

# Sentiment Analysis using LSTM

Ashok Tholusuri, Manish Anumala, Bhagyaraj Malapolu, G. Jaya Lakshmi

**Abstract:**-Extracting the sentiment of the text using machine learning techniques like LSTM is our area of concern. Classifying the movie reviews using LSTM is our problem statement. The reviews dataset is taken from the IMDB movie review dataset. Here we will classify a review based on the memory in the neural network of a LSTM cell state. Movie reviews often contain sensible content which describe the movie. We can manually decide whether a movie is good or bad by going through these reviews. Using machine learning approach we are classifying the movie reviews such that we can say that a movie is good or bad. LSTM is effective than many other techniques like RNN and CNN.

**Keywords**— Sentiment analysis, LSTM, Machine Learning, Natural Language Processing, IMDB reviews, Text Analytics.

## I. INTRODUCTION

In recent years data sciences has got a new prominent branch that deals with investigating the methods of recognizing favorable and unfavorable sentiments from the natural language i.e. textual data. This has applications in various fields like business intelligence, opinion trends, message filtering and customer feedback etc. One of these applications is our working area. Classifying the sentiment of movie review based on the text in the review. There has been a huge development in the fields of natural language processing [1] using methods from deep neural networks for various tasks including simpler ones like POS tagging and Named Entity Recognition, removing stopwords and stemming etc. Sentiment analysis [2] is the fundamental problem of giving a machine to understand the emotions and opinion expressed within the text data.

In its early phase sentiment classification is performed using methods like Naive Bayes, Support Vector Machines etc. Recently deep learning approaches like using neural networks (CNN, RNN, ANN etc.) have gained popularity by showing impressive results. Traditional methods when compared to deep learning [3] methods have its limitations in analyzing the raw data or in its natural form. Whereas in deep learning it learns the representation of data at multiple layers by combining the simple non-linear functions which transforms the raw data into higher abstract level representation.

The present paper is the work on the movie review classification based on the sentiment analysis using the

machine learning approach. There are many ways that we can classify the review based on the emotion yet we are using the most advanced technique which is LSTM networks. With this we are performing the task of sentiment analysis on movie reviews and we are very accurate results.

## II. LITERATURE SURVEY

Aditya Timmaraju and Vikesh Kumar [4] have proposed an intelligent model for classifying movie reviews using Recursive RNN. This is a system which deals with sentiment classification on the natural language i.e. text. This system is an extension for sentence-level sentiment classification. Jan Deriu and Mark Ceilieba [5] have implemented a system that classifies the sentiment of twitter messages. This work builds upon the deep learning approach. Here they used 2-layer convolution neural networks. Here the entire task is divided into 3 sub tasks. Sachin Kumar, Anand Kumar [6] jointly posed a new system for the same task but with different approach i.e. using CNN. In this paper Sentiment analysis task in NLP is performed as the most text contain information in the form of opinions and emotions. This model will give a comprehensive view about an opinion or sentiment which can be categorized into negative, positive or neutral. Fenna Miedema [7] proposed an advanced system that is used to classify the tweets which are in text format. He employed the LSTM networks to design this model which makes this most efficient one out of all existing models. Tony Mullen and Nigel Collier [8] proposed a sentiment classification system which makes use of support vector machines. This model operates on varied information sources. In past researches it is clear that this model is successful with diverse information sources but it lacks a bit of accuracy in the case of long sentences. Many proposed models are not so successful in correctly classifying the long text data. On the other hand models incorporating LSTM networks are showing impressive results which is because its capability of dealing the long text data.

## III. METHODOLOGY

The dataset is the IMDB movie review dataset [10] which is available by default in the python module *keras.dataset*. The dataset has a total of fifty thousand reviews which are text files, the maximum length for each review is 500 words. The entire dataset is divided into two sets in the process for training and testing.

To learn the contextual relations between words in the training data, we use word embedding model. This also helps in the better representation of limited content in short texts. We have used one of the layers of keras module which

**Revised Manuscript Received on 14 August, 2019.**

**Ashok Tholusuri**, Student Scholar, IT Department, V.R.Siddhartha Engineering College, Vijayawada, Andhra Pradesh, India (Email: ashok.srinu97@gmail.com)

**Manish Anumala**, Student Scholar, IT Department, V.R.Siddhartha Engineering College, Vijayawada, Andhra Pradesh, India (Email: manishanumala@gmail.com)

**Bhagyaraj Malapolu**, Student Scholar, IT Department, V.R.Siddhartha Engineering College, Vijayawada, Andhra Pradesh, India (Email: bhagyaraj455@gmail.com)

**G. Jaya Lakshmi**, Assistant Professor, IT Department, V.R.Siddhartha Engineering College, Vijayawada, Andhra Pradesh, India (Email: jaya1123@gmail.com)

is called embedding layer for this purpose. The word embedding is done with the help of this layer which helps in reducing the high dimensionality of text.

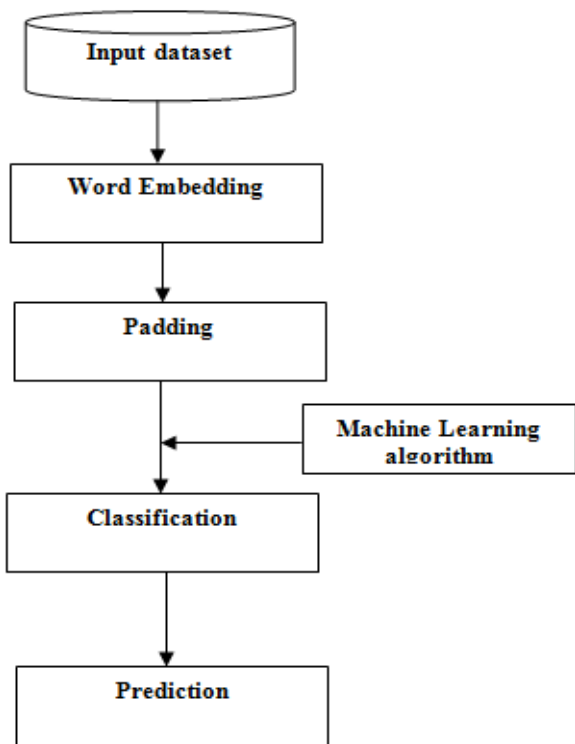


Fig. 1. Architectural diagram of classification process

Padding is done to each review to make sure that every review has the same size so that it is easy to process the data. Padding is nothing adding some extra bits to the data to make it to particular size. Later the data is trained with one of the machine learning algorithms. The built Machine learning models display the confusion matrix and the accuracy of the models. The processing time for running this module is also displayed.

In this stage a model for classifying the reviews based on the sentiment is generated using the LSTM algorithm. The model is trained with the training dataset and then is tested in the coming phases. The model for sentiment classification is completely built in this stage.

The trained sentiment classification model is tested with the test dataset and the accuracy is noted. The performance of the model is based on the accuracy of the model i.e. the total correct classifications of the reviews. The results are made more understandable by using the confusion matrix [11] which all the classifications including the correct and wrong classifications.

IV. EXPERIMENTAL RESULTS

In this research work python is used to find how well the algorithm works on the reviews based on the accuracy metric. The first phase is learning and the second is predicting. In learning phase the model is trained with the dataset and it classifies the train dataset with that perception. The results are recorded and the performance depends on the learning phase. To avoid underfitting we should use a dataset of good size i.e. learning phase should be good complete. The model learns how to classify based on the

training and the model is tested with the test set in testing phase.

```

    Using TensorFlow backend.

    Importing packages done.
    Completed generation of test,train datasets.
    Started Creating Model.
    Model Successfully compiled.
    Starting Training....
    Epoch 1/2
    25000/25000 [=====] - 519s 21ms/step - loss: 0.6265 - acc: 0.6604
    Epoch 2/2
    25000/25000 [=====] - 609s 24ms/step - loss: 0.3240 - acc: 0.8641
    
```

Fig. 2. Learning Results

The Fig 2 shows the results of the learning phase. We can see that we are using tensorflow in the backend which helps in importing all the necessary packages. The first step done is splitting the dataset into train set and test set. After completing this an LSTM model for sentiment classification is generated and compiled. Next the training phase begins and the model learns how to classify the reviews with experience by the end of the training phase. We can see how the performance progresses with the experience in the epochs. The performance increases epoch by epoch indicating that the model learns on experience. We can see the accuracy and loss of the learning phase.

Evaluation of Test dataset started

Accuracy: 86.85%

Test score: 30.60%

0.86848

	precision	recall	f1-score	support
pos	0.93	0.80	0.86	12500
neg	0.82	0.94	0.88	12500
micro avg	0.87	0.87	0.87	25000
macro avg	0.88	0.87	0.87	25000
weighted avg	0.88	0.87	0.87	25000

Fig. 3. Prediction Results

The results of the prediction phase are shown in the fig 3. Here the prediction phase involves the testing and results of the model are displayed where the accuracy of the test is given as shown above. We can see the accuracy of the LSTM model and the correct classification and wrong classifications with the help of positive, negative class labels and their respective precision and support.

Here, the test set is classified using the trained LSTM model based on the sentiment and thus the accuracy of the model with respect to the test dataset is noted. The total number of classifications and the correct, wrong classifications are represented with the help of confusion



matrix. In the above figure we can see the f1-score for both positive and negative reviews.

The classification results can be clearly understood and represented in the form of confusion matrix as shown in the figure 4. It has the count of all the classifications represented as a matrix.

In the confusion matrix we can see the incorrectly classified positive, negative reviews and correctly classified positive, negative reviews. The numbers in the principle diagonal are the correctly classified negatives and positives respectively.

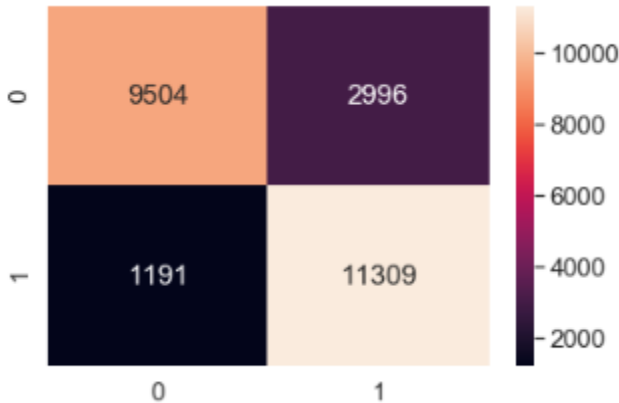


Fig. 4. Confusion matrix

The Fig 3 depicts that the accuracy of the model is 86.85% by using the Long Short-Term Memory networks algorithm. The accuracy of this model is greater than all other machine learning algorithms when we use this dataset. LSTM is very good in classifying the sequence text data.

#### V.CONCLUSION AND FUTURE WORK

Sentence level sentiment mining acclimated remove the remarked regular alternatives and assessment words from the input dataset. The model for understanding the emotion of the movie review from the text of the review is build using LSTM which classifies the reviews very well. This strategy will be useful to improve the movies by getting to know which content has good reception by the audience.

Several sentiment mining and classification systems are audited and referred to. Our results showed that the Long Short Term Memory Networks algorithmic standard beat others as far as exactness.

This paper proposes a sentiment classification model which helps in classifying movie reviews based on the emotion of the sequence text data. This system is different and efficient from others as it catches up in classifying long sequence data with the help of LSTM which makes use of long-term memory and thus can handle the long term dependencies very effectively. By using larger datasets we can still improve the performance of the model.

In the future examination, we will in general complete classifying movies not just reviews by using further analysis on the internal results of each review corresponding to a moive. We tend to additionally expect to improve the execution of both movie classification technique and movie review classification models. By integrating these two models into one we can create a model with which we can classify a collection of movies by classifying the reviews

and each movie and further analyzing the resuts we can classify movies. Thus by passing a set of movies we can classify them all at a time.

#### REFERENCES

1. Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., and Potts, C. (2011, June). Learning word vectors for sentiment analysis. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1 (pp. 142-150). Association for Computational Linguistics
2. ApoorvAgarwal, BoyiXie, Ilia Vovsha, Owen Rambow, Rebecca Passonneau. Sentiment Analysis on twitter data. Department of Computer Science, Columbia University, New York, NY 10027 USA. {apoorv@cs, xie@cs, iv2121@, rambow@ccls, becky@cs}.columbia.edu
3. PreranaSinghal, Pushpak Bhattacharya. Sentiment Analysis and Deep Learning: A survey. Dept. of Computer Science and Engineering , Indian Institute of Technology, Powai Mumbai, India.
4. AdityaTimmaraju, VikeshKhanna.Sentiment Analysis on Movie Reviews using Recursive and Recurrent Neural Network Architectures. Department of Electrical Engineering, Stanford University, Stanford, CA - 94305,adityast@stanford.edu.
5. Jan Deriu, Mark Cieliebak. Sentiment Analysis using Convolutional Neural Networks with Multi-Task Training and Distant Supervision on Italian Tweets. Zurich University of Applied Sciences, Switzerland. deri@zhaw.ch, ciel@zhaw.ch.
6. M. Anand Kumar, S. Sachinkumar. Sentiment Analysis of Tweets in Malayalam Using Long Short-Term Memory Units and Convolutional Neural Nets. Mining Intelligence and Knowledge Exploration: 5th International Conference, MIKE 2017, Hyderabad, India, December 13–15, 2017.
7. FennaMiedema, Prof. dr. SandjaiBhulai. Sentiment Analysis with Long Short-Term Memory networks, VrijeUniversiteitAmsterdam,August 1, 2018.
8. Tony Mullen and Nigel Collier. Sentiment analysis using support vector machines with diverse information sources,National Institute of Informatics (NII),Hitotsubashi 2-1-2, Chiyoda-ku,Tokyo,101-8430,Japan.
9. PRANJAL SRIVASTAVA, Essentials of Deep Learning: Introduction to Long Short Term Memory, DECEMBER 10, 2017.
10. <http://ai.stanford.edu/~amaas/data/sentiment/>
11. <https://towardsdatascience.com/understanding-confusion-matrix-a9ad42dcfd62>
12. Machine Learning Tom M. Mitchell McGraw-Hill