**SENTIMENTAL ANALYSIS ON IMDB REVIEW DATA**

**Phase – 3**

**Contents:**

Introduction

Models used

Future work

**Introduction:**

In phase-3 of Machine learning project we will be doing the model building of the models which we have selected before in the staring of the project, we need to build the model with the training data such that it can predict the test data accurately, the more the data is preprocessed the better accuracy we get as we have seen in the previous phase, in this phase we are going to use Bag of words(BOW) and Term frequency and inverse document frequency(TF-IDF), RNN(Recurrent neural network), LSTM(Long short term memory).

**Models used:**

Here we will be using Bag of words (BOW), Term frequency and inverse document frequency (TF-IDF), RNN(Recurrent neural network), LSTM(Long short term memory) and will be checking the accuracy of the models to check the models form best to the least effective models.

**Bag of words (BOW):**

We convert the sentences to lowercase and remove the stop words which are the common words that doesn’t have that importance then we take all the keywords which comes from the sentence then we will plot the histogram to know the frequency of particular word and we assign weight as 1 if it is occurred once, 2 if it is occurred twice, etc.

In bag of words, we can take the input as unigrams or bigrams, unigrams mean we take one word at a time while bigram means we will be taking two words at a time and do the entire process which was mentioned above.

Example:

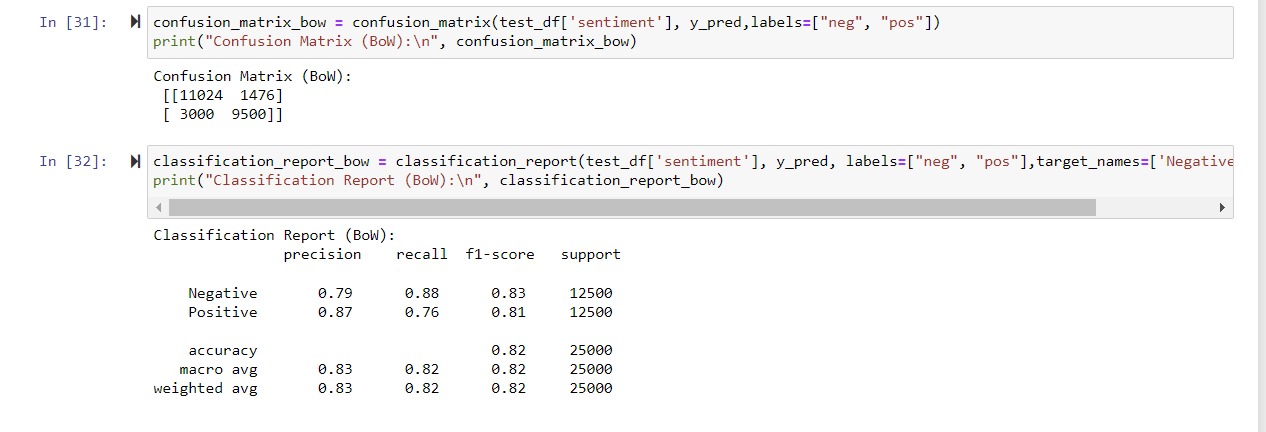
|  |  |  |
| --- | --- | --- |
| Sentence |  | Keywords |
| 1 | He is a good boy | Good, Boy |
| 2 | She is a good girl | Good, Girl |
| 3 | Boy & Girl are good | Boy, Girl, Good |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Sentence | good | boy | girl | o/p |
| 1 | 1 | 1 | 0 |  |
| 2 | 1 | 0 | 1 |  |
| 3 | 1 | 1 | 1 |  |

As shown in the above table the output is decided on the keywords associated with the sentence

**Output:**

It can be seen the accuracy for BOW vectorizer after doing the data preprocessing is 82%.

****

**TF-IDF:**

TF-IDF is also known as Term frequency and inverse document frequency, which is next stage of the Natural language processing which overcome the limitations of bag of words

Her we have two terms term frequency and inverse document frequency

Term frequency =

Term frequency:

|  |  |  |  |
| --- | --- | --- | --- |
| Sentence | good | boy | girl |
| 1 |  |  | 0 |
| 2 |  | 0 |  |
| 3 |  |  |  |

Inverse document frequency = log ()

Inverse document frequency:

|  |  |
| --- | --- |
| words | IDF |
| Good | log () = 0 |
| Boy | log () |
| Girl | log () |

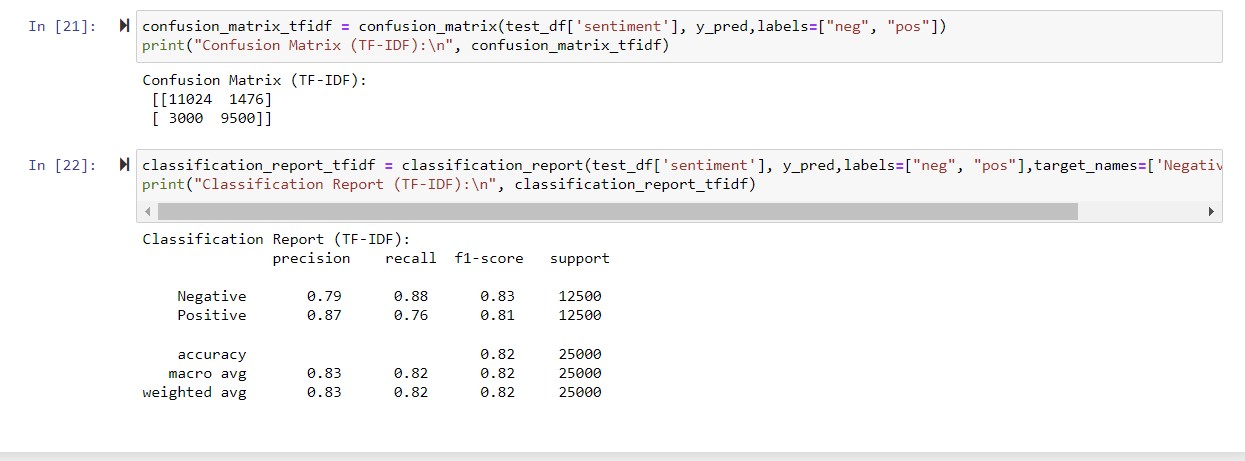
TF\*IDF

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | F1 | F2 | F3 | o/p |
| Sentence | Good | Boy | Girl |  |
| 1 |  |  | 0 |  |
| 2 |  | 0 |  |  |
| 3 |  |  |  |  |

So, by the above table the drawback which is present in the Bag of words is overcomed by giving the importance to the words to rare words and removes the common words.

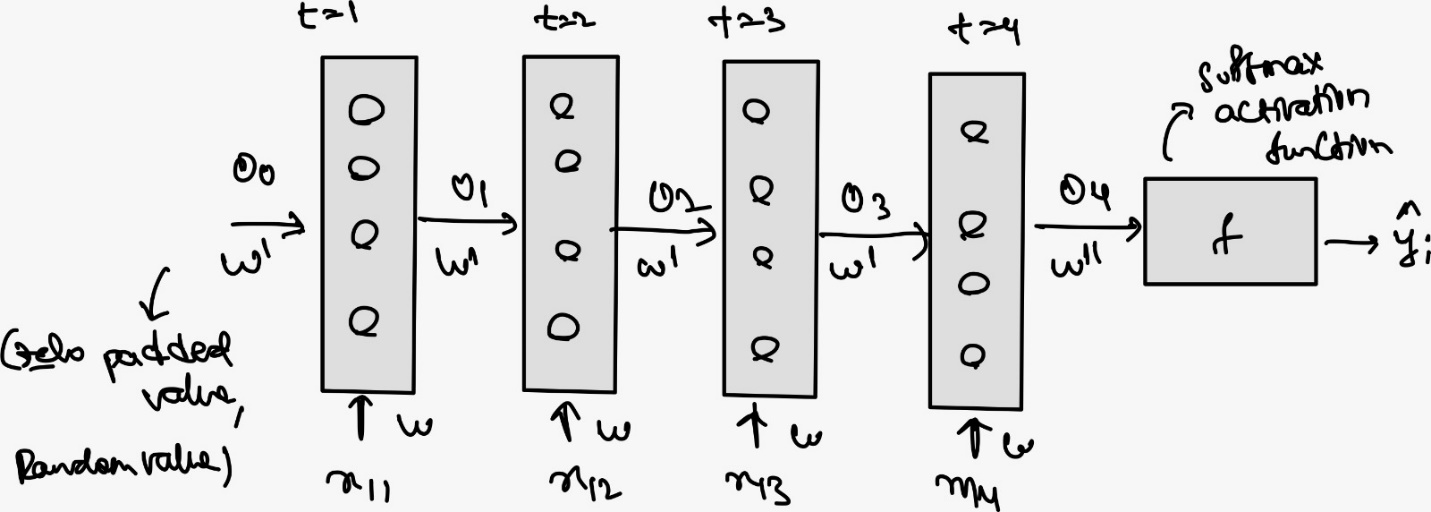
**Output:**

It can be seen that the accuracy for the TF-IDF vectorizer after the data preprocessing is 82%.

****

**RNN (Recurrent neural network):**

Recurrent neural network is used to obtain the output of the sentence in total where it considers the semantic of the sentence which is different from the before vectorizers, If there are 4 words in the sentence it does the calculation at every time instance with hidden layers in the node of the respective position and we will be using the SoftMax function to get the output.



We can observe that there are three nodes as there are 4 words in the sentence and this can increase or decrease respective to the sentence, we can observe that there are weights that are associated to the words that is ‘w’ and is calculated as , there will be an activation function in the hidden layers which gives us the output such as

As we move forward with the propagation the weights are transferred to the next node with the output of the previous node, the output of the final layer will be

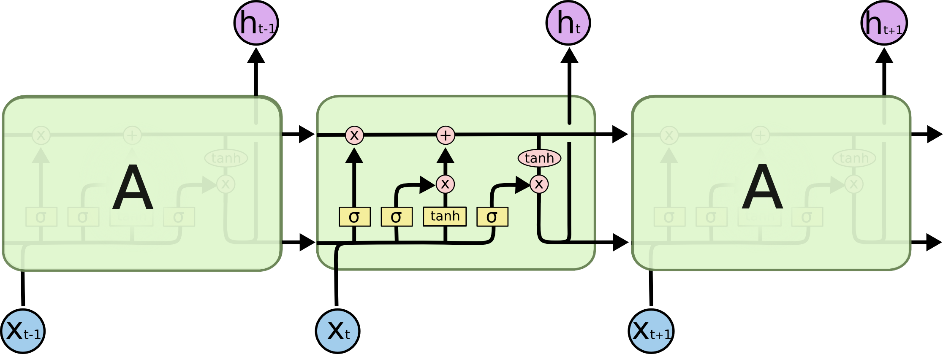
The output furcation will have an activation function called SoftMax which gives the probability that is the range of (0,1)

And the weights are updated with the help of the back propagation which is similar to gradient descent

**Drawbacks:** It has two drawbacks they are vanishing problem occurred by SoftMax activation function and extrapolation problem occurred by Relu activation function.

**LSTM (Long short-term memory):**

In RNN we need to store a lot of values, sometimes it becomes challenging so we go for LSTM which overcomes the drawbacks that have encountered in RNN



Here LSTM consist of memory cell, forget gate, input gate, output gate where if there is similarity in the words then the output will be the same as the input from the previous node and if there is dissimilarity then the output changes according to the present node with the help of the above four parts mentioned above

**Future work:**

Here in this phase we have trained the model with both Bag of words and TF-IDF vectorizers, now we are going through RNN model and LSTM to test the same for the accuracy and will be comparing the outputs by scale such that we get to know how good the models are performing better than the others.