SENTIMENT ANALYSIS FOR MOVIE REVIEWS BY USING CNN

Submitted for the Summer Internship

on

Machine Learning and Deep Learning

(from 8th June, 2021 to 31st July, 2021)

Organised by
DST Centre of Excellence – Artificial Intelligence, IGDTUW
IGDTUW-Anveshan Foundation
Department of AI and Data Sciences, IGDTUW

By

Arushi Garg
BTech, ECE-1st year
Indira Gandhi Delhi
Technical University
For Women

Soumya Vats
BTech, ECE-1st year
Indira Gandhi Delhi
Technical University
For Women

INDIRA GANDHI DELHI TECHNICAL UNIVERSITY FOR WOMEN

(Established by Govt. of Delhi vide Act 09 of 2012)
Kashmere Gate, Delhi-110006

PROJECT REPORT

1. Introduction-

In today's world with ever growing access to the internet and its services , it has become easier for users to express their opinion and reviews about anything and everything. Movies are a widely appreciated art form, with movie reviews by critics as well as regular people holding weightage in forming decisions about the same. In our project we abstract the polarity of the sentiment expressed also known as sentiment analysis in the movie reviews using machine learning and deep learning. In simple terms sentiment analysis is a task of preprocessing the given textual data and extracting the emotion from it also known as opinion mining [1]. Sentiment analysis is a well-known part of Natural Language processing [2]. Traditional text classification methods were dictionary-based and basic machine learning methods but recently they have been replaced by more efficient and accurate deep learning methods such as LSTM (Long Short Temporary Memory) and CNN(Convolutional Neural network) [3].

We have used IMDB dataset with 50,000 reviews where 25,000 are marked as positive and the other 25,000 are marked as negative. We have also used traditional machine learning methods like logistic regression and compared their accuracy with neural network methods.We have used a hybrid model of CNN and LSTM to achieve high accuracy.we have used a 3 layer model of convolutional neural network and later applied LSTM to increase its accuracy. Word2vec is used for word embedding and n-gram analysis is done on the dataset to gain better results.

2. Literature Review-

Sentiment analysis is a widely researched field and with regular advancement in deep learning it is getting easier to analyze the polarity of the sentiment expressed. In [4] the author used CNN and lexical resources to gain an accuracy of 87.9% and concluded that SCNN improves sentiment classification by leveraging word semantic embedding and sentiment embedding. Naive Bayes is a straightforward, yet powerful and regularly utilized, machine learning classifier. In [5] the author used n-gram analysis and NBSVM to achieve an accuracy of 93.05% and hence concluded that when this model is combined with RNN-LSTM, it gives the best result among all the ensemble models. [6] concluded that the Genetic Algorithm performs better than NB and Hybrid classifiers are more accurate than single classifiers. The author in [7] proved that the highest accuracy was achieved by Naive Bayes as compared to KNN and Random forest algorithm.

The k-nearest neighbors (KNN) supervised machine learning algorithm can be used to solve both classification and regression problems. It's easy to implement and understand and is one of the favorites for sentiment analysis. [1] works on the algorithms of Information Gain and KNN. These algorithms enabled them to achieve an accuracy of 96.80%. The author in [8] did the same.

A support vector machine (SVM) is a supervised machine learning model that uses classification algorithms for two-group classification problems. After giving an SVM model set of labeled training data for each category, they're able to categorize new text.Comparative studies have been shown in [1] achieving an accuracy of 82.89% using SVM with linear kernels. A composite model [9] has been proposed which comprises a Probabilistic Neural Network (PNN) and a two-layered Restricted Boltzmann (RBM). [10] showed that the use of feature selection methods, namely IG, can improve the accuracy of the SVM classifiers. Movie review data can be classified into positive reviews and negative reviews.

Authors of [11] have shown that CNN has outperformed LSTM and CNN-LSTM because the aim of the project was to find the polarity of the sentiment and not to rate the intensity of the polarity. So it can be said that LSTM performs well in NLP tasks where the syntactic and semantic structure are both important.

in [12] it is experimentally demonstrated that neural networks work more efficiently than random forest and SVM because they can extract robust features using methods of vectorization. The author of [13] shows that neural network methods not only help in sentiment analysis of textual data but also sentiment analysis of visual data and CNN helps by forming an internal relation between text and image and gives a better result in sentiment analysis.

3. Issues/Challenges with the Existing Work

In this project we have tried to utilise both deep learning and machine learning techniques to achieve high accuracy. One of the major issues in existing work is the usage of incomplete dataset for the model, in our project we have tried to use the biggest dataset available to us to increase the accuracy of the model. A major amount of work has to be done in cleaning and preprocessing of the dataset to achieve better results. In most of the already existing work , only 1-2 models have been used to determine the results whereas in our model is a comparative study between traditional machine learning techniques and deep learning methods, we have experimentally shown the difference in accuracy achieved by both the models.

4. Objectives of the Proposed Work-

The objective of our project “ Sentiment Analysis of Movie Reviews” is to analyse and categorise the sentiment expressed by the viewers on IMDB website into positive and negative. It is humanly impossible and illogical to go through thousands of reviews present on IMDB to conclude the success or failure of the movie. So In this work we propose a sequential model to identify the sentiment analysis of the movie reviews.CNN has a convolutional layer to extract information by a larger piece of text, so we work for sentiment analysis with the convolutional neural network, and we design a simple convolutional neural network model and test it on the benchmark, the result shows that it achieves better accuracy performance on movie review sentiment classification than some of the traditional methods such as the SVM and Naive Bayes methods.

We also want to analyse if the system is capable of analysing the valid result polarity and evaluation.

5. Methodology

5.1. Data Set Used- The IMDB Dataset.csv contains 50000 records with two variables review and sentiment. Out of the total records, 25000 are of positive polarity while the other 25000 are of negative polarity (Table 1). The review field contains noise data which needs to be cleaned.

Reviews	No. of Positive sentiments	No.of Negative sentiments
50000	25000	25000

Table 1 Dataset

5.2. Pre-processing-

Data cleaning-

In machine learning tasks, cleaning or pre-processing the data is as important as model building if not more. And when it comes to unstructured data like text, this process is of most importance. IMDB reviews are posted by users manually, so we observe high usage of contractions and chat words in it. Also, some reviews are collected from other sites, so we also observe usage of many HTML tags in the dataset.

Punctuations are removed and the contractions are replaced with words.

Stopwords are imported from the nltk.corpus and hence are removed. These words do not add in analysis the polarity of the review and are only required to form meaningful sentences. The whole text is converted into lower case and additional urls and other html links are removed.

Stemming and Lemmatization-

Stemming just removes or stems the last few characters of a word, often leading to incorrect meanings and spelling. Lemmatization considers the context and converts the word to its meaningful base form, which is called Lemma.

Tokenization-

Tokenization is the process of replacing sensitive data with unique identification symbols that retain all the essential information about the data.

Vectorisation-

Vectorization is a technique to implement arrays without the use of loops. Using a function instead can help in minimizing the running time and execution time of code efficiently.

5.3. Implementation

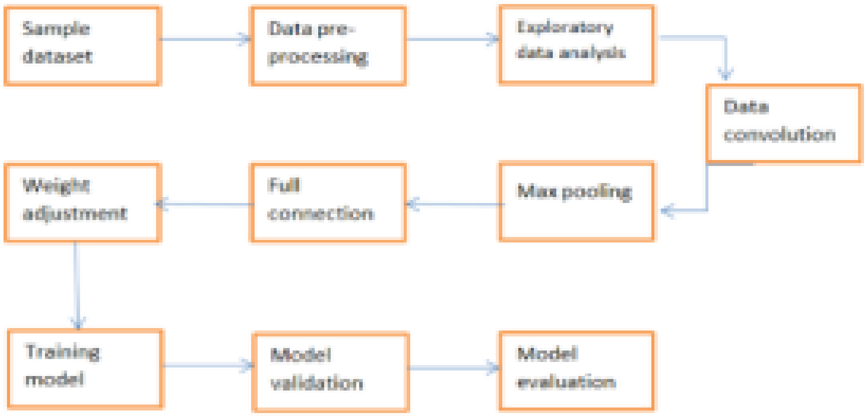


Figure 1

After data cleaning, the processed data is analysed in tabular form.

Word clouds are generated to know the intensity and frequency of the words used in positive and negative reviews separately as shown in figure 2.

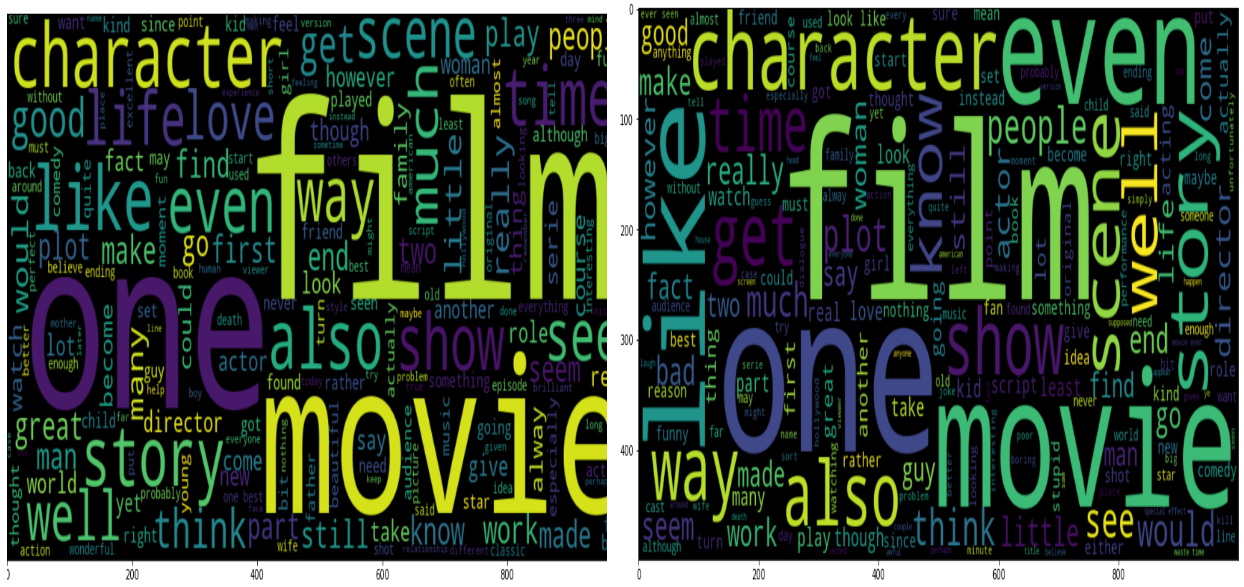


Fig 2.a Positive review **Fig 2.b Negative Reviews**

Figure 2

From these word clouds, we are not able to judge any starling differences in both the sentiments by looking at words. We don’t see usage of extreme negative connotation or abusive language used while writing negative reviews.

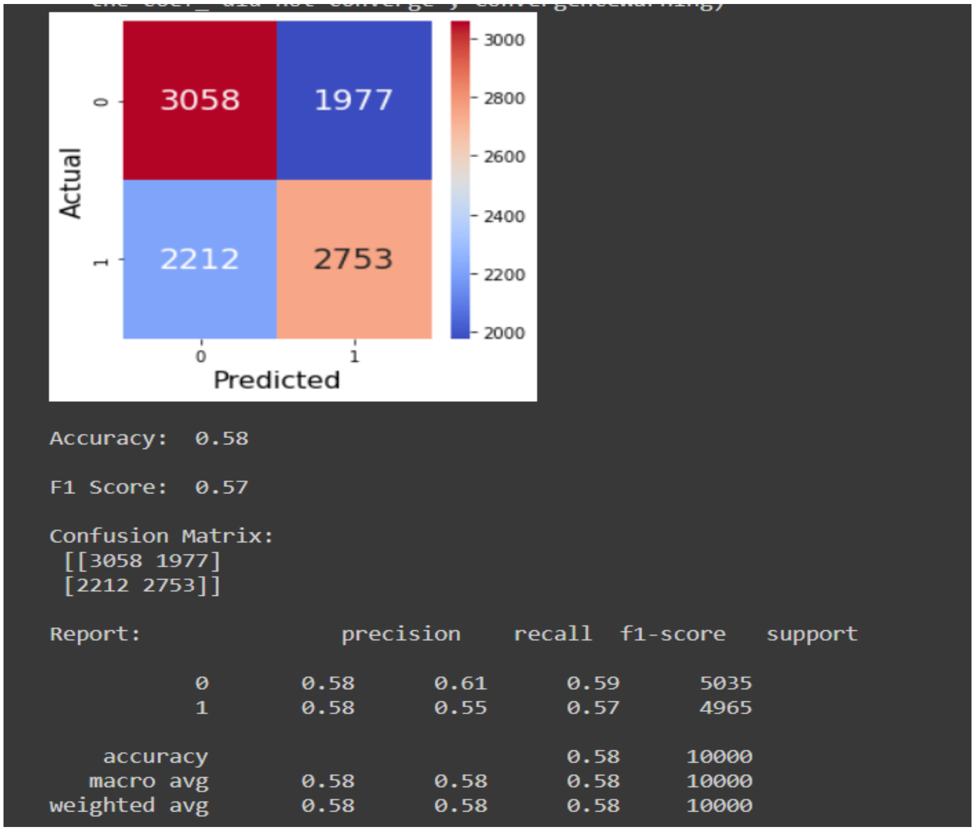
The dataset is divided into X and Y variables , where
X- contains labels that help to predict the sentiment
Y- predicted sentiment as shown in Figure 3.

```
X=data[['count_sent', 'count_word', 'count_unique_word',
       'count_letters', 'count_punctuations', 'count_words_upper',
       'count_words_title', 'count_stopwords', 'mean_word_len',
       'word_unique_percent', 'punct_percent']]
y=data['sentiment']
```

Figure 3

Then, the dataset is splitted into testing and training datasets. 80% of the dataset is used for training whereas the remaining 20% is used for testing.

Initially, the logistic regression model is applied on the dataset. As a result, we achieved an accuracy of 58% which is represented along with the confusion matrix as shown in Figure 4.



```
Accuracy:  0.58

F1 Score:  0.57

Confusion Matrix:
 [[3058 1977]
 [2212 2753]]

Report:               precision    recall  f1-score   support

           0       0.58      0.61      0.59      5035
           1       0.58      0.55      0.57      4965

    accuracy                           0.58     10000
   macro avg       0.58      0.58      0.58     10000
weighted avg       0.58      0.58      0.58     10000
```

Figure 4

Then, in order to improve the accuracy, textblob is imported to perform sentiment analysis. TextBlob is a python library for Natural Language Processing (NLP). TextBlob actively used Natural Language ToolKit (NLTK) to achieve tasks related to sentiment analysis. NLTK is a library which gives easy access to a lot of lexical resources and allows users to work with categorization, classification and many other tasks. TextBlob is a simple library which supports complex analysis and operations on textual data. For lexicon-based approaches, a sentiment is defined by its semantic orientation and the intensity of each word in the sentence. This requires a pre-defined dictionary classifying negative and positive words. Generally, a text message will be represented by a bag of words. After assigning individual scores to all the words, final sentiment is calculated by some pooling operation like taking an average of all the sentiments.

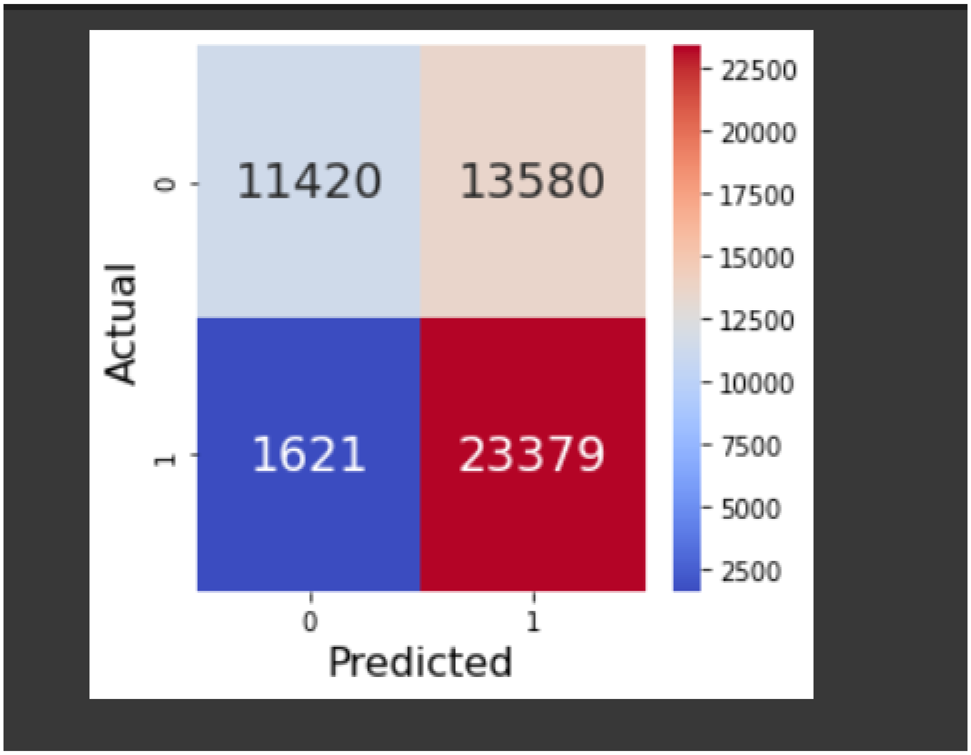


Figure 5

Hence, the accuracy increased to 69% which is also represented with a confusion matrix as shown in Figure 5.

N-gram analysis model- It is used to predict the probability of a given n-gram within any sequence of words in the language. Unigram, bigram and trigram analysis is done for the given dataset as shown in Figure 6.

The order that words are used in text is not random. In English, for example, you can say "the red apple" but not "apple red the ". The general idea is that you can look at each pair (or double, triple etc.) of words that occur next to each other. In a sufficiently-large corpus, you're likely to see "the red" and "red apple" several times, but less likely to see "apple red" and "red the". This is useful to know if, for example, you're trying to figure out what someone is more likely to say to help decide between possible output for an automatic speech recognition system. These co-occurring words are known as 'n-grams', where "n" is a number saying how long a string of words you consider.

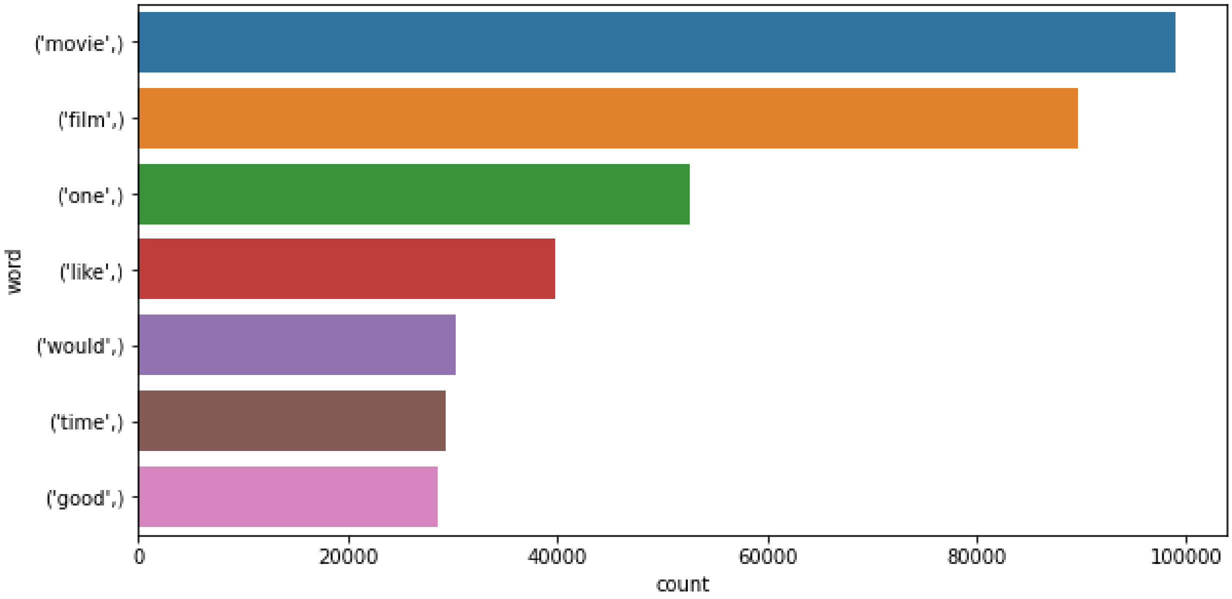


Figure 6.a Uni-gram analysis

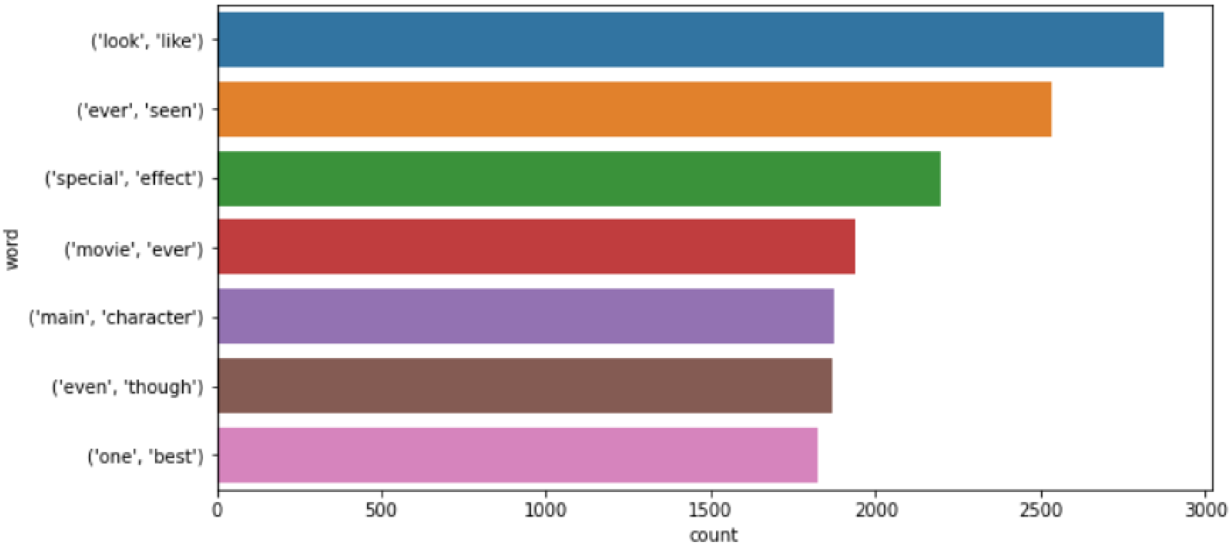


Figure 6.b Bi-gram analysis

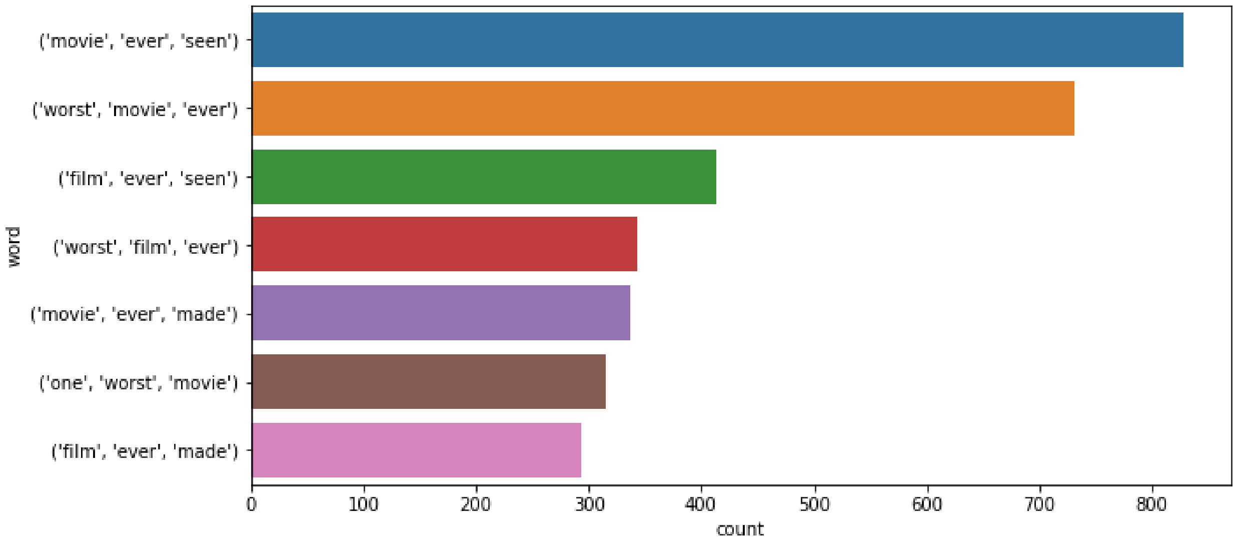


Figure 6.c Trigram Analysis

Word embedding-

Many Machine Learning algorithms and almost all Deep Learning Architectures are incapable of processing strings or plain text in their raw form. They require numbers as inputs to perform any sort of job, be it classification, regression etc. in broad terms. A Word Embedding format generally tries to map a word using a dictionary to a vector.

Frequency based Vectorization- It is used to map words to a vector using a dictionary. The weights are assigned to the vectors depending upon their importance in deterring the sentiment of the review.

Count vectorization-It is used to count the number of words in that particular text sample.

5.4.The model

The model of CNN has a multi-layered feed-forward neural network which is made by loading many hidden layers on top of each other in a sequence.This design enables CNN to observe and learn hierarchical features. The network structure of CNN has three layers: a pooling layer, convolutional layer and a fully connected layer.

Data Convolution-

Data convolution is applied to extract the features variables. 1D convolution layers are applied to help in learning patterns at a specific position in a sentence that is further utilized to recognize the patterns at different positions.

Max Pooling-

In this step, a mask is applied to enhance and reduce the results of the convolution layer. Max-Pooling is applied on each patch of feature map to extract the maximum value and ignore the rest. This helps in reducing the inputs to the next layer. The max-pooling layer allowed to reduce the words by 50 % for the final output. Number of layers used is 3. Matrix is divided into sub-matrices of order 2.

For **model fitting-** 10 epochs are being used with a batch size of 256.

LSTM(long short term memory)-

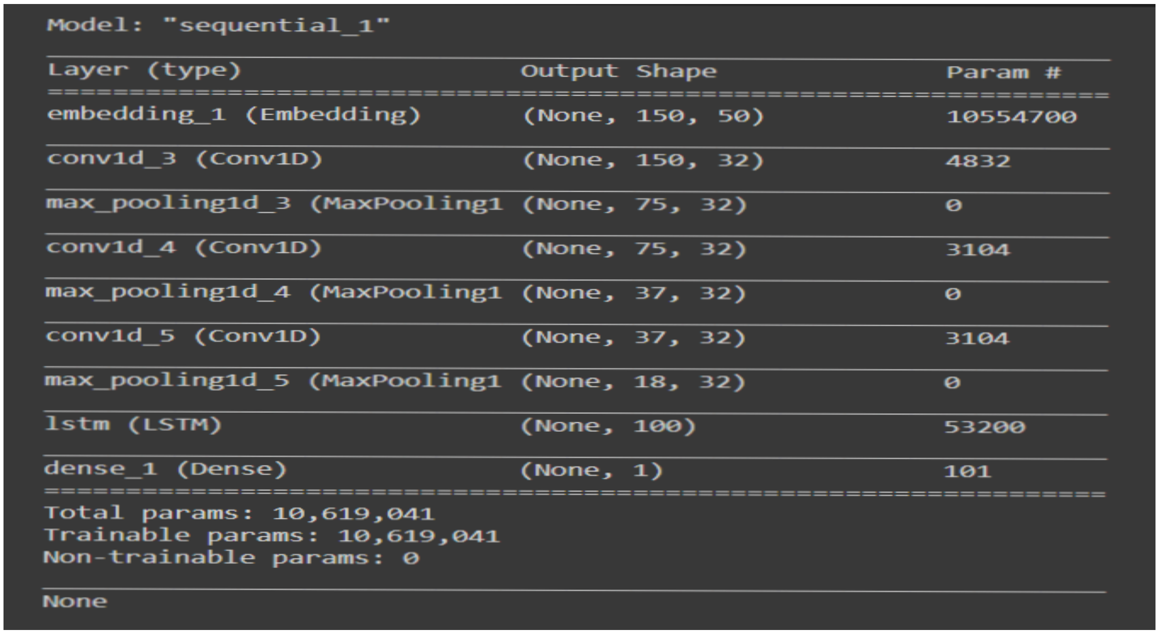
They are considered as artificial RNN. It consists of cells, input, output and forget gates. Data is read or written on the cells as we write or read from the computer memory. The cells make decisions about read or write operations via input & output gates. In the proposed CNN LSTM model, LSTM layer was applied on the output resulting from the max-pooling layer.

5.5. Result and observation-

Following models were fitted on the train data: 1. Convolutional Neural Network (CNN) 2. Long Short-Term Memory (LSTM).

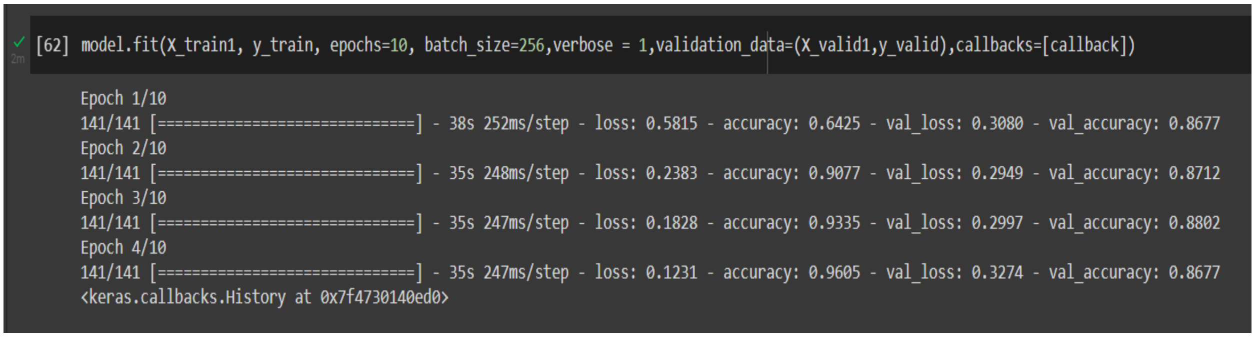
Apart from these two, basic models such as logistic regression is also applied and n-gram analysis is also used. After applying CNN, we achieved an accuracy of 64.4%.

On applying CNN+LSTM model, the accuracy increased to 87.23%. The following Figure 7 shows the model summary on training data. Figure 8 shows the model fitting on the dataset.



```
Model: "sequential_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
embedding_1 (Embedding)      (None, 150, 50)           10554700
_________________________________________________________________
conv1d_3 (Conv1D)            (None, 150, 32)           4832
_________________________________________________________________
max_pooling1d_3 (MaxPooling1 (None, 75, 32)            0
_________________________________________________________________
conv1d_4 (Conv1D)            (None, 75, 32)            3104
_________________________________________________________________
max_pooling1d_4 (MaxPooling1 (None, 37, 32)            0
_________________________________________________________________
conv1d_5 (Conv1D)            (None, 37, 32)            3104
_________________________________________________________________
max_pooling1d_5 (MaxPooling1 (None, 18, 32)            0
_________________________________________________________________
lstm (LSTM)                  (None, 100)               53200
_________________________________________________________________
dense_1 (Dense)              (None, 1)                 101
=================================================================
Total params: 10,619,041
Trainable params: 10,619,041
Non-trainable params: 0
_________________________________________________________________
None
```

Figure 7 Model Summary



```
[62] model.fit(X_train1, y_train, epochs=10, batch_size=256,verbose = 1,validation_data=(X_valid1,y_valid),callbacks=[callback])

     Epoch 1/10
     141/141 [==============================] - 38s 252ms/step - loss: 0.5815 - accuracy: 0.6425 - val_loss: 0.3080 - val_accuracy: 0.8677
     Epoch 2/10
     141/141 [==============================] - 35s 248ms/step - loss: 0.2383 - accuracy: 0.9077 - val_loss: 0.2949 - val_accuracy: 0.8712
     Epoch 3/10
     141/141 [==============================] - 35s 247ms/step - loss: 0.1828 - accuracy: 0.9335 - val_loss: 0.2997 - val_accuracy: 0.8802
     Epoch 4/10
     141/141 [==============================] - 35s 247ms/step - loss: 0.1231 - accuracy: 0.9605 - val_loss: 0.3274 - val_accuracy: 0.8677
     <keras.callbacks.History at 0x7f4730140ed0>
```

Figure 8 Model Fitting

6. Conclusion

To understand and extract valuable knowledge from the huge amount of data present on the internet, artificial intelligence can be used and with the advancement in deep learning and machine learning techniques we can furthermore improve the results. In this project we tried to perform sentiment analysis on movie reviews present on IMDB website to understand people's attitude and emotions towards the movie which has been reviewed. We concluded experimentally that models involving neural network methods like CNN and LSTM predicted the result with better accuracy than models involving traditional machine learning techniques like SVM, LSTM, Naive Bayes, KNN etc. Our proposed model which is a composite model of CNN and LSTM and gives an accuracy of 87.23% and hence cite can be said that the hybrid models used along with other machine learning techniques gives more prospective and promising results.

7. References

[1]. Bodapati, J. D., Veeranjaneyulu, N., & Shaik, S. (2019). Sentiment Analysis from Movie Reviews Using LSTMs. *Ingenierie des Systemes d'Information*, *24*(1).

[2]. Pouransari, H., & Ghili, S. (2014). Deep learning for sentiment analysis of movie reviews. *CS224N Proj*, 1-8.

[3]. . Jang, B., Kim, M., Harerimana, G., Kang, S. U., & Kim, J. W. (2020). Bi-LSTM model to increase accuracy in text classification: Combining Word2vec CNN and attention mechanism. *Applied Sciences*, *10*(17), 5841.

[4]. Yin, R., Li, P., & Wang, B. (2017, June). Sentiment lexical-augmented convolutional neural networks for sentiment analysis. In *2017 IEEE Second International Conference on Data Science in Cyberspace (DSC)* (pp. 630-635). IEEE.

[5]. Dhande, L. L., & Patnaik, G. K. (2014). Analyzing sentiment of movie review data using Naive Bayes neural classifier. *International Journal of Emerging Trends & Technology in Computer Science (IJETTCS)*, *3*(4), 313-320

[6]. Govindarajan, M. (2013). Sentiment analysis of movie reviews using hybrid method of naive bayes and genetic algorithm. *International Journal of Advanced Computer Research*, *3*(4), 139.

[7]. Baid, P., Gupta, A., & Chaplot, N. (2017). Sentiment analysis of movie reviews using machine learning techniques. *International Journal of Computer Applications*, *179*(7), 45-49.

[8].Samat, N. A., Salleh, M. N. M., & Ali, H. (2020, January). The comparison of pooling functions in convolutional neural network for sentiment analysis task. In *International Conference on Soft Computing and Data Mining* (pp. 202-210). Springer, Cham.

[9].Ain, Q. T., Ali, M., Riaz, A., Noureen, A., Kamran, M., Hayat, B., & Rehman, A. (2017). Sentiment analysis using deep learning techniques: a review. *Int J Adv Comput Sci Appl*, *8*(6), 424.

[10]. Maulana, R., Rahayuningsih, P. A., Irmayani, W., Saputra, D., & Jayanti, W. E. (2020, November). Improved Accuracy of Sentiment Analysis Movie Review Using Support Vector Machine Based Information Gain. In *Journal of Physics: Conference Series* (Vol. 1641, No. 1, p. 012060). IOP Publishing.

[11]. Haque, M. R., Lima, S. A., & Mishu, S. Z. (2019, December). Performance Analysis of Different Neural Networks for Sentiment Analysis on IMDb Movie Reviews. In *2019 3rd International Conference on Electrical, Computer & Telecommunication Engineering (ICECTE)* (pp. 161-164). IEEE.

[12]. Jnoub, N., Al Machot, F., & Klas, W. (2020). A domain-independent classification model for sentiment analysis using neural models. *Applied Sciences*, *10*(18), 6221.

[13]. Cai, G., & Xia, B. (2015). Convolutional neural networks for multimedia sentiment analysis. In *Natural language processing and Chinese computing* (pp. 159-167). Springer, Cham.