IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

MULTI-LABEL NEWS CLASSIFICATION USING BI-LSTM

[1*]Yash Kumar Goel, [2]S. D. Samantaray

[1]*Research Scholar, [2] Professor & Head

[12] Department of Computer Science & Engineering, College of Technology, G. B. Pant University of Agriculture and Technology, Pantnagar-263145,India

Abstract: Multi-Label Text Classification is used when there are two or more classes as well as the information to be classified may relate to neither of the classifications or all of them at the very same time. Text classification with multiple labels has many real-world applications, such as categorizing businesses on Yelp or classifying movies into one or more genres. The large number of messages that were published ultimately results in scattered messages and also a very associated with a wide variety homepage that was not categorized through categories like health, sports, technology, economics, tourism, and etc. The lack of classification makes it hard for a person to interpret or obtain data relevant to particular preferred classifications. The technique of text classification, which in the classification stage is capable of classifying instantly against several classifications on unstructured text with natural language, is one remedy that can be used. In this study, the classification procedure was carried out using the BI-LSTM technique, with feature set expansion to include a term in the concepts. Because a News Titles is a short text that could lead to ambiguity in classification class and the title of the news item could be linked to a number of different sources that could lead to ambiguity in classification class, the introduction of phrase seeks to optimize the classification method. The challenge of news classification begins with web scraping to gather real-time news Titles from news websites, which are then instantly classified using different classification methodologies and introduce the Wordnet and WordSense database for multi-label news titles classification. The acquired accuracy of (Bi-LSTM) was 97.91 percent, which exceeded the approximate accuracy of each individual plan. This technique could be very helpful for academic who want to investigate headlines in order to support their instruction.

Index Terms - **News Classification, Multi-label Classification, Data Mining, Bi-LSTM, WordNet, WordSense**.

I. INTRODUCTION

News Corpus, which is news that is spread as well as noted online, is one of the many internet usage in this day and age. Sports, politics, entertainment, as well as other topics are covered in Electronic news (e-news), just as they were in recent press kinds (printed news)[1]. When used appropriately and accurately, it has the potential to increase performance as well as progress in a variety of fields, including education, business, science, as well as social.

Including the growth in internet users, social media is also growing in popularity. One such social networking site is Twitter, which has an exponential growth rate each year. One of the Tweets that emerge on the Twitter homepage is a news tweet; however, these news tweets are not organized into news classifications including such health, sports, technology, economics, tourism, and etc. The lack of categorization makes it hard for people to interpret or obtain data relevant to the particular categories they seek.

One alternative is to use a text classification approach, which could also detect and classify a few classifications of unstructured text using NLP during the classification stage. There are many issues with the text classification stage. One issue with text categorization is that the word that is the feature does not have enough qualities to explain a class [2]. The Feature Method is one method for dealing with these issues.

There are several approaches to feature expansion. One technique is to use WordNet, which is an advanced electronic dictionary. Synonyms set (synset) in WordNet [3]. In this research, the classification process can be carried out using the BI-LSTM technique, with the relation of WordNet enquires. From the actual document, the additional words are hypernym as well as hyponym. Since WordNet has been used in English, the file will be transcribed using the Google Translate API following the initial feature expansion phase.

1.1 Deep Learning for Classification

ANN is a very efficient algorithm. Essentially, it is motivated either by neurons in the human brain. The channels are approximately linear input-output mixtures. The mixture is then put through a series of non-linear functions known as activation. Learning NN is as easy as conducting a difference in calculus once and then repeat, a system called as backpropagation, as well as an optimizer known as gradient descent.

1.2 Bidirectional Long Short-Term Memory

Network In comparison to conventional feed forward systems, RNNs include a novel approach to dealing with data by analyzing the interactions among two data points in the sequence [4]. Furthermore, LSTM networks are equivalent to RNNs in that a built-in cell unit displaces the upgrade of the hidden layer, as shown in Fig.1. As a consequence, they perform better when it comes to detecting as well as exploiting long time sequential correlations in a huge amount of instance space. While organized vectors created from M elements are produced from the n-gram layer in the final step of suggested technique, they are fed into the Bi-LSTM. Bi-LSTM, which was first suggested in [5], combines the benefits of LSTM and bidirectional RNN. In comparison to the preceding, it has the authority to receive both past as well as future contextual information, making it perfect for sentences. This type of NN is made up of two types of LSTMs that are fed data both immediately as well as vice versa at the same time.

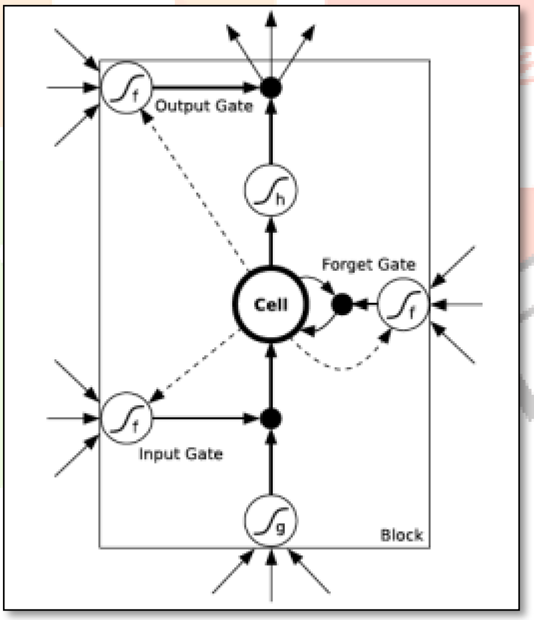

Figure 1.1 : LSTM Node [6]

At time t, the concatenation of both directional LSTM outputs can be:

$$h_t = [\overrightarrow{h}_t, \overleftarrow{h}_t] \quad (1)$$

in which $[\overrightarrow{h}_t, \overleftarrow{h}_t]$ signifies the directional and inverse LSTM outputs respectively. They are measured as:

$$\overrightarrow{h}_t = f(W_{x\overrightarrow{h}} x_t + W_{\overrightarrow{h}\overrightarrow{h}} \overrightarrow{h}_{t-1} + b_{\overrightarrow{h}}) \quad (2)$$

$$\overleftarrow{h}_t = f(W_{x\overleftarrow{h}} x_t + W_{\overleftarrow{h}\overleftarrow{h}} \overleftarrow{h}_{t+1} + b_{\overleftarrow{h}}) \quad (3)$$

in which W represents the weights of the neural networks, b indicates biases and x_t indicates input data at time t.

h_t gives probability of classification prediction in the next time step, with all above, the final prediction to the input sentences by the whole system should be as:

$$y_t = \sigma(W_t h_t + b_t)$$

in which σ, usually in the form of sigmoid, is the activation function, W_t is the weights matrix and b_t represents bias vector. The architecture diagram is as Fig.2.

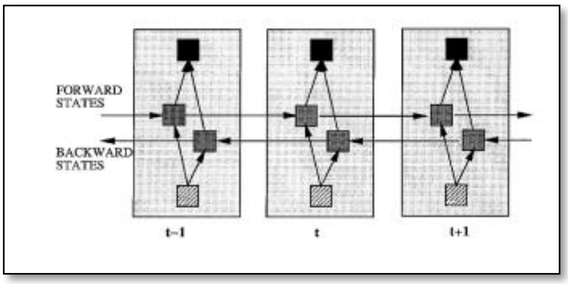

Figure 1.2: Bidirectional LSTM [5]

II LITERATURE SURVEY

Rao *et al.*, (2020) introduced a technique for developing a fast and reliable text classification scheme in both one-to-one and one-to-rest scenarios. The suggested method, dubbed n-Bi-LSTM, uses n-gram methods to transform natural text sentences into similar features to bag-of-words, which are then consumed into a bidirectional LSTM. The 2 components can take benefit of multi-scale feature representation and context knowledge more efficiently. Eventually, the method is designed using two labeled movie review databases, IMDB as well as SSTb, to test one-vs.-one and one-vs.-rest achievement. The outcomes demonstrate that suggested n Bi-LSTM method outperforms the basic LSTM as well as bidirectional LSTM techniques [7].

Zhai *et al.*, (2018) premised on Chi-square Statistics, suggested a technique for retrieving feature words Because the feature words that emerge together or individually in various situations may varies, authors classify texts by utilizing single as well as double words as features at the same time. Using suggested technique method, author ran experiments with the NB and SVM classification approaches. The efficiency of suggested technique was showed through the comparison as well as analysis of experimental outcomes [8].

Gozde *et al.*, (2021) To categorize the news texts, a particularly unique kind of RNN using the deep learning-based Fast-text design, LSTM was used. The Fast-text, Word2vec, and Doc2vec models are being used to categorize data from the data set, and their chances of success are contrasted. LSTM is used to categorize news data on the Fast-text framework, yielding the best results [9].

Saeed et al., (2021) choose the best-performing method from two Machine Learning techniques: NB, SVM, as well as three DL models: LSTM, NN with Keras (NN-Keras), and Neural Network with Tensor Flow (NN-TF). Utilizing two different English language news datasets, author tested five models. The models' function was measured using four metrics: accuracy, precision, recall, as well as F1-score. The findings demonstrated that DL techniques outperformed traditional ML techniques in terms of accuracy. All other designs tested were outperformed by the LSTM design. It obtained 94.21 percent accurate on average. The NN-Keras also performed well, with an average accuracy of 92.99 percent [10].

Batool alsukhni *et al.*, (2021) show how deep learning methods can be used to solve the Arabic multi-label text classifier model. Authors used Multilayer Perceptron (MLP) as well as RNN with LSTM to create two methods in Python. To enhance the quality of experimental data, all data has been cleaned. The LSTM technique produced a score of 82.03, while the MLP framework produced a score of 80.37, and both models were analyzed using the F1 score [11].

Umer *et al.*, (2020) obtained a database from the Fake News Challenges (FNC) website containing four types of stances: agree, disagree, discuss, and unconnected. The nonlinear data are classified into PCA and chi-square, which provide additional texture elements for fake news detection. First, feed the non-reduced feature set to the neural network, both with and without pre - processing. The outcomes of the dimensionality reduction approaches are then compared. PCA improves the classifier's performance for fake news detection by removing irrelevant, noisy, and redundant features from the feature vector. This method generates promising results, scoring up to 97.8 percent accuracy, which is significantly higher than previous studies [12].

Tanvir *et al.*, (2020) To classify the Bangla news headline, a classification technique is based on bi-directional LSTM was suggested. To improve the performance of proposed classification method, designers used the Bangla stop word corpus to remove stop words. Designers vectorized our text using Genism and the fast Text model to make it consistent with machine learning algorithm. Also created a dataset with approximately 10 lakh articles from various renowned Bangladeshi newspapers and 8 different classifications. The data was then given training in different models. Among these designs, Bi-LSTM obtained an accuracy of 85.14 percent which is outperforming on all others techniques [13].

III PROPOSED METHODOLOGY

3.1 Problem Formulation

With the rapid growth of online unstructured textual data, it has become a need to classify the text into categories so as to analyze and interpret relevant insights that contribute to decision making. Thus text categorization or classification is the process of assigning tags or classes to unstructured data according to its semantic content. This not only eases the procedure of indexing the rapidly growing data but also helps in retrieval of desired content from a large information space. A challenging problem in natural language processing, information retrieval and machine learning is the classification of the semantic content. News Titles provide a particularly great example of such classification owing to the fact that the content of news articles is generally precise, incisive and consistent. Most news groups generate a large number of news stories on a daily basis which should be available to individual users in an organized or classified way. The task of manually labeling news articles is not only tedious but also time consuming. It makes it difficult for an application to manually label the latest news articles and feed then to the readers in real time. This demands the use of a tool that would automatically classify the news articles in real time so that the users can access the latest labeled news stories.
Classification of news can be automated with the help of machine learning. The process is related to natural language processing where in the classifier tries to obtain the relationship between the text features and the text categories as per the labeled training dataset and then uses this classifier to label the latest news articles. The classifier tries to discover the most probable words for each category and takes them into consideration while classifying new articles. The classifier discards all such words that appear frequently for all categories. The main objective of news classification is to analyze news data technically to uncover patterns in news production and content.

3.2 Proposed Methodology

With the existence of a number of sources on the internet generating immense amount of daily news, there is a necessity to classify the news articles to make the information available to users quickly and effectively. So the task of news classification starts by collecting real time news articles from news websites through web scraping and then automatically classifying it using various classification algorithms. Thus news classification is a way to identify topics of untracked news as well as make Individual suggestions based on the user's prior interest. Some researches discusses various steps of news classification and implements a few algorithmic approaches including Naïve Bayes, Logistic Regression, SVM, Decision tree and Random forest for automatic classification of news articles into topics using the News dataset that contains articles belonging to five different categories. But proposed classifier has some drawbacks their performance degrades due to huge amount of data so that we will propose deep learning algorithm for news classification. In the proposed section we will use WordNet and Word Sense database to improve efficiency of classifier and to handle huge amount of data we will propose as classification deep learning approach i.e. Bidirectional Long/Short-Term Memory (Bi-LSTM). The task of News classification system we will train the Bidirectional Long/Short-Term Memory (Bi-LSTM) with multi-label classes.

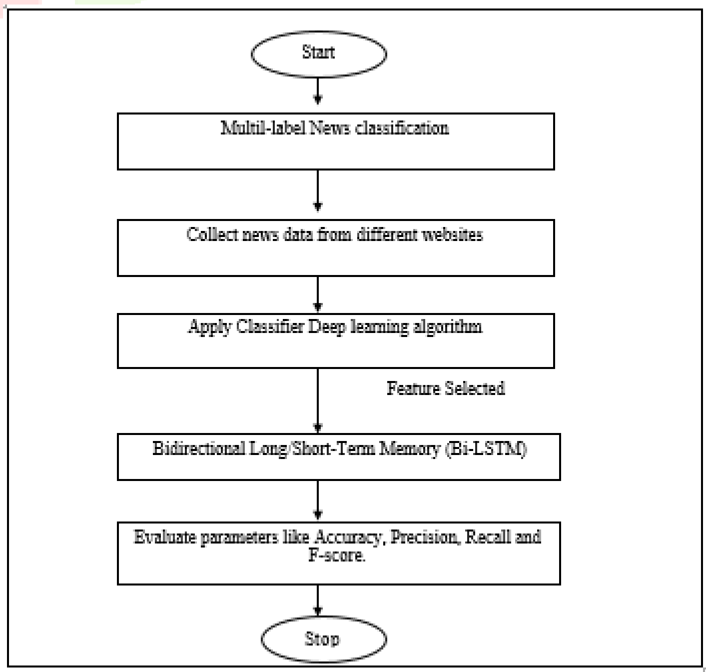

Figure 3.1: Flowchart of Proposed Work

IV. RESULTS AND DISCUSSION

The main objective is to establish a new classification scheme in python by employing an existing system. The purpose of this research is to use datasets that depict actual data so that the forecasting algorithm can make judgments from any advanced data. For proposed work, five test cases are used, each with a various testing size as well as Random_State. As well as the classification methods of Bi- LSTM were analyzed and contrasted using only real-time data using Random Forest, Naive Bayes, and MLP because Bi- LSTM- a deep learning approach- cannot be implemented on synthetic data due to the small size of the data. As a consequence, the bidirectional LSTM outcomes are thoroughly examined, and it is discovered that the suggested Bi-LSTM outshines the other classifiers.

The dataset used for news classification consisting of news articles collected from website. The dataset consists of many documents corresponding to stories in Thirty One topical areas namely business, entertainment, politics, sport and tech etc. The dataset is further divided into training and testing data. The number of articles in the training dataset which constitutes 60% of the entire dataset. The remaining 40% of the dataset forms the testing dataset. The machine learning classification algorithms have been implemented in python using a laptop .To measure the performance of all these algorithms, precision, recall, accuracy rate and f1 score have been considered.

By adjusting the size of the training / testing ratios, different cases are obtained. Cases are generated based on test size variation as well as random case variation.

Random state: that provide serves as a seed for the random number generator. As a result of which your train-test splits are always constant.

Batch Size: The batch size identifies the amount of specimens that will be transmitted across the system.

Test Case: characterized the testing dataset ratio.

Epoch: An epoch is frequently confused with an iteration. The computational time necessary to finish one epoch is the amount of batches or steps via partitioned packets of training data.

4.1 Test case 1:

Random_State=42,Test_Size=0.3, Epochs=20, Batch_Size=32

In the first case , the random state 42 and if we have taking testing ratio is 30 percent with the training ratio of 70 percent then the Bi-LSTM accuracy is 97.88 %.

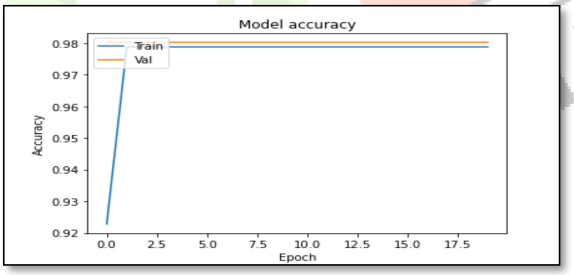

Figure 4.1: Model Accuracy

The model accuracy score reflects the model's potential to effectively forecast both positive & negative outcomes out of all assumptions.

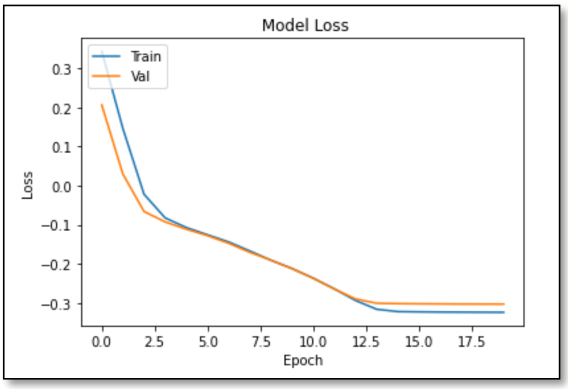

Figure 4.2: Model Loss

The estimation of the losses over every batch of training data is the Model Loss.

```
[4] Epoch 1/20
5/5 [==============================] - 12s 842ms/step - loss: 0.3430 - accuracy: 0.9228 - val_loss: 0.2059 - val_accuracy: 0.9801
Epoch 2/20
5/5 [==============================] - 1s 185ms/step - loss: 0.1456 - accuracy: 0.9788 - val_loss: 0.0295 - val_accuracy: 0.9801
Epoch 3/20
5/5 [==============================] - 1s 154ms/step - loss: -0.0221 - accuracy: 0.9788 - val_loss: -0.0660 - val_accuracy: 0.9801
Epoch 4/20
5/5 [==============================] - 1s 155ms/step - loss: -0.0825 - accuracy: 0.9788 - val_loss: -0.0926 - val_accuracy: 0.9801
Epoch 5/20
5/5 [==============================] - 1s 145ms/step - loss: -0.1070 - accuracy: 0.9788 - val_loss: -0.1112 - val_accuracy: 0.9801
Epoch 6/20
5/5 [==============================] - 1s 147ms/step - loss: -0.1254 - accuracy: 0.9788 - val_loss: -0.1277 - val_accuracy: 0.9801
Epoch 7/20
5/5 [==============================] - 1s 150ms/step - loss: -0.1437 - accuracy: 0.9788 - val_loss: -0.1471 - val_accuracy: 0.9801
Epoch 8/20
5/5 [==============================] - 1s 154ms/step - loss: -0.1667 - accuracy: 0.9788 - val_loss: -0.1703 - val_accuracy: 0.9801
Epoch 9/20
5/5 [==============================] - 1s 154ms/step - loss: -0.1903 - accuracy: 0.9788 - val_loss: -0.1901 - val_accuracy: 0.9801
Epoch 10/20
5/5 [==============================] - 1s 167ms/step - loss: -0.2116 - accuracy: 0.9788 - val_loss: -0.2119 - val_accuracy: 0.9801
Epoch 11/20
5/5 [==============================] - 1s 161ms/step - loss: -0.2363 - accuracy: 0.9788 - val_loss: -0.2361 - val_accuracy: 0.9801
Epoch 12/20
5/5 [==============================] - 1s 165ms/step - loss: -0.2635 - accuracy: 0.9788 - val_loss: -0.2640 - val_accuracy: 0.9801
Epoch 13/20
5/5 [==============================] - 1s 145ms/step - loss: -0.2931 - accuracy: 0.9788 - val_loss: -0.2892 - val_accuracy: 0.9801
Epoch 14/20
5/5 [==============================] - 1s 150ms/step - loss: -0.3155 - accuracy: 0.9788 - val_loss: -0.3002 - val_accuracy: 0.9801
Epoch 15/20
```

Figure 4.3: Epoch readings

```
#Accuracy of system
BiLSTMAccuracy = hist.history['accuracy']
BiLSTMAccuracy = (max(BiLSTMAccuracy))*100
print("Accuracy : "+"{:.2f}".format(BiLSTMAccuracy)+" %");
```

```
Accuracy : 97.88 %
```

Figure 4.4: Accuracy

4.2 Test case 2:

Random_state=40, Test_size=0.5, Epochs=20, Batch_size=32

In the Second case, the random state 40 and if we have taking testing ratio is 50 percent with the training ratio of 50 percent then the Bi-LSTM accuracy is 97.95 %.

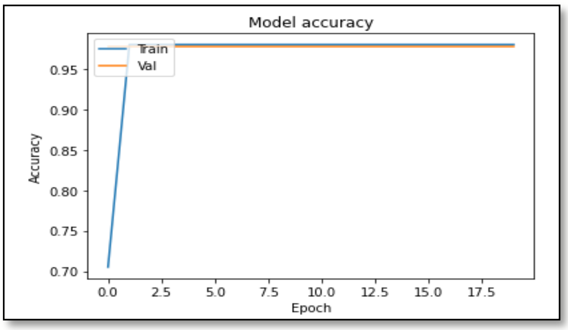

Figure 4.5: Model Accuracy

The model accuracy score reflects the model's potential to effectively forecast both positive & negative outcomes out of all assumptions.

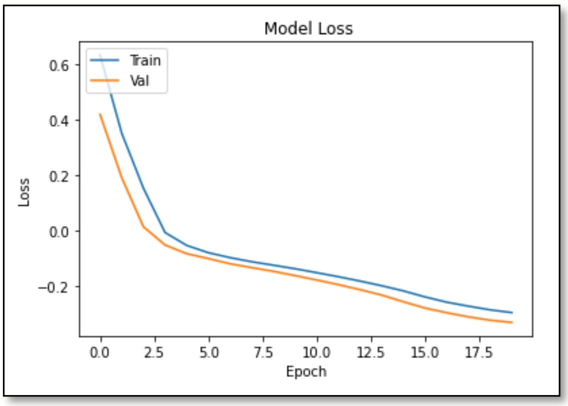

Figure 4.6: Model Loss

```
[24]
Epoch 1/20
4/4 [==============================] - 6s 439ms/step - loss: 0.6325 - accuracy: 0.7051 - val_loss: 0.4187 - val_accuracy: 0.9781
Epoch 2/20
4/4 [==============================] - 1s 169ms/step - loss: 0.3509 - accuracy: 0.9802 - val_loss: 0.1909 - val_accuracy: 0.9781
Epoch 3/20
4/4 [==============================] - 1s 168ms/step - loss: 0.1531 - accuracy: 0.9802 - val_loss: 0.0143 - val_accuracy: 0.9781
Epoch 4/20
4/4 [==============================] - 1s 167ms/step - loss: -0.0066 - accuracy: 0.9802 - val_loss: -0.0515 - val_accuracy: 0.9781
Epoch 5/20
4/4 [==============================] - 1s 170ms/step - loss: -0.0531 - accuracy: 0.9802 - val_loss: -0.0828 - val_accuracy: 0.9781
Epoch 6/20
4/4 [==============================] - 1s 166ms/step - loss: -0.0794 - accuracy: 0.9802 - val_loss: -0.1004 - val_accuracy: 0.9781
Epoch 7/20
4/4 [==============================] - 1s 165ms/step - loss: -0.0971 - accuracy: 0.9802 - val_loss: -0.1192 - val_accuracy: 0.9781
Epoch 8/20
4/4 [==============================] - 1s 179ms/step - loss: -0.1119 - accuracy: 0.9802 - val_loss: -0.1330 - val_accuracy: 0.9781
Epoch 9/20
4/4 [==============================] - 1s 165ms/step - loss: -0.1243 - accuracy: 0.9802 - val_loss: -0.1464 - val_accuracy: 0.9781
Epoch 10/20
4/4 [==============================] - 1s 167ms/step - loss: -0.1371 - accuracy: 0.9802 - val_loss: -0.1616 - val_accuracy: 0.9781
Epoch 11/20
4/4 [==============================] - 1s 164ms/step - loss: -0.1513 - accuracy: 0.9802 - val_loss: -0.1776 - val_accuracy: 0.9781
Epoch 12/20
4/4 [==============================] - 1s 168ms/step - loss: -0.1658 - accuracy: 0.9802 - val_loss: -0.1943 - val_accuracy: 0.9781
Epoch 13/20
4/4 [==============================] - 1s 166ms/step - loss: -0.1815 - accuracy: 0.9802 - val_loss: -0.2121 - val_accuracy: 0.9781
Epoch 14/20
4/4 [==============================] - 1s 166ms/step - loss: -0.1982 - accuracy: 0.9802 - val_loss: -0.2319 - val_accuracy: 0.9781
```

Figure 4.7: Epoch readings

```
#Accuracy of system
BiLSTMAccuracy = hist.history['accuracy']
BiLSTMAccuracy = (max(BiLSTMAccuracy))*100
print("Accuracy : "+"{:.2f}".format(BiLSTMAccuracy)+" %");
```

```
Accuracy : 97.95 %
```

Figure 4.8: Accuracy

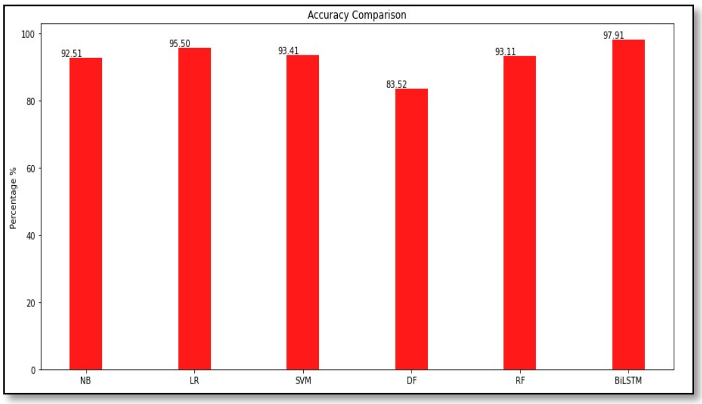

Figure 4.9: Comparsion of different techniques for Accuracy

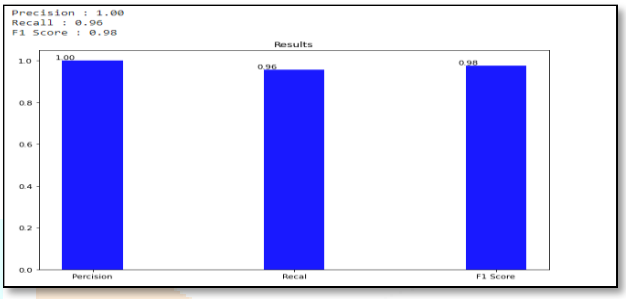

Figure 4.10: Graphical representation of Bi-LSTM Precision, Recall, F-measure

Table 4.1: Comparison between various algorithms in terms of Accuracy

S. No	Algorithms	Accuracy
1.	NB	92.51
2.	LR	95.50
3.	SVM	93.41
4.	DF	83.52
5.	RF	93.11
6.	Bi-LSTM	97.91

compares the forecast precision of ML methods (RF, SVM, or KNN), as well as suggested methodology. This Hybrid approach (Bi-LSTM) maintained an accuracy of 97.91 percent, which outperformed the estimated accuracy of each individual design. This method may be quite valuable in researchers to investigating news in way to support their instruction.

V.CONCLUSION

Text classification is broadly used in e-commerce as well as log message assessment. Furthermore, it is a necessary device in text data processing. Authors introduced a technique for developing a quick and accurate text classification scheme in this article. In this article, authors suggest a text classification framework in the context of NLP. To create structured features, we first use the WordNet and WordSense technique. The signals are fed into bidirectional LSTM neural nets, which create accurate prediction in both One-vs.-one and One-vs.-rest scenarios. Studies on datasets demonstrate that the suggested Bi-LSTM approach outperforms current methods.

REFERENCES

[1] Irfani, F. F., Ali Fauzi, M., & Sari, Y. A.., "News Classification on Twitter Using Naive Bayes and Hypernym-Hyponym Based Feature Expansion", International Conference on Sustainable Information Engineering and Technology (SIET), 2018.

[2] B. Sriram, D. Fuhry, E. Demir, H. Ferhatosmanoglu, and M. Demirbas, "Short text classification in twitter to improve information filtering", In Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval, pp. 841-842. ACM, 2010.

[3] W. Li, D. Ganguly, and G J. F. Jones. "Using WordNet for Query Expansion: ADAPT@ FIRE 2016 Microblog Track." In FIRE (Working Notes), pp. 62-65. 2016.

[4] Zhang, Y., & Rao, Z., "n-BiLSTM: BiLSTM with n-gram Features for Text Classification",IEEE 5th Information Technology and Mechatronics Engineering Conference (ITOEC),2020.

[5] Schuster, Mike, and Kuldip K. Paliwal., "Bidirectional recurrent neural networks", IEEE transactions on Signal Processing , Vol. 45,No. 11, pp. 2673- 2681,1997.

[6] Kawakami, K., "Supervised sequence labelling with recurrent neural networks", 2008.

[7] Yunxiang Zhang1, Zhuyi Rao, "n-BiLSTM: BiLSTM with n-gram Features for Text Classification", IEEE 5th Information Technology and Mechatronics Engineering Conference (ITOEC), 2020.

[8] Zhai, Y., Song, W., Liu, X., Liu, L., & Zhao, X., "A Chi-Square Statistics Based Feature Selection Method in Text Classification", IEEE 9th International Conference on Software Engineering and Service Science (ICSESS), 2018.

[9] Gözde N., Y. Safali, "Classification of Turkish News Content by Deep Learning Based LSTM Using Fasttext Model", International Artificial Intelligence and Data Processing Symposium (IDAP),2021.

[10] Saeed A.Alameri, Masnizah, "Comparison of Fake News Detection using Machine Learning and Deep Learning Techniques", 3rd International Cyber Resilience Conference (CRC), 2021.

[11] Batool alsukhni,"Multi-Label Arabic Text Classification Based On Deep Learning", 12th International Conference on Information and Communication Systems (ICICS),2021.

[12] Umer, M., Imtiaz, Z., Ullah, S., Mehmood, A., Choi, G. S., & On, B.-W., "Fake News Stance Detection Using Deep Learning Architecture (CNN-LSTM)" IEEE Access, 4(1), 2020.

[13] Md. Mahmudul Shahin, Tanvir , Shahriar Hasan Piyal, Md., "Classification of Bangla News Articles Using Bidirectional Long Short Term Memory", IEEE Region 10 Symposium (TENSYMP),2020.