Attention Is All You Need [Implementation Or Demo]

Aniket Pandey
Department of Computer Science
George Mason University
apandey7@gmu.edu

Adel Alkhamisy
Department of Computer Science
George Mason University
aalkhami@gmu.edu

1 Abstract

This report details the steps taken to replicate the Transformer model that Vaswani et al. described in their key paper Attention Is All You Need (2017). The report details the project's finished components, outstanding work, difficulties encountered, and early findings. The goal is to get BLEU scores for the English-to-German and English-to-French machine translation tasks that are equivalent to those published in the original study in order to validate the model's performance.

2 Introduction

The development of new and effective models like the Transformer has contributed to considerable developments in the fields of natural language processing (NLP) and machine learning in recent years. By relying purely on attention mechanisms and eschewing the requirement for recurrent and convolutional neural network-based models, the Transformer model, first presented by Vaswani et al., has transformed the NLP landscape. The model has proven to be more effective than humans at a variety of activities, including question answering, sentiment analysis, and machine translation.

Our project's main driving force is to replicate the findings from the original publication because doing so is crucial to ensuring that our implementation is accurate and to laying the foundation for further study. We intend to confirm the effectiveness of the Transformer model and gain important insights into the underlying mechanisms that contribute to its success by recreating the Transformer model and getting BLEU scores similar to those presented in the paper.

Also, replicating the outcomes of the initial study will provide us with a deeper comprehension of the model's advantages and disadvantages, allowing us to spot prospective upgrades and changes that might further improve its performance. In addition to benefiting the larger NLP research community, this information will make it easier to create future models that are stronger and more effective.

In this progress report, we describe our progress in implementing the Transformer model, focusing on the completed components, remaining tasks, and challenges faced throughout the process. We also discuss our plans for testing and training the model on various datasets, including English-to-German, English-to-French, and English-to-Hindi translation tasks. By sharing our progress and insights, we hope to contribute to the ongoing dialogue surrounding the Transformer model and its applications in the field of NLP.

3 Approach

3.1 Data Pipeline and Preprocessing

Any machine learning project's success is heavily reliant on how well-designed and effective the data pipeline and preparation methods are. For our project, we have built a solid data pipeline that efficiently manages the data needed for the training and testing of the Transformer model.

3.1.1 Dataset Acquisition and Preprocessing

For our machine translation tasks, the WMT 2014 dataset, which comprises of parallel English-German, English-French, and English-Hindi sentences, has been successfully acquired from https://nlp.stanford.edu/projects/nmt. To verify that the dataset is acceptable for training the model, popular preprocessing methods including tokenization, lowercasing, and cleaning have been used as suggested by Koehn et al.

3.1.2 Data Loading and Batching

We have created a data loading and batching mechanism that optimizes the process to enable effective and seamless data feeding into the model throughout training and inference. The system efficiently reads the dataset, organizes it into mini-batches, and feeds it into the model while ensuring that the data is appropriately shuffled and padded (Vaswani et al., 2017). The Transformer model's performance and computational effectiveness during training are dependent on this procedure.

In conclusion, our data pipeline and preprocessing methods are critical to the accomplishment of our mission. We have created a reliable and effective system that efficiently manages the data needed for training and testing the Transformer model by utilizing cutting-edge techniques and referencing the enormous literature in the field of NLP.

3.1.3 Byte-Pair Encoding

Managing the extensive vocabulary and the variety of word forms is one of the most difficult tasks involved in handling natural language data. The authors have adopted byte-pair encoding (BPE) Sennrich et al., a popular method for subword-level tokenization, to address this problem. BPE makes it possible to handle uncommon and non-vocabulary terms more effectively, which can have a substantial impact on the model's performance. By using BPE, we may represent words as collections of more manageable subword units, which reduces the quantity of the vocabulary and the computational difficulty.

3.2 Architecture

In contrast to conventional recurrent and convolutional neural network-based models, the Transformer model developed by Vaswani et al. offers a distinctive and unique architecture. It instead only relies on attention techniques to create interdependence between input and output tokens. In this section, we detail the progress made in implementing the key components of the Transformer architecture.

3.2.1 Multi-Head Attention

The multi-head attention mechanism serves as the fundamental structural component of the Transformer model. This element enables the model to focus on various locations in the input sequence, simultaneously capturing distinct characteristics of the input data (Vaswani et al., 2017). Our implementation of the multi-head attention mechanism closely follows the original work, allowing the model to analyze many attention heads in simultaneously and combine their results to generate the final attention output.

3.2.2 Encoder and Decoder

The encoder and decoder stacks, each with several levels, make up the Transformer architecture. While the decoder creates the output sequence based on the encoder's output and its own prior outputs, the encoder is in charge of processing the input sequence and creating a continuous representation (Vaswani et al., 2017). The encoder and decoder layers, as well as their sublayers including position-wise feedforward networks, encoder-decoder attention, and self-attention, have been developed effectively.

3.2.3 Position-wise Feedforward Networks

The input representation from the multi-head attention sublayer is transformed by the position-wise feedforward networks, which are a crucial part of the Transformer architecture suggested by Vaswani et al.. According to the original paper's description, we developed the position-wise feedforward networks using a two-layer fully connected feedforward network with a ReLU activation function and dropout.

3.2.4 Positional Encoding

The Transformer model uses positional encoding to incorporate the positioning information of tokens in the input sequence (Vaswani et al., 2017). This element is essential to the model's effectiveness since it enables it to accurately represent the relative positions of tokens within the sequence. To make sure that our model is able to interpret and comprehend the sequential nature of the input data, we are now working on implementing the positional encoding mechanism.

In conclusion, our progress in implementing the Transformer architecture has been significant, with most components already in place. We are confident that, upon completing the remaining tasks, we will be able to fully reproduce the original model and achieve comparable performance on our chosen machine translation tasks. By thoroughly understanding and implementing the Transformer architecture, we will be better equipped to explore potential improvements and

modifications that can further advance the state-of-the-art in natural language processing.

4 Testing and Training (Pending)

Our next steps involve finalizing the implementation of the Transformer model, including the positional encoding mechanism, and integrating all the components to form a complete and functional model. Once the implementation is complete, we will proceed to the training and testing phases of our project.

4.1 Training Setup

We will leverage the computational resources provided by GMU ORC `https://ondemand.orc.gmu.edu`, which grants us access to multiple GPUs for training our model. Our training strategy will involve dividing the training batches into smaller sub-batches to calculate gradients. Although this approach may increase the training time, it will not affect the fundamental calculations and is essential due to the absence of computations in the model design that require the entire training batch (Vaswani et al., 2017).

We will use the optimizer hyperparameters and regularization settings specified in the original paper to ensure a fair comparison of our implementation with the baseline results. Furthermore, we will employ techniques such as learning rate scheduling and gradient clipping to optimize the training process and enhance the model's performance.

4.2 Testing and Evaluation

Upon completing the training phase, we will test our implementation on the machine translation tasks of the WMT 2014 dataset, namely English-to-German, English-to-French, and English-to-Hindi. Our primary evaluation metric will be the BLEU score (Papineni et al., 2002), which will enable us to compare our model's performance with the baseline results reported by (Vaswani et al., 2017). Additionally, we will conduct an ablation study to understand the contributions of individual components of the Transformer model to its overall performance.

5 Preliminary Results

As our implementation is still in progress, we have not yet obtained any experimental results. However, we are confident that our carefully designed data pipeline, efficient training strategy, and faithful implementation of the Transformer architecture will enable us to achieve competitive results on our chosen machine translation tasks. Upon completing the implementation and training phases, we will present our findings, including the BLEU scores obtained on the WMT 2014 dataset, and compare them with the baseline results established by (Vaswani et al., 2017). Our analysis will also include insights gained from the ablation study, shedding light on potential avenues for future research and improvement of the Transformer model.

As we progress through the remaining tasks and obtain experimental results, we will continue to document our findings and update our progress report accordingly. Our final report will provide a comprehensive account of our project, including the challenges faced, the solutions employed, and the results achieved. We are eager to explore the potential of the Transformer model and contribute to the ongoing research in natural language processing and machine translation.

6 Challenges

As we progress through the implementation of the Transformer model, we have encountered several challenges. In this section, we discuss these challenges and the steps we have taken or plan to take to address them.

6.1 Data Pipeline and Preprocessing

Obtaining the WMT 2014 dataset and creating an efficient pipeline for data preprocessing, including byte-pair encoding, posed an initial challenge. We have addressed this issue by developing a robust and efficient data pipeline that handles the dataset and ensures a seamless flow of data into our model for training and inference.

6.2 Positional Encoding

Implementing the positional encoding mechanism is a critical aspect of the Transformer model, as it allows the model to capture the relative positions of tokens within the input sequence. We are currently working on completing the implementation of this component, and we expect to resolve this challenge shortly.

6.3 Understanding and Implementing the Transformer Architecture

Understanding the intricacies of the Transformer architecture, particularly the multi-head attention mechanism and the position-wise feedforward networks, was a challenge. We overcame this challenge through thorough analysis and discussion of the original paper, as well as by referring to other relevant literature on the Transformer model (Alammar, 2018).

6.4 Stochastic Reproducibility

Reproducing the results of the original paper in a stochastic manner is a challenge due to differences in random seeding and code structure, which could lead to different random sampling. Although we cannot entirely eliminate this issue, we will strive to achieve results that are as close as possible to the baseline established by Vaswani et al.

6.5 Library Differences

(Vaswani et al., 2017) used TensorFlow for their implementation, while we are using PyTorch. These differences in libraries for automatic differentiation and other computations might lead to discrepancies in the results. However, we have been meticulous in our implementation to minimize these discrepancies and ensure that our results are comparable to the baseline.

By addressing these challenges and remaining mindful of potential issues that may arise, we believe that our project will provide valuable insights into the Transformer model and its application to machine translation tasks.

7 Conclusion

In conclusion, this progress report outlines the ongoing efforts to replicate the Transformer model proposed by (Vaswani et al., 2017) for English-to-German, English-to-French, and English-to-Hindi machine translation tasks. The report covers the project's finished components, outstanding work, challenges encountered, and early findings. The main goal is to achieve BLEU scores similar to those published in the original study to validate the model's performance. The report highlights the progress made in implementing the data pipeline, preprocessing, and Transformer architecture. Additionally, it discusses the pending testing and training phases and the preliminary results expected once the implementation is complete. By overcoming the challenges and completing the remaining tasks, the project aims to contribute to the ongoing research in natural language processing and machine translation.

References

Jay Alammar. 2018. The illustrated transformer.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need.