Global Journal on Innovation, Opportunities and Challenges in AAI and Machine Learning
http://eurekajournals.com/IJIOCAAIML.html
ISSN: 2581-5156

Leveraging Bidirectional LSTM for Enhanced Sentiment Analysis: A Deep Learning Approach

Khushi Anil Teli[1]

[1]Computer Science and Engineering, Vellore Institute of Technology, Vellore.

Abstract

In this work, we use LSTM for this model to show that it is effective for NL processing tasks such as sentiment analysis as discussed herein. It was thus the purpose of this paper to leverage LSTM in capturing long-term dependencies of sequential data for improved identifications of sentiments in a big Twitter data set. The rationale for doing so is that the given set of data provided a huge number of textual inputs for training and testing of the proposed model as all the additional tweets were marked as positive, negative, or neutral emotion. Hyper parameters were also adjusted to set the best scenario since we_sess2rnted on believes that such probing areas like several LSTM units, batch size, learning rate, and dropout rates among others could best deliver. As predicted the better-proposed model and the newly proposed improved model got 84% of accuracy this confirms the efficiency of LSTM in sentiment analysis. This method brought out the seriousness of choosing the right model parameter and it also helped proclaim the flexibility of the deep learning model in handling Real-World noisier Social Media data for sentiment analysis.

Keywords: Sentiment analysis, LSTM, Twitter Data, Deep Learning, RNN.

Introduction

For instance, the determination of the emotional sentiment of the text also termed sentiment analysis, has turned into a key technique for ascertaining users' visions and actions, especially on the micro-blogging site-Twitter. While they are quite effective, sometimes traditional methods of sentiment analysis using machine learning fail to capture the sequential and contextual relationship observed in text [1]. Now, when it comes to satisfying the condition of text analysis from the aspect of long-term dependencies, Recurrent neural networks (RNN) of the Long Short-Term Memory (LSTM) type can be considered one of the most powerful models of deep learning, applicable when performing sentiment analysis [2]. In this paper, therefore we are concerned with whether or not the utilization of LSTM networks can significantly improve performance in sentiment categorization. So our method aims to mitigate the problems of noisy data in sentiment analysis by collecting a large amount of Twitter data, fine-tuning parameters, and evaluating the model. This technique should give a more accurate and reliable way of gauging emotions in social media [3].

The expandability of media coupled with the massive information flow from the users has made it crucial for various researchers, companies, and even the government to conduct this analysis. Some of the sites, which contain real-time views, reviews, and comments on various issues might include Twitter [4]. However, it does not translate easily into a process through which it is possible to analyze such big and noisy data. Satire, negation, and context-dependent interpretations are some of the rich patterns found in textual data that early sentiment analysis algorithms often missed because they were designed to use rule-based systems or traditional machine learning classifiers such as SVM and Naive Bayes [5]. Popular architectures such as LSTM with the coming of deep learning have demonstrated reasonable improvement in sequential data applications such as language modeling, translation, and most recently sentiment analysis [6]. This is because LSTM networks have the capability of preserving very important information from the previous sections of the text while discarding other irrelevant information-this is very important when processing context-rich phrases as found in tweets. However, it is possible with deep learning models, which can be trained in an end-to-end manner thus eliminating the need for extensive feature engineering [7].

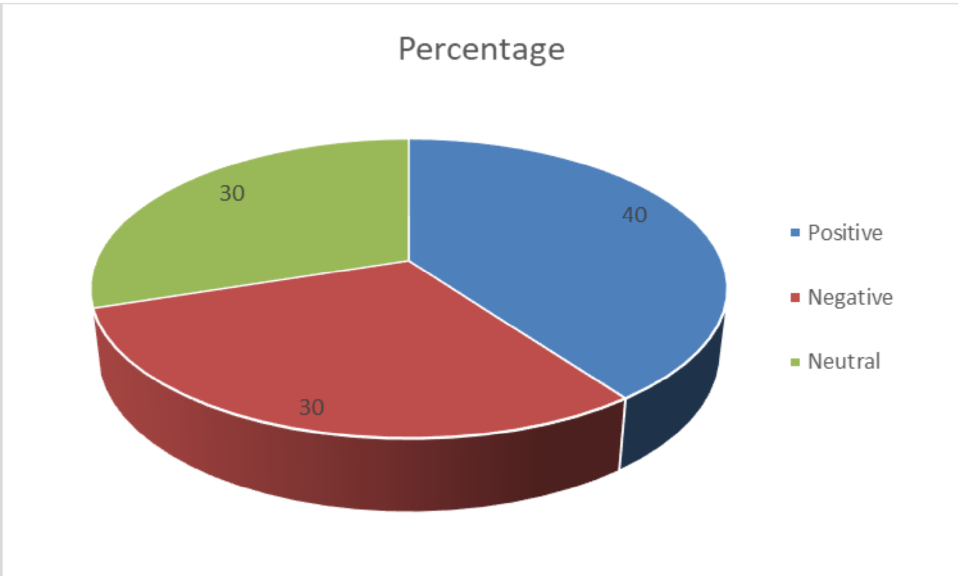

Figure 1: Sentiment distribution in the Twitter dataset

Figure 1. visually represents the sentiment distribution in a Twitter dataset. It is divided into three categories: Positive (40%), Negative (30%), and Neutral (30%). The chart reflects the balance of different sentiment classifications, indicating that 40% of the analyzed tweets expressed positive sentiment, while both negative and neutral sentiments each accounted for 30%. This distribution offers insights into the overall mood or opinions reflected in the dataset, helping to understand public opinion or reactions across various topics [8]. There are two types of negative and positive sentiments about the product in question, for example, which are always combined in actual data, with negative or positive sentiments predominating. Such imbalance may distort the performance of the model in favor of the dominant class [9]. The pie chart is good for simple illustrations and when doing a sentiment analysis, one has to ensure they have equal distributions to be fair in their analysis [10]. To gain better results in LSTM for sentiment

analysis, it is vital to enhance the model's performance through hyperparameter tuning. Reduced to its essence, sentiment analysis is an integral part of personalized advertising, active users' monitoring, and content moderation for corporations such as Twitter and Facebook as well as Google. Sentiment analysis is also being used by recommendation systems and customer support bots to give more accurate and personalized results based on the user's emotions [11].

This is in line with the increasing importance of monitoring the public mood in modern society which relies on data analysis [12]. Tens of millions of people or more people post their thoughts on social networking sites such as Twitter daily, and these Web sites provide a tremendous amount of unstructured data that researchers, businesses, and legislators can mine to gain such knowledge. Traditional approaches of sentiment analysis often succumb to subtleties in language or contextual interdependency as well as sarcasm and other richness of human feelings [13]. This constraint highlights the need for better approaches to mitigate complexity in textual data as compared to the organizational structures investigated here. Conventional RNNs known as LSTM networks have been proposed as a potential solution because of their capability to memorize long-term dependencies present in the temporal data thereby making them ideal for sentiment analysis [14]. Largely, this paper seeks to demonstrate this by analyzing the use of LSTM for sentiment analysis to prove that deep learning could improve the accuracy and reliability of sentiment predictions especially in complex real-life data from Twitter [15].

In this regard, using Long Short-Term Memory (LSTM) networks, the research that is proposed in this paper aims to enhance the implementation of sentiment analysis while offering a superior classification of emotions [12]. The literature review is provided in Section II wherein the focus is placed on how LSTM networks and other deep learning models can improve sentiment prediction, particularly on social media which is unstructured and often contains sarcasm and other related features. To ensure that the study reflects on as many linguistic, cultural, and contextual variables as possible. An approach for selecting a diverse set of datasets ranging from Twitter data to other social networks like Facebook is explained in Section III. Section IV, it is describes how LSTM networks enable to identification of long-term dependencies from textual data for more precise determination of contours and sentiment categories in parallel to CNNs and other conventional machine learning techniques. Section V is the conclusion where the current study offers a viable way for evoking a sentiment analysis based on LSTM to other applications such as social media trends, customer feedback, and brand monitoring.

Literature Review

Table 1: Comparative Analysis of Sentiment Analysis Models

Authors	Dataset	Methodology	Accuracy
Authors [1]	Mendeley data and manual tweets	VADER, Transformers-Roberta	Evaluated using multiple metrics
Authors [2]	Three datasets for product sentiment	ULM-SVM deep learning	98.7%
Authors [3]	1.6 million tweets	Machine learning and deep learning	Compared using multiple metrics
Authors [4]	228,207 COVID-19 tweets	LSTM, BiLSTM-CNN	93.66% - 94.10%
Authors [5]	Persian dataset	Pre-trained deep learning + human agent	4% F1 score improvement
Authors [6]	Pharmaceutical dataset with emoticons	Text and emoticon analysis using BiLSTM	85%-90%
Authors [7]	Various datasets in social media	Deep learning, optimization-based models	Compared using various datasets
Authors [8]	Second wave COVID-19 tweets	LSTM, CNN	Classified into five categories
Authors [9]	Expression dataset	RNN, LSTM	Evaluated using multiple metrics
Authors [10]	Twitter dataset	Naive Bayes	Polarity classification

Opinion polls from social media channels are highly relevant to the determination of the prevailing sentiment. The authors [16] focused only on sentiment analysis of the messages that are posted on Twitter and categorized the messages into positive, negative, and neutral categories using VADER and Transformers-RoBERTa. In pre-processing, they simplified the dataset by removing some elements such as hashtags and URLs. To prove the effectiveness of the auto-generated Machine Learning-based sentiment analysis tools for sentiment analysis on Twitter, they assessed the models' performances based on parameters like confusion and accuracy, precision, recall, and F1-score.

The authors [17] developed an approach for sentiment analysis of the data obtained from Twitter using ULM-SVM, a deep learning tool. It concerned itself with placing attitudes regarding specific goods and services into definable categories. The accuracy that was determined was 98 %. It seems that the addition of Adaboost takes its accuracy to 7%, after having been tested on several datasets. This study focuses on the potential benefits that DL methods may bring into play when data set is big in providing meaningful information.

To the best of my knowledge, the authors [18] categorized That is why, in this research, 1.6 million tweets are classified into the positive and negative sentiment categories applying machine learning and deep learning models. This paper illustrated how such models can potentially provide organizations with valuable knowledge regarding the public's perception, thereby enabling organizations to make informed decisions based on a model comparison based on metrics including accuracy, F1-score, recall, and precision.

The authors [19] have explored the comments on the Twitter platform, particularly in the context of the COVID-19 epidemic using Sentiment analysis. They used LSTM and BiLSTM-CNN models to categorize attitudes into three categories: to classify as positive, negative, and neutral using a dataset of more than 228,000 tweets. Their findings indicated the high levels of accuracy of results provided by the software which stood at 93%, 66%, and 94.10 % for these models, which confirms their effectiveness when it comes to evaluating the population's understanding during critical periods, such as the epidemic.

This problem is particularly apparent in Persian, where there are limited-labeled datasets, that were covered by the writers [20]. To address this problem, they introduce a semi-automatic approach that combines annotation with state-of-the-art deep learning algorithms. Due to this, they develop a new dataset and thereby improve ParsBERT baseline F1 by 4%. The technique shows how deep learning models with human labor can significantly enhance sentiment analysis performance, particularly with low-resource languages.

To ascertain the classification of emoticons into sentiment analysis for feedback on pharmaceuticals the research was conducted by the author [21]. They further showed that emoticons gave a significant difference in results by employing the BiLSTM and using machine learning algorithms on both text-only and text-plus-emoticon datasets. Having achieved 85 to 90% accuracy with this deep learning approach, the authors showed that moving with emoticons improves sentiment analysis results more than conventional methods like machine learning algorithms.

The authors [22] stressed how the increase in social media data necessitates the importance of sentiment analysis. They explored optimization-based models and deep learning and compared the effectiveness of two machine learning approaches. They proved the ability of deep learning architectures and optimization techniques to increase the correctness of the sentiment analysis and described them as being very effective in the extraction of useful information from people's opinions by exploring well-known datasets and describing their characteristics.

In India, during the second wave of COVID-19, the authors [23] analyzed public opinion through Twitter data. As for classification, they categorized the tweets into five classes namely "Strongly Negative" to "Strongly Positive". It used CNN and LSTM for training models and Python VADER for attitude classification. By applying the same machine learning algorithms together with a labeled dataset, their results showed how the impact of the public attitude may be accurately measured during a crisis like the current epidemic.

The author's [24] main intention was to use deep learning and recurrent neural networks (RNN) to develop an expressive sentiment analysis model. They divided attitudes into three categories: classification of words into positive, negative, and neutral using the Long Short-Term Memory (LSTM) model. Thus, using the measures, including accuracy, precision, recall, and F1-score, they were able to state that using the deep learning approach based on RNNs, it is possible to perform the tasks associated with sentiment categorization and obtain rather accurate data on sentiment.

Text preparation techniques and Naive Bayes classifiers were used by the authors [25] to analyze the sentiment of tweets. Some of the techniques that were applied in text processing include stop word removal and lemmatization or stemming to make the text more normalized. The outcomes showed how preprocessing techniques can be applied in combination with basic approaches like NB to obtain sentiment info from tweets and how they may be sorted by the polarity of the data: positive, negative, or neutral.

Methodology

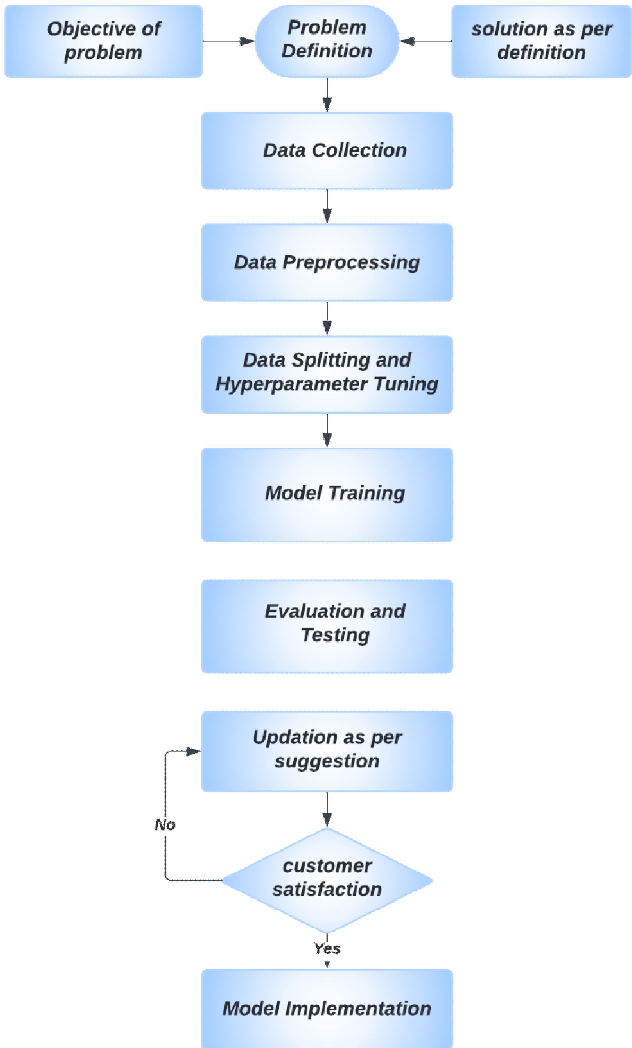

Figure 2: Proposed Methodology

The use of the LSTM model with the hyperparameter adjustment ensures that there will be an enhancement in the performance of the models by practicing these procedures. This technique also mitigates the likelihood of an inefficient Deep Learning model that fails to classify the sentiments with the scale of accuracy needed, from the initial stage of data acquisition and Data Preprocessing right down to the model deployment and hyperparameter tuning step as shown in Figure 2.

Data Collection and Preprocessing

The first step in the approach is the collection of good datasets and this is a crucial but basic requirement for any further model, especially that of sentiment analysis. The data in this case can be pulled from Twitter. The dataset contains 74683 records and two columns illustrating the text and their labels. Sentiment labels referring to certain textual examples should be included in the collection. After collection, characters that are other than alphabets and numbers are eliminated; identifications of URLs, punctuation marks, and converting all the text to lowercase. Then words in phrases are split up Next the phrases are segmented to break down the phrases into several words. The meaning of words is then encoded by dense vectors that we derive from these tokens employing word embeddings. The third thing about feeding data into an LSTM model is that it is always advisable to ensure that the sequences are of equal length while feeding them into the model by either padding or truncating.

Data Splitting

After the preprocessing step of the dataset, the dataset is divided into a training set, a validation set, and a test set. In data partitioning, training typically takes up to 70% of the data while validation takes up to 15% and testing takes up to 15%. This makes the model not fit the data points and ensures that changes that are made are the best that the model can use to handle new data in the future. The given model is learned on the training dataset; tuning of hyperparameters is done on the validation dataset; whereas the final evaluation of the given model is made on the test dataset.

Hyperparameter Tuning

As for the crucial point to enhance the performance, the hyperparameter tuning is the main step to tune the parameters that govern a model’s training. This approach self-tunes several parameters toward the direction that delivers superior performance. The first of these rates as stated by Kaskavelis (2005) is the learning rate and such rates including 0. 001, 0. 01, and 0. 1, are used to establish the right rate which will enable the model to converge in the best way possible without going up or down the expected middle rate. Tuning the learning rate to the optimal threshold may be achieved via a grid search or a random search method. The next option is epochs and it refers to the total amount of full swept through the given set. An early stopping procedure is used to mitigate the problem of overfitting, although different epoch values are tried out. One thing to note is that training is stopped right after one does not get an improved performance on the validation set. Stop loss contribution is centered on tracking the cross-entropy loss function which is standard in the classification setting. The model continues training in this case only if the validation loss starts to rise implying that overfitting is likely to have set in. Another crucial parameter that is adjusted to see if normalizing the output from every layer speeds up the model's convergence is batch normalization. Raining may become more stable with the addition of batch normalization layers. Last but not least, by randomly removing input units during training, dropout regularization is adjusted to avoid overfitting. Standard dropout rates of 0.2, 0.4, and 0.5 are evaluated to determine the best trade-off between model performance and regularization.

Model Training

The training procedure starts when the LSTM model is built and the hyperparameters are adjusted. Using the selected hyperparameters and training data, the model is trained. Gradient descent and backpropagation are used by the model to continually update its weights during training. Every epoch ends with a performance check of the model on the validation set. If the validation loss rises over a certain number of epochs, early halting may be implemented.

Evaluation and Testing

The test set is used to evaluate the model once it has been trained. Important performance measures including recall, accuracy, precision, and F1-score are computed to find out how well the model classifies attitudes. To have a deeper understanding of the model's predictions across several sentiment classes, one might create a confusion matrix. Against evaluating the model's progress, its performance may also be compared against baseline models (such as logistic regression or SVM).

Model Implementation

The basic requirement for deployment is that the model has to be at an acceptable performance level. Creating a simple API or web interface through which the user can feed plain text data for analysis falls under the deployment phase. Depending on the specific framework utilized, the model is then deployed into a format such as a PyTorch script or a TensorFlow Saved Model among others. When implemented on the cloud platform including Amazon web service, Google Cloud, or Microsoft Azure, the model may be used to ensure scalability and real-time analysis. The time of inference and amount of required memory could be optimized and reduced with the help of methods like model pruning andTensorRT. If the performance of the model is worse, then retraining is anticipated, and a performance check is also employed to monitor the model on new data from real-life situations.

Result

Table 2: Training Report at Different Epoch Values

Epoch	Accuracy	Loss	Val_Accuracy	Val_Loss
1	0.5292	1.087	0.6988	0.7711
2	0.7475	0.659	0.7579	0.639
3	0.8198	0.482	0.7882	0.5718
4	0.8584	0.381	0.8055	0.5503
5	0.8853	0.3086	0.8185	0.5379
6	0.9064	0.2536	0.8302	0.5292
7	0.9207	0.2122	0.836	0.5557
8	0.9349	0.1731	0.836	0.5821
9	0.9439	0.1491	0.8448	0.5982
10	0.9479	0.132	0.8451	0.6111

Global Journal on Innovation, Opportunities and Challenges in AAI and Machine Learning - Vol. 8, Issue 1 – 2024

Table 2 shows the comparison of the model's performance of Validation Accuracy (Val_Accuracy), Validation Loss (Val_Loss), and Accuracy during ten epochs of training. The epoch loss decreases significantly 1.087 to 0.132; however Accuracy starts from 0 and rises progressively with the level 0.5292 in epoch 1 to 0.9479 in epoch 10. Continuing this rise from 0.6988 to 0.8451 in Validation_acccuracy Optimizes through epoch 10, Accuracy_validation shows better capability of generalizing the models. On the other hand, Validation Loss first decreases, and attains its minimum of 0.5292. It reaches 0.5292 in epoch 6 and then slowly increases up to 0.6111, It will also be observed that the number of epochs has led to overfitting of the model in the subsequent epochs.

Precision

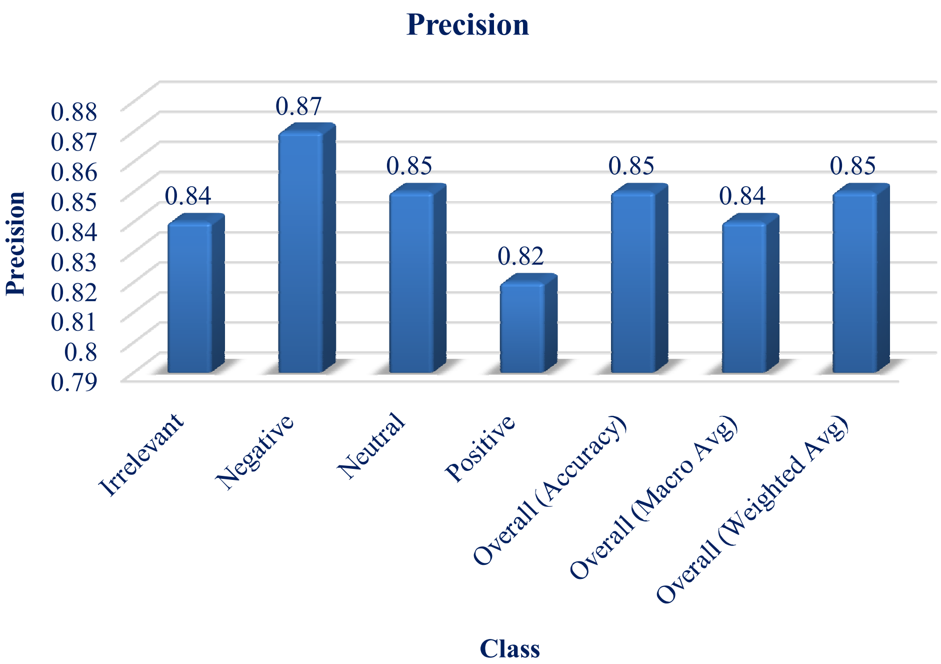

Figure 3: Precision at different Classes

The accuracy values for the irrelevant, negative, neutral, and positive classes that are determined in the training and testing of a sentiment analysis model are depicted in Figure 3. Recall quantifies how tolerant a model is to the false negatives by determining the true positives among the predicted positives. At 0.87, moreover the accuracy of the Negative class is the highest suggesting that the model is doing quite well in identifying negative emotions. Press, and closely with accuracy ratings of 0.85 and 0.84. Proper classification of articles leads to the identification of the articles that can be classified under the two categories that are the least relevant to the work and these are Neutral and Irrelevant articles constituting 0.84 respectively. On the other hand, the pleasant class has the lowest accuracy rating of 0.82, implying slightly less accurate statements concerning pleasant emotions. With the weighted average of 0.85 together with a macro average of 0.84, the model's overall accuracy is as follows, Thus the accuracy rate for testing on the different models is as follows. 0.85, this proves a fairly good performance of the model in all four classes of sentiment.

Global Journal on Innovation, Opportunities and Challenges in AAI and Machine Learning - Vol. 8, Issue 1 – 2024

Recall

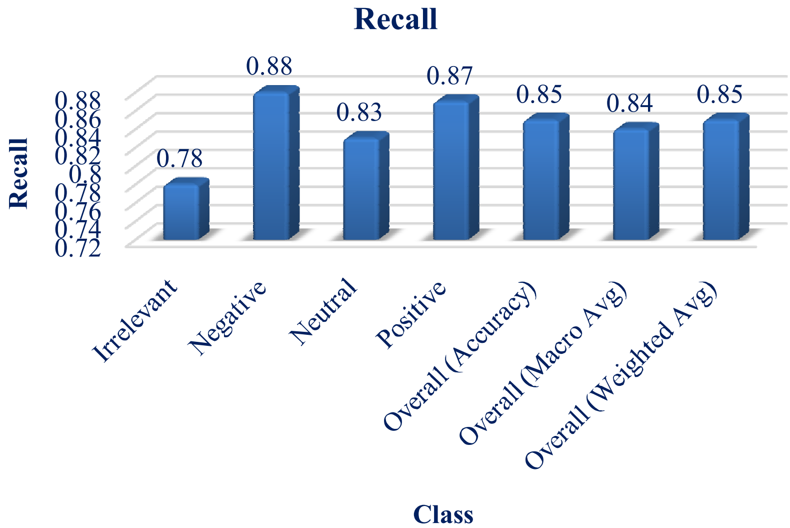

Figure 4: Recall different Classes

The recall values for a sentiment analysis model are shown in Figure 4 for the following classes: overall performance, they are considered as irrelevant, negative, neutral, positive, and positive. The model's recall defines how well the model filters through the items in a specific class to identify each of the relevant ones. There was no recall of 0.88, It shows the Maximum accuracy in the Negative class which indicates that it can detect negative thoughts. Recall of the positive class is 0.87, the Positive class is just following it with good rates showing that the positive emotion has been identified successfully. In terms of recall, the Irrelevant class has the lowest recall of 0.78, While the recall of the Neutral class is slightly lower at 0.83, which means there is still room for improvement in identifying irrelevant occurrences. So once we applied the weights the average turns out to be 0.85 while that of the macro average is 0.84 a ratio which implies different classes are performing an average of 84 % on the activities expected of them. The overall recall is 0.85.

F1-Score

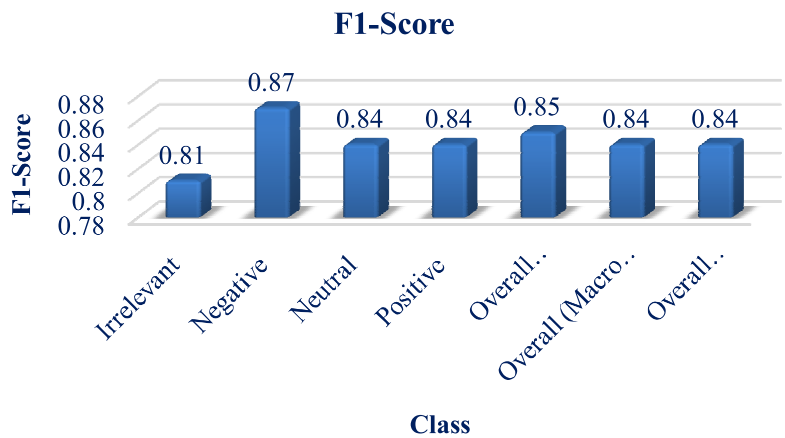

Figure 5: F1-Score at different Classes

The sentiment analysis model's F1-scores for the following classes are shown in Figure 5, It included irrelevant, negative, neutral, positive, and overall performance metrics of network computers. The further combination of accuracy and recall in the form of the harmonic mean is the F1-score, which provides a rational evaluation of the model. Hence, model I with an F1-score of 0.87, the Negative class, as already explained, illustrates how good the model is in labeling negative emotions. The F1-score shows a fairly good and consistent ability of the model in recognizing these feelings as 0.84 for both the Positive and Neutral classes. Curiously, the average of F1-score of all participants was 0.81, a result lower than others, which suggests that the Irrelevant class has the lowest overall accuracy, which also points to some degree of difficulty in accurately predicting irrelevant material. Here, it is possible to note that two types of averages-macro and weighted average-are equal to 0.84. A test accuracy score of 84% which is an affirmation of an average model across the different classes. The precision of the model can be calculated as an average of 0.85.

Accuracy

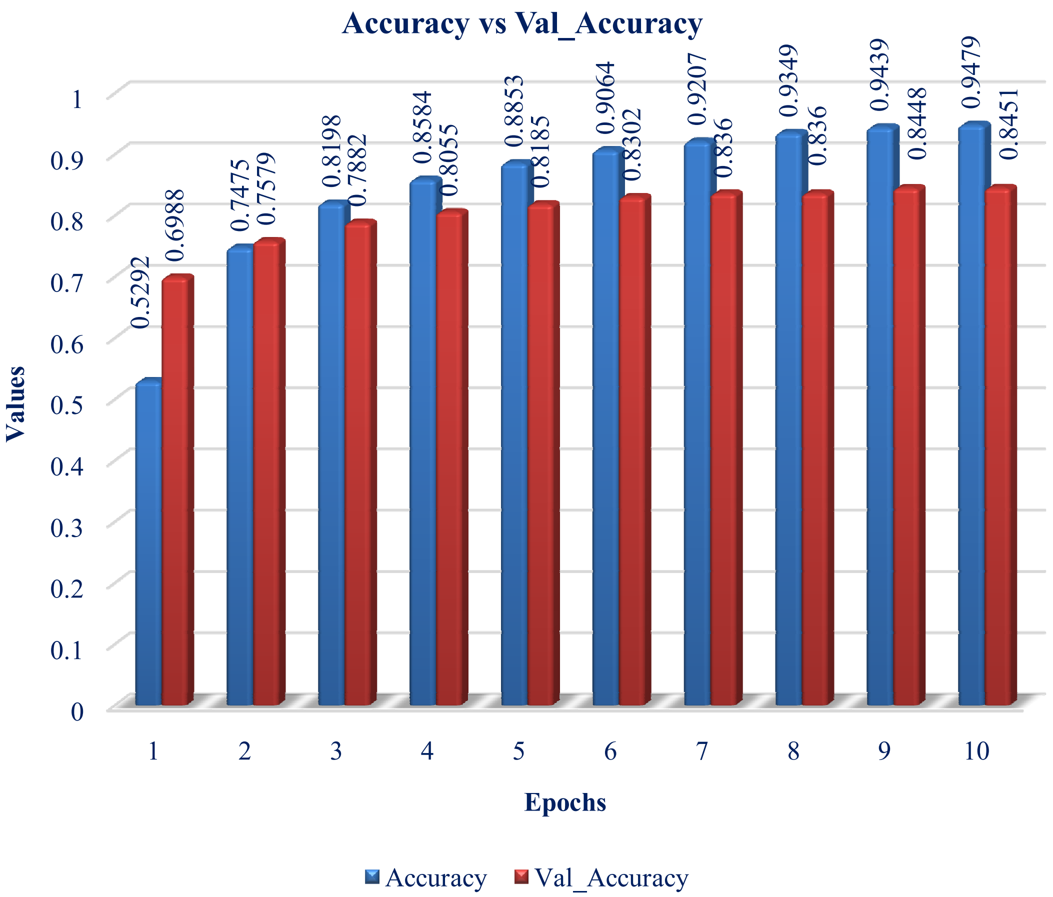

Figure 6: Comparison between Accuracy and Val_Accuracy

Figure 6 illustrates where a sentiment analysis model is training in blue and validating in red over ten epochs. Although the validation accuracy starts at a higher value at 0.6998 in epoch 1, the training accuracy is comparatively low that is, 0.5292, which may be attributed to the better generalization of the model on unseen data. Thus, when it comes to increasing epochs, both of these accuracy metrics are witnessed to improve. The accuracy of training rises to 0.9479. Increase epoch From 4 to 6 the training accuracy is 0.9604 and the validation accuracy climbs to 0.8302 This design decision thereby equates to, hence a significant enhancement of performance. In addition to this, after the epoch of training, the model increases its accuracy to the nearest integer only, which indicates that the model is converging; with the training accuracy of As for the training accuracy it is 94.479%, the validation accuracy on the other hand is equal to 0.8451. It therefore implies that after epoch 6 the performance of the model becomes constant.

Loss

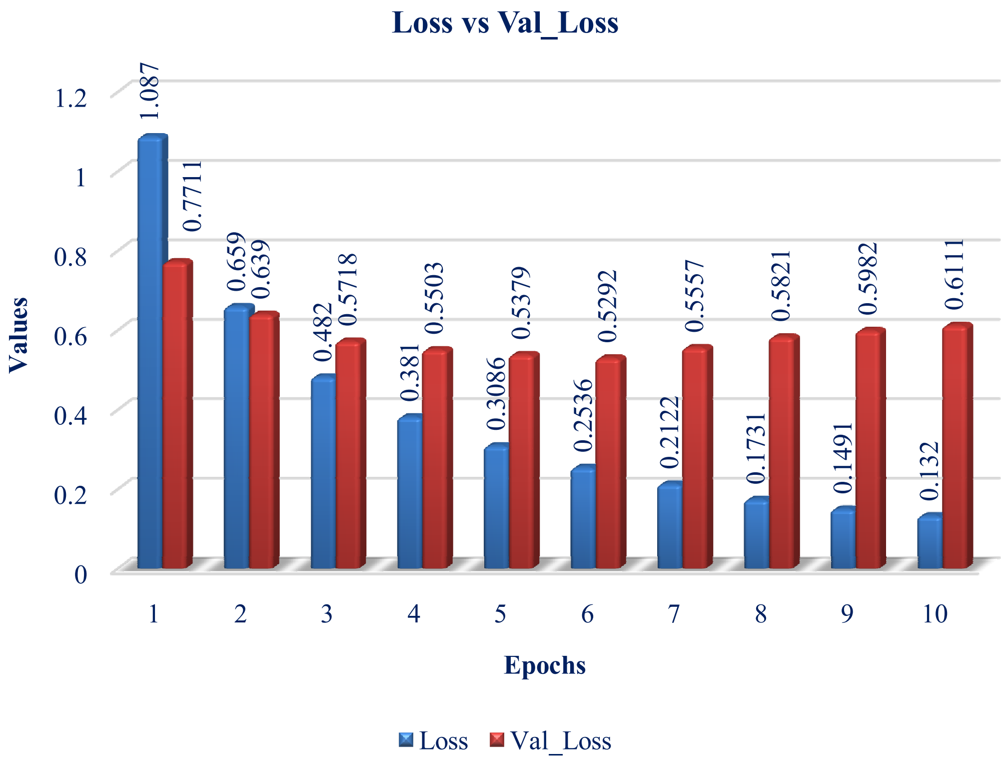

Figure 7: Comparison between Loss and Val_Loss

Plotted in Figure 7 below is the training loss in blue and the validation loss in red for the ten epochs of training. This is true because in the case of Manpower utilization the lower the number, the better it is considered a higher employee performance. This observation of the difference between the forecasted value and the actual observation is referred to as loss in the context of the model. Epoch 1 shows a high training loss of 1, Epoch 1 from the training of the model in Epoch 1 of value 1.087 and a validation loss of zero as shown below. Before iteration 0.7711, the model showed generalization issues. Both losses decrease significantly with better training. As seen in Epoch 5, there is an enhanced performance with a generalized model with the training loss having been reduced to 0.3086 Epoch 6 shows 0.2536 and the validation loss to 0.5292. Next, the training loss keeps on getting smaller until it becomes zero as shown below; 0.132 by epoch 10. However, as you can observe, after epoch 7 validation loss increases slightly and reaches 0.22 At epoch 10 the accuracy of the model was 0.6111, which means the model might have to overfit because it has shown more accuracy in the case of the training set rather than validation set.

Confusion Matrix

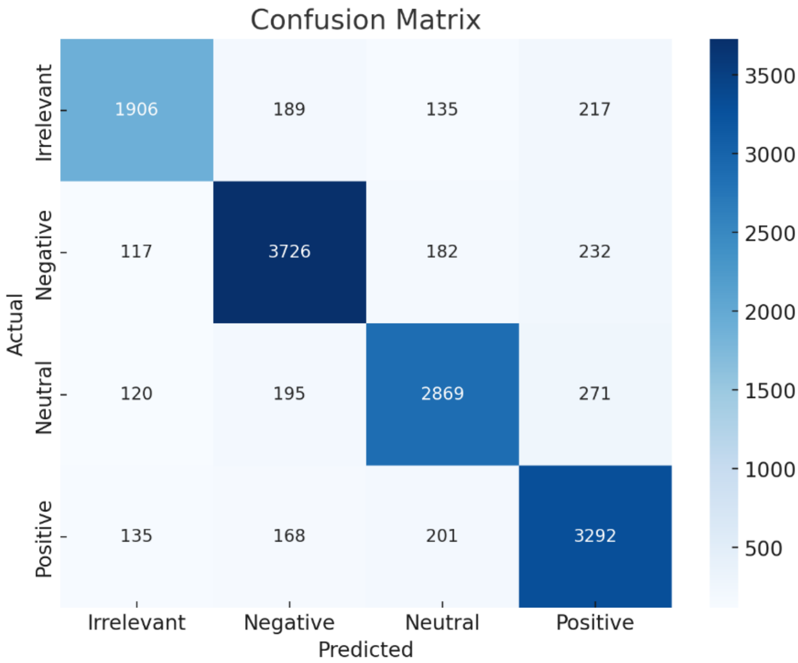

Figure 8: Confusion Matrix of results

Figure 8 depicts a sentiment analysis model's confusion matrix, demonstrating the model's prediction accuracy over four classes: In traditional sentiment analysis, it is possible to identify words and documents as irrelevant, negative, neutral, and positive. While off-diagonal entries present incorrect classifications, diagonal entries present correct classifications. In an irrelevant class false classification of the negative class was 189, of the neutral class was 135, and the positive class was 217. Out of the whole instances, only 1906 of them are correctly identified. Thus, the Negative class contains 3726 instances properly identified and significantly fewer misclassifications; therefore, it has the highest percentage of right predictions. There is a high variance in the forecast of other classes themselves; however, for the Neutral class, we have 2869 valid forecasts. Thus, the Positive class has performed 3292 true predictions some of which are held in common with other classes. Given that, the matrix gives a comprehensive idea about the effectiveness of the model together with the scope for refining the class difference.

Conclusion and Future Scope

In this study, this work empirically proved that LSTM networks used appropriately could offer a higher accuracy of about 85% toward better sentiment analysis. This included; the learning rate, epoch, stop loss, batch normalization, and dropout rates, which were some of the areas that needed to be tweaked for better performance of the trained model. The LSTM model was found to be effective for sentiment classification since the model was capable of recognizing patterns in the order of the text inputs. By using these refined parameters, we found out that it was possible to achieve a balance between generalization on the one hand and on the other, model complexity through which sentiments are predicted accurately. The achieved accuracy proves the model's ability to recognize and sort the emotions in different text inputs.

However, it is good to note that when 'forecasting' the accuracy rate was at 85% meaning that there is always more that could be achieved. For better understanding and extracting more fine-grained features from text input, future work can explore the integration of LSTM with other architectures, such as Hybrid, CNNs, etc, In the current study context however, LSTM's ability was sufficiently demonstrated to extract the required characteristics from the movie reviews. Incorporating attention processes also might help the model to pay more attention to other important passages of the text which in turn improves the overall performance of the work. Moreover, examining the transfer learning strategies can enhance the performance of sentiment analysis by feeding better contextual information by integrating the pre-trained language formats such as BERT or GPT. Their robustness within the context and variation will be secured by constant monitoring and training with actual and operative data sets.

References

J. Ferdoshi, S. D. Salsabil, E. R. Rhythm, M. H. K. Mehedi and A. A. Rasel, "Unveiling Twitter Sentiments: Analyzing Emotions and Opinions through Sentiment Analysis on Twitter Dataset," 2023 Computer Applications & Technological Solutions (CATS), Mubarak Al-Abdullah, Kuwait, 2023, pp. 1-7, doi: 10.1109/CATS58046.2023.10424206.

L. Sudha Rani, S. Zahoor-Ul-Huq and C. Shoba Bindu, "A Deep Learning Approach for Twitter Sentiment Analysis using ULM-SVM," 2024 11th International Conference on Computing for Sustainable Global Development (INDIACom), New Delhi, India, 2024, pp. 1639-1643, doi: 10.23919/INDIACom61295.2024.10498677.

V. Pandya, A. Somthankar, S. S. Shrivastava and M. Patil, "Twitter Sentiment Analysis using Machine Learning and Deep Learning Techniques," 2021 2nd International Conference on Communication, Computing and Industry 4.0 (C2I4), Bangalore, India, 2021, pp. 1-5, doi: 10.1109/C2I454156.2021.9689241.

M. Amirpour, H. Shakibian and R. Azmi, "Sentiment Analysis Of Twitter Data Towards COVID-19 Vaccines Using A Deep Learning Approach," 2023 9th International Conference on Web Research (ICWR), Tehran, Iran, Islamic Republic of, 2023, pp. 7-11, doi: 10.1109/ICWR57742.2023.10139297.

G. G. Shirazi, R. Azmi and H. Shakibian, "A Semi-Automated Labeled Data Generation Approach Based on Deep Learning to Improve Sentiment Analysis in the Persian Language," 2023 9th International Conference on Web Research (ICWR), Tehran, Iran, Islamic Republic of, 2023, pp. 242-246, doi: 10.1109/ICWR57742.2023.10138965.

B. S, V. M.R.M, S. Hariharan, N. Bharathiraja, P. K and R. Marappan, "Sentiment Analysis Model using Text and Emoticons for Pharmaceutical & Healthcare Industries," 2022 2nd International Conference on Innovative Sustainable Computational Technologies (CISCT), Dehradun, India, 2022, pp. 1-4, doi: 10.1109/CISCT55310.2022.10046449.

P. Saxena and S. Sharma, "Systematic Literature Review for Sentiment Analysis Using Big Data Social Media Streams," 2022 5th International Conference on Contemporary Computing and Informatics (IC3I), Uttar Pradesh, India, 2022, pp. 2173-2178, doi: 10.1109/IC3I56241.2022.10073043.

M. S. Islam Shopnil, S. M. Mahedy Hasan, M. A. Yakin Srizon and M. F. Faruk, "Post-Pandemic Sentiment Analysis Based on Twitter Data Using Deep Learning," 2022 25th International Conference on Computer and Information Technology (ICCIT), Cox's Bazar, Bangladesh, 2022, pp. 704-709, doi: 10.1109/ICCIT57492.2022.10055507.

P. Bharti, V. Sagar and B. Wadhwa, "An Analysis on Sentiments Using Deep Learning Approaches," 2022 International Conference on Computational Intelligence and Sustainable Engineering Solutions (CISES), Greater Noida, India, 2022, pp. 355-360, doi: 10.1109/CISES54857.2022.9844318.

A. Kumar, A. Nashte, R. Porwal Amit and C. Choudhary, "Hypergraph Neural Networks with Attention-based Fusion for Multimodal Medical Data Integration and Analysis," 2023 Seventh International Conference on Image Information Processing (ICIIP), Solan, India, 2023, pp. 628-633, doi: 10.1109/ICIIP61524.2023.10537751.

M. Sindhuja, K. S. Nitin and K. S. Devi, "Twitter Sentiment Analysis using Enhanced TF-DIF Naive Bayes Classifier Approach," 2023 7th International Conference on Computing Methodologies and Communication (ICCMC), Erode, India, 2023, pp. 547-551, doi: 10.1109/ICCMC56507.2023.10084106.

N. Osman and I. M. Husien, "Comparison of Sentiment Analysis Techniques for Twitter Posts Classification," 2022 International Conference on Data Science and Intelligent Computing (ICDSIC), Karbala, Iraq, 2022, pp. 93-97, doi: 10.1109/ICDSIC56987.2022.10075895.

V. Vanthana and K. Kartheeban, "Estimation of Accuracy Level for Sentiment Analysis using Machine Learning and Deep Learning Models," 2022 International Conference on Automation, Computing and Renewable Systems (ICACRS), Pudukkottai, India, 2022, pp. 717-722, doi: 10.1109/ICACRS55517.2022.10029086.

S. S. G, V. D, J. Dias and S. Patra, "Twitter Data-based Sarcastic Sentiment Analysis using Deep Learning Framework," 2022 International Conference on Artificial Intelligence and Data Engineering (AIDE), Karkala, India, 2022, pp. 326-330, doi: 10.1109/AIDE57180.2022.10060759.

N. Vyas, N. N. Jadhav, V. V. Raje and C. Choudhary, "Distributed Deep Reinforcement Learning with Graph Neural Networks for Personalized Drug Interaction Analysis," 2023 Seventh International Conference on Image Information Processing (ICIIP), Solan, India, 2023, pp. 610-615, doi: 10.1109/ICIIP61524.2023.10537643.

B. Madhurika and D. N. Malleswari, "Implementation of Sentiment Analysis System using DLCNN based Opinion Mining," 2022 International Conference on Augmented Intelligence and Sustainable Systems (ICAISS), Trichy, India, 2022, pp. 7-10, doi: 10.1109/ICAISS55157.2022.10010990.

O. Matías-Cristóbal, J. Padilla-Caballero, R. Gonzales-Rivera, R. Benavente-Ayquipa, S. Pérez-Saavedra and F. Cardenas-Palomino, "Enhancing Sentiment Analysis In Text Of Social Media Texts Using Hybrid Deep Learning Model And Natural Language Processing," 2023 6th International Conference on Contemporary Computing and Informatics (IC3I), Gautam Buddha Nagar, India, 2023, pp. 1776-1780, doi: 10.1109/IC3I59117.2023.10397710.

S. Islam, M. Jahidul Islam, M. Mahadi Hasan, S. M. Shahnewaz Mahmud Ayon and S. Shabnam Hasan, “Bengali Social Media Post Sentiment Analysis using Deep Learning and BERT Model,” 2022 IEEE Symposium on Industrial Electronics & Applications (ISIEA), Langkawi Island, Malaysia, 2022, pp. 1-6, doi: 10.1109/ISIEA54517.2022.9873680.

C. Choudhary, L. S. Nagra, P. Das, J. Singh and S. S. Jamwal, “Optimized Ensemble Machine Learning Model for Chronic Kidney Disease Prediction,” 2023 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS), Greater Noida, India, 2023, pp. 292-297, doi: 10.1109/ICCCIS60361.2023.10425073.

Lakshmana Rao, C. Gupta and T. S. R. Kiran, “Airline Twitter Sentiment Classification using Deep Learning Fusion,” 2022 International Conference on Smart Generation Computing, Communication and Networking (SMART GENCON), Bangalore, India, 2022, pp. 1-4, doi: 10.1109/SMARTGENCON56628.2022.10084207.

P. Isaranontakul and W. Kreesuradej, “A Study of Using GPT-3 to Generate a Thai Sentiment Analysis of COVID-19 Tweets Dataset,” 2023 20th International Joint Conference on Computer Science and Software Engineering (JCSSE), Phitsanulok, Thailand, 2023, pp. 106-111, doi: 10.1109/JCSSE58229.2023.10201994.

M. Sharma and M. Parmar, “Sentiment Analysis of Twitter Dataset using Ensemble Classification Approach,” 2022 International Conference on Artificial Intelligence and Data Engineering (AIDE), Karkala, India, 2022, pp. 306-311, doi: 10.1109/AIDE57180.2022.10060212.

İ. Aygün, B. Kaya and M. Kaya, “Aspect Based Twitter Sentiment Analysis on Vaccination and Vaccine Types in COVID-19 Pandemic With Deep Learning,” in IEEE Journal of Biomedical and Health Informatics, vol. 26, no. 5, pp. 2360-2369, May 2022, doi: 10.1109/JBHI.2021.3133103.

C. Imvimol and P. Chongstitvatana, “Sentiment analysis of messages on Twitter related to COVID-19 using deep learning approach,” 2021 25th International Computer Science and Engineering Conference (ICSEC), Chiang Rai, Thailand, 2021, pp. 363-367, doi: 10.1109/ICSEC53205.2021.9684587.

C. Choudhary, N. Vyas and U. K. Lilhore, “An Optimized Sign Language Recognition Using Convolutional Neural Networks (CNNs) and Tensor-Flow,” 2023 3rd International Conference on Technological Advancements in Computational Sciences (ICTACS), Tashkent, Uzbekistan, 2023, pp. 896-901, doi: 10.1109/ICTACS59847.2023.10390276.