.11%
-0.75	47.2%

B. Sentiment Analysis

1) Experiment No. 4: Here, the training file has one of the following labels (1, 2). There are 2 types of files- training file and test file. Both the training file and test file has same types of label. Here, bigger file have been taken as training file and relatively smaller file as test file. In this training data and test data the words along with their number of occurrences as well as corresponding label are supplied. So, there is no need to generate all unigrams and all bigrams from the supplied training and test data. At first, all distinct words are generated from training data. Next, point wise mutual information is calculated for all these words for all categories. Next, we fix a threshold value and the words whose point wise mutual information value is greater than that of the threshold value are considered for feature set in the feature matrix calculation. As this is to some extent big data, so doing experiments with different threshold value and then generating feature set and finally calculating feature matrix will be difficult and computationally intractable. Therefore, the following procedure is followed:

Previously, we find that among the all unigrams and all bigrams, a large portion of unigrams and bigrams are present in 1 category (or cluster) and are absent in others category (or cluster). Some others unigrams and bigrams are present in most of the categories or are present in 2 categories. Now, we have to fix manually a threshold value such that we can just ignore (or bypass) those words (unigrams and bigrams) that are present in 1 category and are absent in another category, then we shall get a very small amount of features. This can be done as follows: Let us consider the values of point wise mutual information of the words (those are present in 1 category and absent in others categories) for all categories and then, find the biggest value among these values of point wise mutual information. Now, we fix the threshold value as little bit bigger value of this biggest value.

Now, with this threshold value, feature set are generated by computer program in Visual Basic 6.0. With this feature set, a computer program is written that calculate feature matrix for training data and from this feature matrix a computer program is written that generate weka file (.arff file, ANSI). Similarly, feature matrix is calculated for test data for the same feature set and from this feature matrix weka file is generated for test data. These weka files are run in Weka 3.6.11 by command prompt to avoid out of memory in weka. The results for different types of datasets are given below:

Table - 4 shows the performances of classifiers in weka for the datasets of sentiment analysis.

Table - 4: Performances of classifiers in weka (point wise mutual information + threshold value = -0.01) in Sentiment Analysis.

Datasets	Results (%) on Naïve Bayes Multinomial
KITCHEN	89.3
DVD	84
ELECTRONICS	88.4934
BOOKS	85.5

V. SOME EXPERIMENTS THAT GIVES NEGATIVE RESULTS

B. *Experiment No. 5(Machine Learning Approach + Model: 1)*

1) The emotion of a sentence depends on the following features:

□ Parts Of Speech (POS) information (adjective, verb, noun, adverb),

□ Words of the title sentence or the first sentence of the document,

□Bengali Emotion words: There are some emotion words in Bengali. Some words express happiness, some words expresses sadness. The emotion state Happy, Sad, Anger, Disgust, Fear, and Surprise are represented by emotion words.

□ Reduplication words : Figure – 1 shows the examples of reduplication words.

ভাঙাচোরা রীতিমতো বড়সড় বড়বড়

Figure – 1: Examples of reduplication words.

□ Question words : Figure – 2 shows the examples of question words.

কী, কেন, কেমন, কীভাবে, কি, কোথায়

Figure – 2: Examples of Question words.

□ Colloquial words: Colloquialism is a word, phrase or other form used in informal language. Colloquialism is related to, but not the same as slang. Slang is permitted in colloquial language, but it is not a necessary element. Figure – 3 shows the examples of colloquial words.

চাঁই লাথি খোঁয়াড় বস্তি ভাগাড়

Figure – 3: Examples of Colloquial words.

□Foreign words: The words which have come from other language like English, French are called foreign words. Figure – 4 shows the examples of foreign words.

খুবসুরত বলিউড রিমেক ক্লাস বেদম

Figure – 4: Examples of Foreign words.

□ Special punctuation symbols: Figure – 5 shows the examples of special punctuation symbols.

- ! ? , ‘ ’ ’ ; ( ) :

Figure – 5: Examples of Special punctuation symbols.

□ Quoted sentence: Figure – 6 shows the examples of Quoted sentence.

তিনি বলেন, " অস্ত্রে যুদ্ধ অনেক হয়েছে।

Figure – 6: Examples of Quoted sentence.

□ Sentence length: The number of words of a sentence is the sentence length. (>=8, <15),

2) *Corpus Preparation:* We prepared corpus for different types of words like reduplicated words, foreign words, emotion words, Question words, Colloquial, Quoted words, Special punctuation symbols.

3) *Procedure:* The training data by giving document number, sentence number, Bengali sentence and corresponding emotion label was prepared. Next, a computer program that calculates the feature matrix wax written. In the feature matrix, emotion words, foreign words, and colloquial words, reduplication words have been taken as a binary feature (1 for presence and 0 for absence). And, for the question words, quoted symbols and special punctuation symbols, instead of taking whether these words are present or not, which of these words are present are considered in feature matrix. In the feature matrix for all sentences the values of the features are calculated and saved in a file. The format of feature matrix is shown in Table – 5.

Table –5: Format of feature matrix (Machine Learning Approach + Model: 1).

0	1	2	3	4	
Document number	Sentence Number	Colloquial words	Emotion words	Foreign Words	
5	6	7	8	9	10
Question words	Quoted Symbol	Reduplication words	Special Punctuation symbols	Length of sentence	Emotion label of the sentence

Then, this feature matrix is converted to weka format file (.arff file) by giving attribute and relation name. Then this weka format file is run in weka. The results (Functions. Multilayer Perceptron) of weka file = 27.00%.

B. Experiment No. 6 (Parts of Speech) The number of parts of speech (Noun, Pronoun, Verb etc.) are also important in the emotion of a sentence. There are different types of parts of speech—Noun(NN),Proper Noun(NNP), Pronoun(PRP), Demonstrative(DEM), Verb-finite(VM), Verb Auxiliary(VAUX), Adjective(JJ), Adverb(RB), Post Position(PSP or NST), particles(RP), Conjunction(CC), Question Words(WQ), Quantifiers(QF), Cardinals(QC), Intensifier(INTF), Interjection(INJ), Negative(NEG), Symbol(SYM), Reduplication(RDP), Compound(XC), Unknown(UNK).

1) Question words:

In Bengali question words are considered as Pronoun.

2) Demonstrative: Demonstrative are those the speaker refers to. Demonstrative are used to indicate some entities. Demonstrative point to a entity that is currently being said or was said earlier by author. Figure – 7 shows the example of Demonstrative.

এ এই ওই ঐ

Figure – 7 : Example of Demonstrative.

3) Cardinals: The numbers are called cardinals. Figure – 8 shows the example of Cardinals.

৪৮টি চৌত্রিশের এক ১০

Figure – 8: Example of Cardinals.

4) Quantifiers: The words that are used to quantify the noun are called quantifiers.

5) Negatives:-These are negative words.

6) Compounds:-These are noun-noun compound.

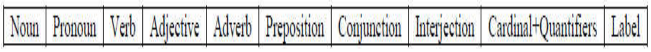
The identification of Noun Noun Compound is required for the corpus preparation in our present task. Vivekananda Gayen, Kamal Sarkar (2013) [9] has shown automatic identification of Bengali Noun-Noun Compounds can be done by Random Forest. Tanmoy Chakraborty [10] has shown that identification of Noun-Noun (N-N) collocations as multi-word expressions in Bengali corpus can be done by unsupervised approach with various statistical measures.

7) Parts of Speech Tagging:

Dandapat et, al(2007) [11] has used Hidden Markov Model (HMM) and Maximum Entropy (ME) based stochastic taggers for Part-of-Speech (POS) tagging for Bengali. Kamal Sarkar, Arup Ratan Ghosh(2013) have used Memory Based Learning (MBL) techniques for Bengali POS tagging. For parts of speech tagging we submit every sentence one by one to the pos tagger of International Institute of Information Technology, Hyderabad (http://ltrc.iiit.ac.in/analyzer/Bengali/) (Language Technology Research Centre). This gives POS as well as chunking information of a sentence.

Then, the POS information of a sentence is considered as features in sentence label emotion tagging. Here, the number of noun, pronoun, verb, adjective, adverb, conjunction, interjection, (cardinals + quantifiers) are calculated. Table – 6 shows the format of feature matrix when Parts of Speech Tagging is used.

Table – 6: Format of feature matrix (Parts of Speech Tagging).



Noun	Pronoun	Verb	Adjective	Adverb	Preposition	Conjunction	Interjection	Cardinal+Quantifiers	Label

The results(Functions.MultilayerPerceptron) of weka file = 25.96%.

C. Experiment No. 7 (Unigrams) At first, all punctuations are deleted from training data. In this experiment all the unique words of the training data are considered as features. These words of all the sentences are called unigrams (features).

With this features set a Computer program that calculates feature matrix is written. The column of the feature matrix is the words of the feature set. And, the row of the feature matrix is the one by one sentence of the training data. In the feature matrix the number of occurrences of the words of feature set are calculated and saved in a file. From the feature matrix, weka file is generated and is run in weka. In our present task, the total numbers of unigrams are 3984.

The results(Trees.REPTree) of weka file =41.7%.

D. Experiment No. 8 (Unigrams & Bigrams) At first, all punctuations are deleted from training data. In this experiment all the unique words (unigrams) and all the consecutive forward bigrams of the training data are considered as features. These words of all the sentences are called unigrams and bigrams (features). With this features set, a Computer program that calculate feature matrix is written. The column of the feature matrix is the words of the feature set. And, the row of the feature matrix is the one by one sentence of the training data. In the feature matrix the number of occurrences of the words of feature set are calculated and saved in a file. From the feature matrix, weka file is generated and is run in weka. In our present task, the total numbers of unigrams and bigrams are 12677.The results(Bayes.NaiveBayesMultinomialUpdateable) of weka file =45.39%.

E. Experiment No. 9 (Unigrams & Bigrams and MALT parser) Previously, bigrams of the training data was used. But, the number of words in the bigrams is very large. To reduce the number of bigrams dependency parser are used. The words of the sentence are dependent on each other. Figure – 9 shows dependency between words in a sentence.

Adjective <-------------------- Noun
Verb ----------------------- > Adverb
(Dependency)
(Dependency)

Figure – 9: Dependency between words in a sentence.

Using dependency parser, the dependency information of the training data is found. The words having same dependency information are used to generate bigrams. For dependency parser, Indian Statistical Institute, Kolkata (http://www.isical.ac.in/~utpal/resources.php) and MALT (Models and Algorithms for Language Technology Group – Vaxjo University and Uppsala University, Sweden) are used. Bengali_stack.mco is downloaded from the website of ISI, Kolkata . Then, a file named maltparser-1.7.1 is downloaded. To use MALT parser, a special type of input file named inputfile.conll is used.

This ISI, Kolkata malt parser takes parts of speech (POS) tagging and chunking information as input.

For POS tagging, Indian Statistical Institute, Kolkata POS tagger is used. This POS tagger consists of Stanford-postagger (1,471 KB) and Bengali ISI tagger

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication Pvt. Ltd.

(bengaliModelFile.tagger – 1,163 KB). Then, the training data is given as input and the output obtained is shown in Figure -10. Stanford-postagger is a probabilistic parser of Stanford University.

ভেঙে VM পড়া VM বিমান NN থেকে PSP বেঁচে VM ফিরলেন VM তরুণী NN

Figure – 10: Output of (Indian Statistical Institute, Kolkata and Stanford University) parts of speech tagger.

For chunking information IBO (Intermediate, Beginning and Outside) format is used. The shallow-parser-Bengali is used (http://www.ltrc.iiit.ac.in/analyzer/bengali) (Language Technology Research Centre) of International Institute of Information Technology, Hyderabad for chunking information of the training data.

Here the consecutive words those have same dependency tag or line number are taken for generation of bigrams. Some of the dependency tags of ISI, Kolkata are shown in Figure – 11.

k1 = karta (subject), k2 =karma (object), k3= karana (instrument), k4= sampradaana (recipient), k5 =apaadaana (source) ……….

Figure – 11: Dependency tag of the Indian Statistical Institute, Kolkata MALT parser (Dependency parser).

The selective bigrams generation from the malyparser-1.7.2 output is important. Because, it reduces the number of bigrams. The results(Bayes.NaiveBayesMultinomialUpdateable) of weka file =45.25%.

1) Dependency Parser: Dependency parser is a full parser. This parser determines the relationship between words of a sentence. This parser determines the dependency between the words in a sentence.

F. Experiment No. 10 (Unigrams & Bigrams & Trigrams – Model 1) Here, trigrams are generated using 3 words of a sentence. The following combinations are used:

1) Subjective + Verb (any place in a sentence) +Negative Word. Subjective means Noun (NN), Proper Noun (NNP), k1 (karta (subject) in ISI, Kolkata dependency tag).

2) Verb + Adverb (any place in a sentence) +Negative Word.

3) Verb + Adjective (any place in a sentence) +Negative Word.

4) Adverb + Adjective (any place in a sentence) +Negative Word.

5) Adjective + Noun (any place in a sentence) +Negative Word.

Negative words are not, no, never, can’t, don’t. The results(Bayes.NaiveBayesMultinomial) of weka file =41.91%.Figure – 12 shows the example of trigrams.

<ভাবেন নি হয়তো> <কেউ ভাবেন নি> <ছিল সোজা না>

Figure – 12: Example of trigrams of (Unigrams & Bigrams & Trigrams –Model 1).

G. Experiment No. 11 (Unigrams & Bigrams & Trigrams – Model 2) Here, trigrams are generated using 3 words of a sentence. The following combinations are used:

1) Subject + Verb (consecutive word or immediate neighborhoods in a sentence) +Negative Word. Subject means Noun (NN), Proper Noun (NNP), k1 (karta (subject) in ISI, Kolkata dependency tag). Object means Noun (NN), Proper Noun (NNP), and k2 (karma (object) in ISI, Kolkata dependency tag).

2) verb+ Object (consecutive word or immediate neighborhoods in a sentence) + Negative Word

3) Verb + Adverb (consecutive word or immediate neighborhoods in a sentence) +Negative Word.

4) Verb + Adjective (consecutive word or immediate neighborhoods in a sentence) +Negative Word.

5) Adverb + Adjective (consecutive word or immediate neighborhoods in a sentence) +Negative Word.

6) Adjective + Noun (consecutive word or immediate neighborhoods in a sentence) +Negative Word.

Negative words are not, no, never, can’t, don’t. The results (Bayes.NaiveBayesMultinomial) of weka file =42.3%. Figure – 13 shows the example of trigrams.

<ক্ষমতা ছিল না> <আহত ছিল না> <বিশ্বাস পারেন নি>

Figure – 13: Example of trigrams of (Unigrams & Bigrams & Trigrams –Model 2).

H. Experiment No. 12 (point wise mutual information and MALT parser) Here, point wise mutual information is applied on (all unigrams and bigrams generated by ISI, Kolkata malt parser). With the help of MALT parser, the numbers of bigrams are reduced. Now, point wise mutual information is applied on these reduced features. The results(Bayes.NaiveBayesMultinomial) of weka file =50.57%.In our experiment, the efficiency of my project with the help of MALT parser is little bit worse than all bigrams.

I. Experiment No. 13 (Machine Learning Approach + Model: 2) The format of feature matrix in this model is shown in Table – 7.

Table – 7: Format of Feature Matrix (Machine Learning Approach + Model: 2).

0	1	2	3	4
Document number	Sentence number	Number of Colloquial words	Number of Emotion words	Number of Foreign Words
5	6	7	8	9
Question words	Quoted symbol	Number of Reduplication words	Special Punctuation symbols	Length of sentence
Number of Noun	Number of Pronoun	Number of Verb	Number of Adjective	Number of Adverb
10	11	12	13	14

Number of Preposition	Number of Conjunction	Number of Interjection	Number of Negative words
15	16	17	18

Percentage(%) of (Noun + Verb + Adjective + Adverb)	First sentence in a document	Ranges of story progress	Emotion Label of the sentence
19	20	21	22

For 5th column identification of question words among the following symbols (কী কেন কেমন কীভাবে কি কোথায়) are done. For 6th column identification of quoted (direct speech) symbols among the following symbols (“ ” " ‘ ' ’) are done. For 8th column identification of special - punctuation symbol among the following symbols - ! ? , ; ( ) : are done. For 5th, 6th and 8th column which question words (or quoted symbols or special – punctuation symbols) is present is more important than the number of question words (or quoted symbols or special – punctuation symbols). For 19th column Percentage (%) of (Noun + Verb + Adjective + Adverb) is calculated by dividing total number of (Noun + Verb + Adjective + Adverb) by the total number of words in a sentence. For 20th column, first sentence in a document is a binary feature. If the sentence number of a document equals to 1, then the value of this feature equals to 1, otherwise the value of this feature equals to 0. For 21st column, ranges of story progress are calculated by dividing the sentence number of a sentence by the total number of sentences in that document. The results (Functions. logistic) of weka file =30.2398%.

VI. Conclusions

In our work, we have designed such a system that can classify the emotion of the sentence in Bengali documents. We have also designed another system for sentiment analysis of English documents. We have used machine learning algorithm for prediction task and point wise mutual information for feature selection. We have used unigrams and bigrams features for this task. Other features such as reduplication words, emotion words, and foreign words have also been tried but we did not obtain better results for emotion tagging on our Bengali datasets.

The performance of the proposed system may be improved in the following ways:

1) Introducing new features.
2) Using any other new feature selection technique.
3) Introducing new machine learning algorithm.

REFERENCES

1. Das, S. Bandyopadhyay, “Analyzing emotion in blog and news at word and sentence level,”.
2. Das, S. Bandyopadhyay, ”Word to sentence level emotion tagging for bengali blogs,” ACL-IJCNLP Conference Short Papers, Suntec, Singapore, pp. 149–152, 2009.
3. Das, S. Bandyopadhyay,”Document level emotion tagging: machine learning and resource based approach,” Computación y Sistemas, Vol. 15, No. 2, pp. 221-234, ISSN 1405-5546, 2011.
4. O. Alm, D. Roth and R. Sproat, “Emotions from text: machine learning for text-based emotion prediction,”.
5. Das, S. Bandyopadhyay, “Identifying emotional expressions, intensities and sentence level emotion tags using a supervised framework,” 2010.
6. Y. Xu, G. Jones, J. T. Li, B. Wang and C. Sun, “A study on mutual information-based feature selection for text categorization,” Journal of Computational Information Systems, 3:3,pp. 1007-1012, 2007.
7. M. Hu, B. Liu, “Mining and summarizing customer reviews,” In proceedings of the tenth ACM SIGKDD international conference on knowledge discovery and data mining, pp. 168-177, ACM, August ,2004.
8. McCallum, K. Nigam, “A comparison of event models for Naïve Bayes text classification”.
9. V. Gayen, K. Sarkar, “Automatic identification of Bengali Noun-Noun compounds using Random Forest,” 9th Workshop on Multiword Expressions (MWE 2013), pp. 64–72, Atlanta, Georgia,13-14 June,2013.
10. T. Chakraborty ,” Identification of Noun-Noun (N-N) collocations as multi-word expressions in Bengali corpus”.
11. S. Dandapat, S. Sarkar and A. Basu, “Automatic part-of-speech tagging for Bengali: an approach for morphologically rich languages in a poor resource scenario,” ACL, Demo and Poster Sessions, pp.221–224, Prague, June 2007.
12. K. Sarkar, A. R. Ghosh, “A Memory based pos tagger for Bengali,” 1st Indian Workshop on Machine Learning, IIT Kanpur, India, 2013.
13. Das, S. Bandyopadhyay, “Emotion tagging – a comparative study on Bengali and English blogs,” 7th International Conference on Natural Language Processing (ICON-2009), Hyderabad, India, pp. 177–184, 2009.
14. D. Das, S. Bandyopadhyay, “Developing Bengali WordNet Affect for analyzing emotion,” 23rd International Conference on Computer Processing of Oriental Languages, California, USA, pp. 35–40,2010.
15. D. Das, S. Bandyopadhyay,” Labeling emotion in Bengali blog corpus – a fine grained tagging at sentence level,” 8th WorshoponAsianLanguageResources (COLING-2010), Beijing, China, pp.47–55, 2010.
16. D. Das, S. Bandyopadhyay, ”Sentence level emotion tagging on blog and news corpora,” Journal of Intelligent System, 19(2), pp.145–162, 2010.
17. D. Das, S. Bandyopadhyay, “Sentence to document level emotion tagging – a coarse-grained study on Bengali blogs,” 2nd Mexican Conference on Pattern Recognition: Advances in pattern recognition (MCPR’10), pp.332–341,2010.

Partha De received B.E. (Computer Science & Engineering) from Jalpaiguri Government Engineering College (University of North Bengal) in 2004 and M.E. (Computer Science & Engineering) from Jadavpur University in 2015. He works as Astt. Prof. in Birbhum Institute of Engineering & Technology.