iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 9 **Issue:** X **Month of publication:** October 2021

DOI: https://doi.org/10.22214/ijraset.2021.38488

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 9 Issue X Oct 2021- Available at www.ijraset.com

Evaluation of Transformer Model for Conversational Chatbots Using Cosine Similarity Technique

Atharva Joshi[1], Aniket Kulkarni[2], Manas Addepalli[3]
[1, 2, 3]Final Year, Electronics and Communication Engineering, MIT World Peace University, Pune

Abstract: The use of chatbots has seen a significant rise in the past few years, these pieces of software are seen on various E-commerce platforms or banking platforms. These applications mostly involve the use of Rule-based chatbots, in this paper we have explored the possibility of using smart chatbots which can have a free-flowing conversation with the user. The training of these chatbots determines their ability to have a human-like conversation. The transformer model in this study has an encoder-decoder architecture, with a self-attention layer. The self-attention layer helps the model to have a context of the input, which gives it a reference of each word with respect to the other words in the sentence to the model. The model is being trained for various epochs and these outputs have been based against expected output and their similarity is calculated using Cosine Similarity Technique. The results from these tests were observed and presented in the paper.
Keywords: Chatbot, Transformer Model, Self -Attention Layer, Epochs, Cosine Similarity.

I. INTRODUCTION

Chatbots are artificial intelligence-based software which come under the cluster of Natural Language Processing. They are utilized for various applications where human intervention is not required. They help organizations to reduce time utilized by human experts on these mundane tasks. There is a significant development in the field of Natural Language Processing, the more commercially available chatbots are namely IBM Watson, Google assistant, Samsung Bixby and Amazon Alexa. Chatbots are of three types simple chatbots, these are called rule based chatbots and are task specific; Smart Chatbots, they are designed to simulate near human Interaction, it is possible to have free flowing conversation with such chatbots these are AI – enabled and are bit harder to implement; hybrid chatbots are a combination of smart and simple chatbots. In our project we have tried to implement a Smart Chatbot using transformers and a self-attention layer. The objective of this project is to find out the amount of training accuracy we get when we train the bot for various amounts of epochs. To implement the chatbot we have researched various methods including LSTM (Long Short-Term Memory) Networks [7], Seq2Seq models [5], but we found Transformer architecture to be fascinating and hence we decided to implement it. The self-attention technique has been implemented with Recurrent Neural Networks, but the paper Attention is all you need has proposed the idea of using them in conjunction with Transformer model.

The I section covers the basics of the transformer model architecture; it explores self-attention technique and how it is implemented in the model. Section II focuses on the dataset and gives information about the number of conversational exchanges in the dataset. Section III gives information on training of the model and presents the training results in a graphical format. Section IV focuses on evaluation of the model by measuring the similarity between the expected output and the output given out by the model, which is done using Cosine Similarity Technique [6]. The final section concludes the paper and presents our findings.

II. TRANSFORMER AND SELF -ATTENTION LAYER

Transformer model is a combination of an Encoder and a Decoder. When further the individual components are studied it is seen that the encoder is composed of a self-attention layer, Normalization Layer and a Feed Forward Neural Network. The Decoder is also composed of a self-attention layer along with normalization layer and feed forward neural network and Encoder Decoder Attention layer. The word vectors are given as an input to the encoder and positional encoding is performed on the vector. As these vectors follow a specific pattern that the model learns, it helps the model to determine the position of each word. Now coming to the self-attention layer, the concept on which it is based is the mechanism of relating different positions of a single sentence in order to get a more detailed representation. As humans we can understand the reference of a single word in a sentence with respect to other words in the same sentence, but that sense of comparison is a luxury to a machine learning model and hence we need to devise a method for the model to get that sense of understanding; here Self Attention concept comes to play.

The paper All you need is attention has put calculation of self-attention into six major steps where in the vector inputs are scored, which are then multiplied in dot product and normalized using softmax activation function to get a self-attention value for each word. This value helps the model further to get a comparative representation. Finally, the transformer includes a Linear layer and a softmax layer which normalizes the output. Fig:1 represents the Transformer model in a Graphical Format.

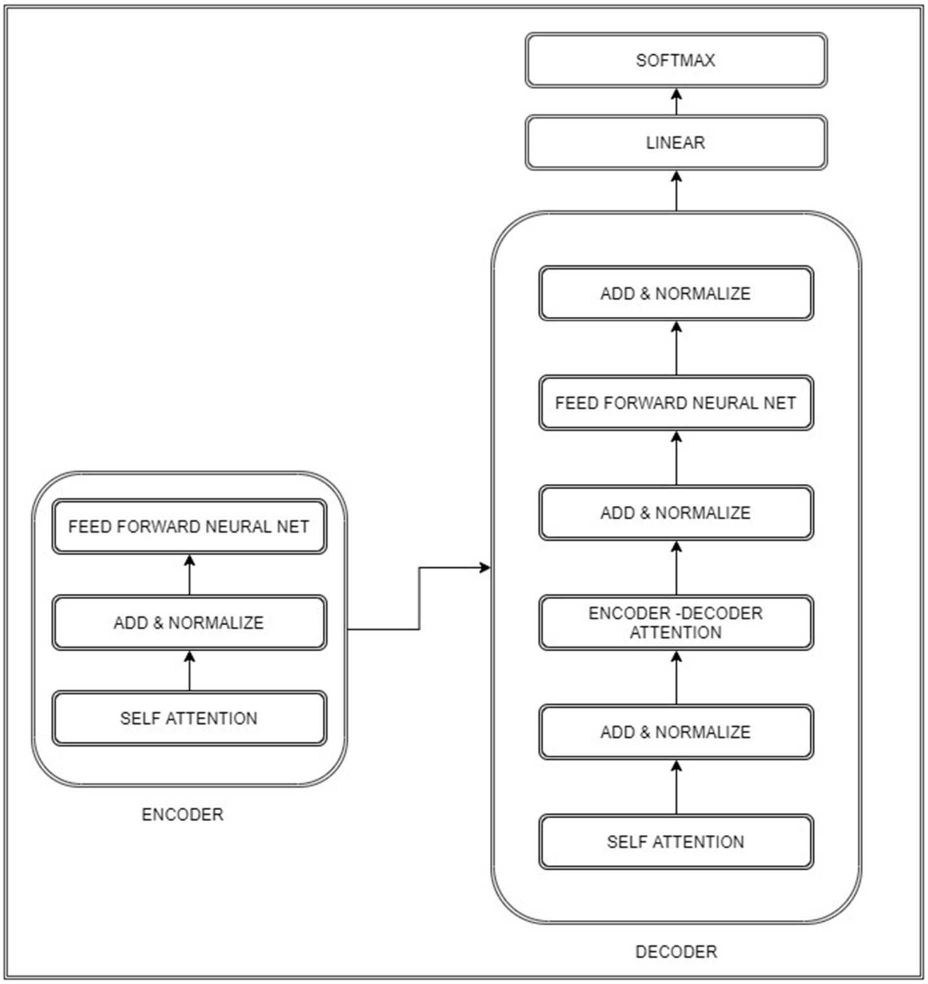


Fig 1: Block diagram of transformer

III.DATASET

The Dataset that has been obtained from Cornell Movie Dialog Corpus. This corpus is a large collection of fictional conversations extracted from raw movie scripts. The Dataset contains 220,579 conversational exchanges between 10,292 pairs of movie characters. After the Dataset has been downloaded and imported into the code it needs to be cleaned to remove the garbage values and informal words from the dataset. These operations are performed by the regular expression library.

IV.TRAINING

The model has been trained for various amounts of epochs which are 10, 25, 50, 100 ,250. The learning rate that we considered was adapted from the paper, Attention is all you need. Which is given below in equation 1.

$$l_{rate} = d_{model}^{-0.5} \cdot \min(step\_num^{-0.5}, step\_num \cdot warmup\_steps^{-1.5}) \text{ …. Equation 1 [1]}$$

This has been done to observe the training accuracy and Loss of the model. It also helps to put a perspective on how the model performs when all other hyperparameters are kept constant and brute force training is used to get a better accuracy on the model. The Average time taken for 1 Epoch to complete was around 90 seconds on Google Colab. Figure 2 and Figure 3 represent the graph of Epochs Plotted against Training loss and Accuracy respectively.

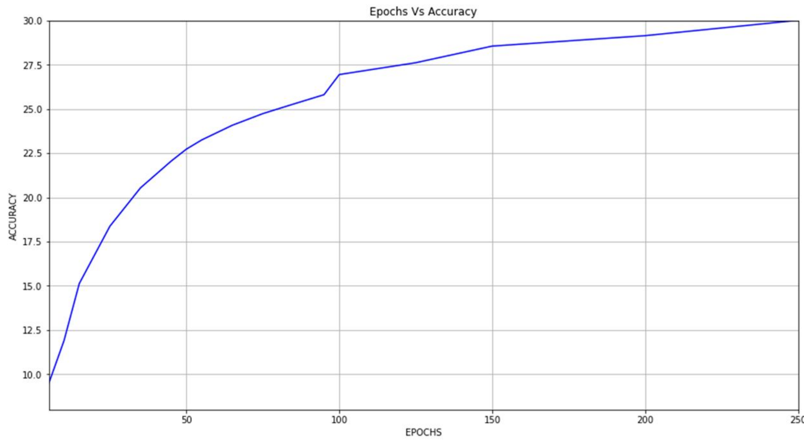


Fig 2: Epoch Vs Accuracy - Accuracy Graph

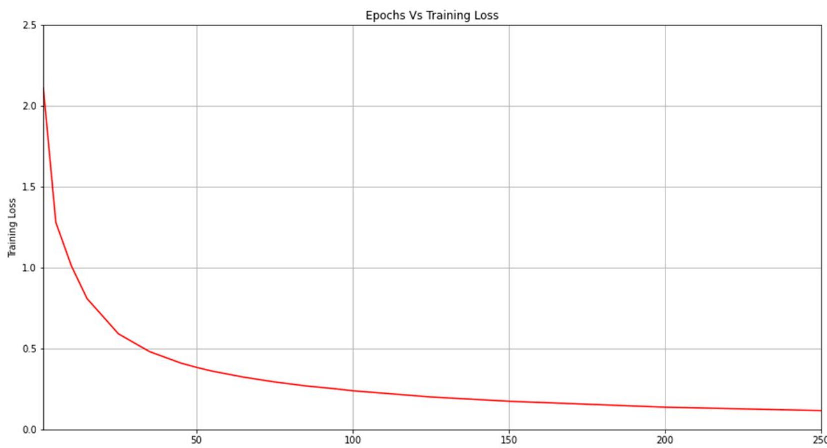


Fig 3: Epoch vs Training - Loss Graph

V. EVALUATION OF RESULTS

Evaluation of the trained model was done based on Cosine Similarity between the expected output and the output given by the models. A mean was taken and hence compared.

The expected outputs are those responses which humans answer and if the similarity between this and the output from the model achieves good similarity it can be assumed that the model is close to having human-like attributes and replies.

In the given table as the epochs are increased, mean cosine similarity increases. The model which has been trained for 10 epochs has shown least cosine similarity of 0.3698 and model which has been trained for 250 epochs shows a cosine similarity of 0.67733. The table below shows the comparison for each epoch on cosine similarity training loss and accuracy.

Table 1

Comparison for each epoch on cosine similarity training loss and accuracy.

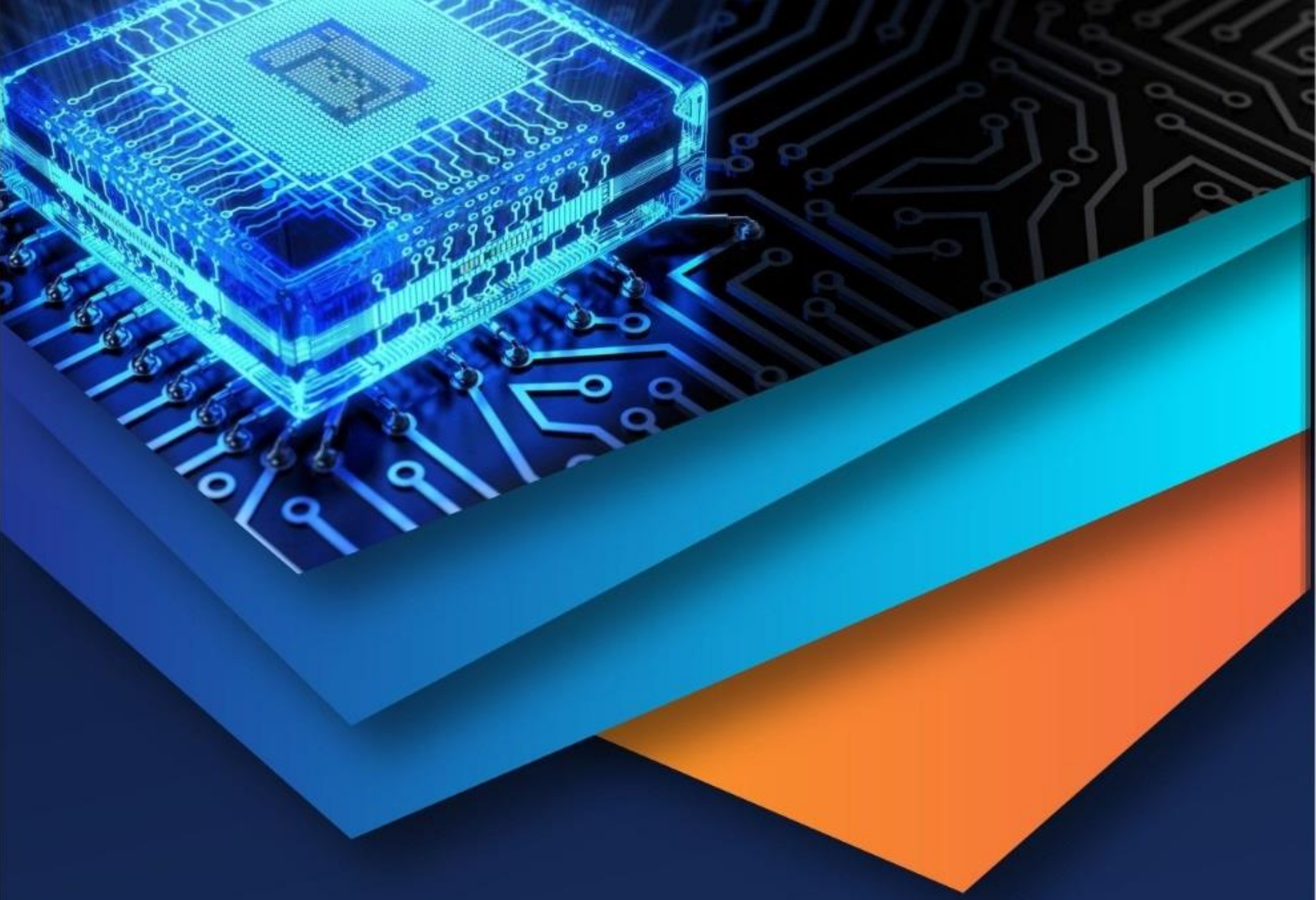
Epoch	Cosine Similarity 1	Cosine Similarity 2	Cosine Similarity 3	Mean Similarity	Training Loss	Accuracy
10	0.3235	0.3779	0.4082	0.3698	0.9452	0.1277
25	0.4371	0.2042	0.5303	0.3905	0.8763	0.136
50	0.4354	0.4577	0.3323	0.4084	0.3883	0.2251
100	0.5	0.3803	0.4842	0.4548	0.1897	0.2536
250	0.7489	0.6935	0.5896	0.6773	0.1152	0.2915

VI. CONCLUSION

In this work, we presented our study on the Transformer model with the self-attention layer, we have trained the model on the Cornell movie corpus for various number of epochs to get a vivid picture of how the training epochs affect the accuracy of the model by keeping other hyperparameters are kept constant. We have found that the model seems to have better accuracy when trained for about 250 epochs. After this point the accuracy seems to flatline and is seen to have much smaller increments. It is also seen that the mean cosine similarity for the model trained for 250 epochs is 67.73% and for that of 10 epochs is 36.98%.

REFERENCES

[1] Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł. & Polosukhin, I. (2017), Attention is all you need, in 'Advances in Neural Information Processing Systems', pp. 5998--6008 .
[2] Pearce, K., Zhan, T., Komanduri, A., & Zhan, J. (2021). A Comparative Study of Transformer-Based Language Models on Extractive Question Answering.
[3] Wang, S., & Jiang, J. (2017). Machine Comprehension Using Match-LSTM and Answer Pointer. ArXiv, abs/1608.07905.
[4] Yin, W., Ebert, S., & Schütze, H. (2016). Attention-Based Convolutional Neural Network for Machine Comprehension. ArXiv, abs/1602.04341.
[5] Sutskever, I., Vinyals, O., & Le, Q.V. (2014). Sequence to Sequence Learning with Neural Networks. NIPS.
[6] A. R. Lahitani, A. E. Permanasari and N. A. Setiawan, "Cosine similarity to determine similarity measure: Study case in online essay assessment," 2016 4th International Conference on Cyber and IT Service Management, 2016, pp. 1-6, doi: 10.1109/CITSM.2016.7577578.
[7] Hochreiter, Sepp & Schmidhuber, Jürgen. (1997). Long Short-term Memory. Neural computation. 9. 1735-80. 10.1162/neco.1997.9.8.1735.
[8] A. M. P. Braşoveanu and R. Andonie, "Visualizing Transformers for NLP: A Brief Survey," 2020 24th International Conference Information Visualisation (IV), 2020, pp. 270-279, doi: 10.1109/IV51561.2020.00051.
[9] V. A. Mandge and M. A. Thalor, "Revolutionize Cosine Answer Matching Technique for Question Answering System," 2021 International Conference on Emerging Smart Computing and Informatics (ESCI), 2021, pp. 335-339, doi: 10.1109/ESCI50559.2021.9396864.
[10] J. N. Madhuri and R. Ganesh Kumar, "Extractive Text Summarization Using Sentence Ranking," 2019 International Conference on Data Science and Communication (IconDSC), 2019, pp. 1-3, doi: 10.1109/IconDSC.2019.8817040.
[11] E. Kasthuri and S. Balaji, "A Chatbot for Changing Lifestyle in Education," 2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV), 2021, pp. 1317-1322, doi: 10.1109/ICICV50876.2021.9388633.
[12] W. Lu, Y. Duan and Y. Song, "Self-Attention-Based Convolutional Neural Networks for Sentence Classification," 2020 IEEE 6th International Conference on Computer and Communications (ICCC), 2020, pp. 2065-2069, doi: 10.1109/ICCC51575.2020.9345092.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY