iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 10 Issue: VII Month of publication: July 2022

DOI: https://doi.org/10.22214/ijraset.2022.45669

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue VII July 2022- Available at www.ijraset.com

Neural Machine Translation

Kodithala Rahul[1], Muthyam Kailash[2], Vuppugandla Manish Kumar[3], Mr. Ch. Vijaya Bhaskar[4]
[1, 2, 3]Department of Information Technology, Sreenidhi Institute of Science and Technology, Hyderabad, Telangana, India
[4]Assistant Professor,Dept of IT

Abstract: The project's novelty is not merely importing modules and preparing data and feeding the data to the model but understanding how the real language translation works and implementing the logics underlying each method utilized and creating every function from scratch, resulting in the creationof a Neural Machine Translation model. Initially, translation was accomplished by simply substituting words from one language for those from another. However, because languages are essentially different, a greater degree of knowledge (e.g., phrases/sentences) is required to achieveeffective results. With the introduction of deep learning, modern software now employs statisticaland neural techniques that have been shown to be more effective when translating. We are essentially translating German to English utilizing Sequence to Sequence models with attention and transformer models. Of course, everyone has access to Google Translates power, but if you want to learn how to implement translation in code, this project will show you how. We are writingour code from scratch, without using any libraries,in order to understand how each model works.While this design is a little out of date, it is still a great project to work on if you want to learn more about attention processes before moving on to Transformers. It is based on Effective Approaches to Attention-based Neural Machine Translator and is a sequence to sequence (seq2seq) model for German to English translation.
Keywords: Translation, Attention, sequence to sequence, Deep learning

I. INTRODUCTION

Translating German to English using seq2seq with attention and transformer models

1) seq2seq using cnn.
2) seq2seq using rnn.
3) Transformers.

The encoder in the Seq2seq model is a stacked bi-directional LSTM or GRU that generates a variable length output that is fed into the decoder, which is also a stacked bi-directional RNN/LSTM. They communicate by means of a gentle attention technique. In many methods, such asmachine translation and speech recognition, this architecture has long surpassed phrase-based models. CNN enables a hierarchical learning process in which adjacent input sequences interact in the lower layers and distantinput sequence interactions are learned as we travel downstream inthe layers. The encoder-decoder model is analogous to sequence2sequence learning.

II. LITERATURE SURVEY

Neural Machine Translation is based on translating phrase from one language to another language,in our proposed system, we achieved the neural machine translation using 3 models namely seq2seq using conventional neural networks, seq2seq using recurrent neural networks, by using transformers. We observed the metrics for this task using BLEU, we got the best BLEU score usingtransformers. Aswe saw the existing system with encoder decoder, we optimized the results usingtransformers model. We use feed forward network in transformers for getting better results. We also gone through few challenges like two languages may have different structures, words may not have an equivalent meaning in other language. Facing the challenges, we have achieved a good BLEU score using transformers model. We did not use any third-party libraries and defined every function including encoder and decoder functions.

III. PROBLEM STATEMENT

Neural Machine Translation is a field of computational linguistics, main focus of neural machine translation is to convert text from one language to another language based on some rules. Neural machine translation is derived from the principles of deep learning, neural machine translation hadbeenbest to complete the task of translation based on its algorithm. This algorithm is aproduct of deep learningwhich is tested for large number of datasets with many languages across the worldand proved to be effective across all the datasets.

With many years of proper research many variations of neural machine translation came out and some of them are being investigated also few of them are deployed and found to be effective in the industry.

One of the best versions of neural machine translation is transformer structure using neural networks.

IV. METHODOLOGY

Architecture

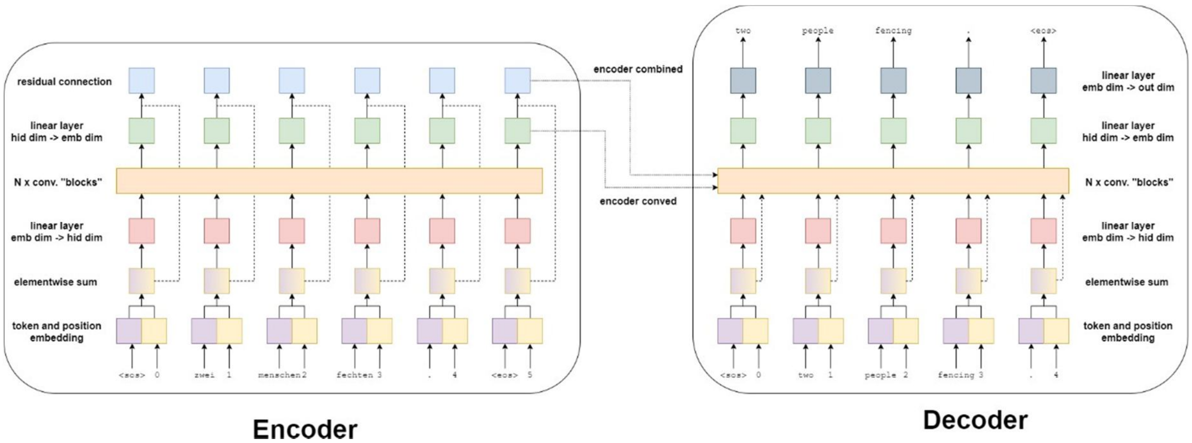

Figure 2: Shows the Architecture of the CNN Model

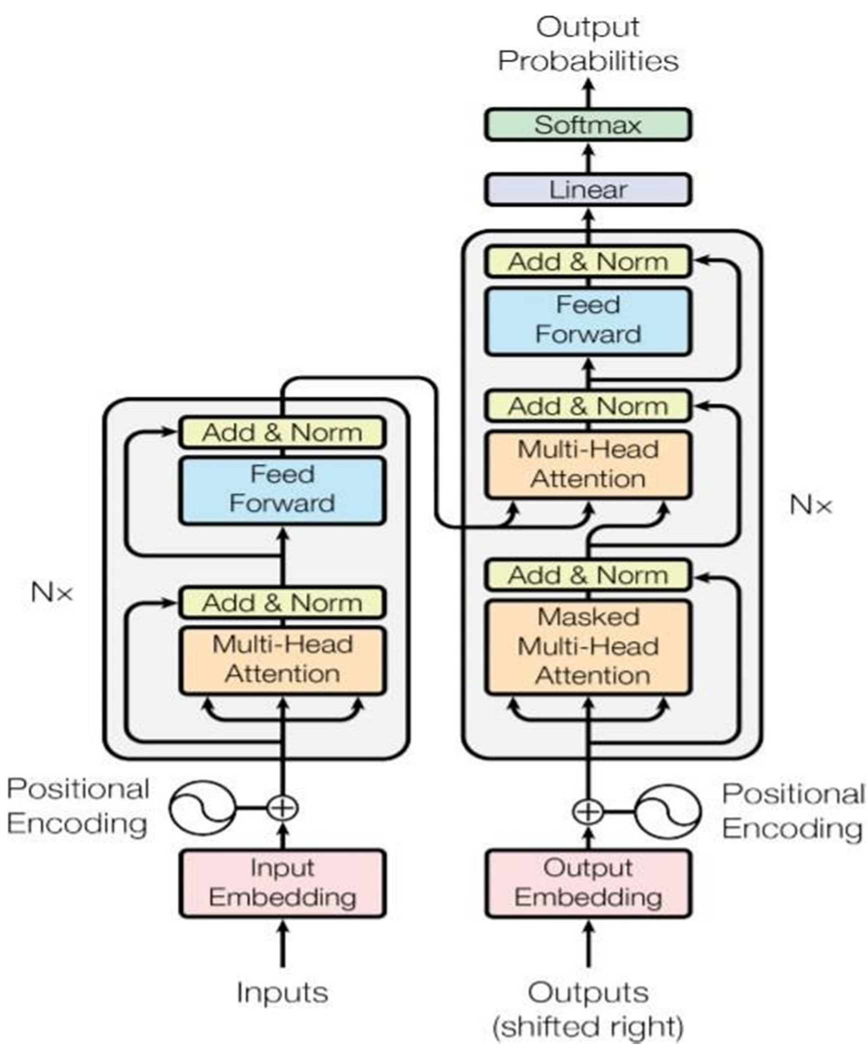

Figure 2: Shows the Architecture of the Transformer Model

V. IMPLEMENTATION

A. Importing the libraries

To begin, we must import a few libraries like as Pytorch, Matplotlib, Numpy, and Spacy so that we mayuse the built-in methods to make our model effective and avoid writing repetitive code.

B. Loading Cuda

Load CUDA, as the whole architecture is a transformers model built from scratch general CPU'scomputing power isn't enough thus CUDA – graphic processing unit.

C. Splitting train and test data

Traditional splitting of data into train and test which then are used for actual training and validations.

D. Defining Encoder and Decoder functions

Built transformer model consists of two parts an encoder and a decoder. Basically all encoder does isconvert the text based input into vectors and decoder then reforms the output text from the vectors.

E. Defining Seq2Seq neural network and feeding the encoder and decoder to the function

The model is separated into two sub-models: the encoder, which produces a fixed- length encoding of the input English sequence, and the decoder, which predicts the output sequence one letter at a time.

F. Defining the actual language translation model

All the magic happens between the vectors that are fed to the decoder from the encoder, technically here's where the actual language translation happens the this when converted back to the desired output format forms the traditional translation we understand.

G. Defining evaluation model and loop functions.

Every model needs to be evaluated and looped into training based on the evaluation results, it's what makes the model more accurate.

H. Actual Training

Finally a master method defined which simply uses all the above defined functions and iterates the trainingthrough discriminate time intervals called as epochs.

VI. CONCLUSION

In both research and practice, transformers had become the dominant approach to handling neural machine translations. This project covered the most commonly used NMT methods, including as encoding, decoding, data augmentation, interpretation, and assessment. Despite NMT's enormous success, there are still numerous issues to be resolved. The followingaresome key and difficult difficulties for NMT, Learning about NMT.

Despite several attempts to examine and interpret NMT, our knowledge of the phenomenonremains limited.

Understanding how and why NMT generates translation results is critical for identifying thebottleneck and flaws in NMT models.

VII. RESULTS

```python
text1 = "Ein Mann geht die Straße entlang, während er eine Tasse in der Hand hält"
og_translation1 = "a man is walking down the street while holding a cup"
```
Python

```python
pred_list = translate(net, text1, german, english, device)
pred_text = ""
for word in pred_list:
    pred_text += word + " "

print(pred_text)
print("")
```
Python

```
a man is walking down the street holding a cup . <eos>
```

Fig. German phrases translated to English phrases Using Transformers

```python
text1 = "Ein Mann geht die Straße entlang, während er eine Tasse in der Hand hält"
og_translation1 = "a man is walking down the street while holding a cup"
```
Python

```python
pred_list = translate(net, text1, german, english, device)
pred_text = ""
for word in pred_list:
    pred_text += word + " "

print(pred_text)
print("")
```
Python

```
a man walking down the street holding a cup of food . <eos>
```

Fig. German phrases translated to English phrases Using seq2seq RNN

```python
text1 = "Ein Mann geht die Straße entlang, während er eine Tasse in der Hand hält"
og_translation1 = "a man is walking down the street while holding a cup"
```
Python

```python
pred_list, attention = translate(text1, SRC, TRG, net, device)
pred_text = ""
for word in pred_list:
    pred_text += word + " "

print(pred_text)
print("")
```
Python

```
a man is walking down the street while holding a cup of art . <eos>
```

Fig. German phrases translated to English phrases Using seq2seq CNN

VIII. AKNOWLEDGEMENT

This research was made possible under the guidance, support, and motivation provided by our faculty, who have our esteem to pursue our interests in the field of image processing. We are thankful to Mr. Ch. Vijaya Bhaskar, Assistant Professor, Dept of IT, SNIST; and Dr. K. Vijaya Lakshmi, Professor, Dept of IT, SNIST; and Dr. Sunil Bhutada Professor & Head, Dept of IT, SNIST.

REFERENCES

[1] Wu, Yonghui and Schuster, Mike and Chen, Zhifeng and Le, Quoc V. and Norouzi, Mohammad and Macherey, Wolfgang and Krikun, Maxim and Cao, Yuan and Gao, Qin and Macherey, Klaus and Klingner, Jeff and Shah, Apurva and Johnson, Melvin and Liu, Xiaobingand Kaiser, Łukasz and Gouws, Stephan and Kato, Yoshikiyo and Kudo, Taku and Kazawa, Hideto and Stevens, Keith and Kurian, George and Patil, Nishant and Wang, Wei and Young, Cliff and Smith, Jason and Riesa, Jason and Rudnick, Alex and Vinyals, Oriol and Corrado, Greg and Hughes, Macduff and Dean, Jeffrey. 2016. Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation. Available: https://arxiv.org/abs/1609.08144

[2] Liqun Chen and Yizhe Zhang and Ruiyi Zhang and Chenyang Tao and Zhe Gan and Haichao Zhang and Bai Li and Dinghan Shen and Changyou Chen and Lawrence Carin. 2019.Improving Sequence-to-Sequence Learning via Optimal Transport. Available:https://arxiv.org/abs/1901.06283

[3] Wei Zou and Shujian Huang and Jun Xie and Xinyu Dai and Jiajun Chen. 2019, A Reinforced Generation of Adversarial Examples for Neural Machine Translation. Available: https://arxiv.org/abs/1911.03677

[4] Chulhee Yun and Srinadh Bhojanapalli and Ankit Singh Rawat and Sashank J. Reddi and Sanjiv Kumar. 2019. Are Transformers universal approximators of sequence-to-sequence functions?.Available:https://arxiv.org/abs/1912.10077

[5] Jinhua Zhu and Yingce Xia and Lijun Wu and Di He and Tao Qin and Wengang Zhou and Houqiang Li and Tie-Yan Liu. 2020. Incorporating BERT into Neural Machine Translation. Available: https://arxiv.org/abs/2002.06823

[6] Biao Zhang and Philip Williams and Ivan Titov and Rico Sennrich. 2020. Improving Massively Multilingual Neural Machine Translation and Zero-Shot Translation. Available:https://arxiv.org/abs/2004.11867

[7] Marion Weller-Di Marco and Alexander Fraser. 2020. Modeling Word Formation in English– German Neural Machine Translation. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 4227–4232, Online. Association for Computational Linguistics. Available: https://aclanthology.org/2020.acl-main.389/