Exploring Transformers Outside of NLP

Arsyi Syarief Aziz
Computer Science Study Program
Department of Mathematics
Universitas Hasanuddin

CONTENTS

LIST OF FIGURES

LIST OF TABLES

Exploring Transformers Outside of NLP

***Abstract*—The transformer is a robust neural network architecture that has surpassed the performance of RNNs, CNNs, and even ensemble models in machine translation. In addition to being performant in NLP, previous research has shown that transformers can also work in other domains. To further explore this capability, the transformer was tested in its ability to complete two tasks, the first was to reverse a sequence, and the second was to identify the anomaly image in an image set. The models performance on these tasks was assessed based on its accuracy in producing the expected output. The results of the experiments show that transformers are capable of completing both tasks well. In the first task, the model produced an accuracy of 100% in both its training set and test set, and in the second task, the model produced an accuracy of 96.41% and 94.41% in the training set and test set, respectively.**

I. Introduction

This report discusses about the capability of transformers outside of the natural language processing (NLP) domain.

The transformer in this report is defined as a robust neural network architecture that has been proven perform well in the domain of natural language processing. It has surpassed the performance of RNNs, CNNs, and even ensemble models in machine translation [1]. The high performance can be attributed to its use of attention mechanisms, which allows the model to be parallelized and learn long-term dependencies [1].

Not only are transformer performant in NLP, but they have also been shown to work in other domains, such as computer vision [2]. In order to explore the capability of this architecture, I have conducted experiments using transformers to complete tasks based on two different types of data, which include sequenced data and set data. These data types were specifically chosen because they can be expressed as a sequence; allowing the attention mechanism in transformers to be utilized.

More details about the steps of the experiments are elaborated in the following sections.

II. Problem Definition

The experiments conducted for this report include two problems, one to test the performance of the transformer in processing sequenced data and another to test its performance on set data.

In the first problem, the transformer was given a random sequence of numbers of length N. The task of the transformer was to reverse the sequence of data, such that a sequence that might originally be 123456789 is transformed into the expected output of the 987654321.

For the second problem, a set of N images was sampled from the CIFAR100 dataset. In each set of images, there were $N-1$ images of the same category and 1 image that is of a different category, which is called an anomaly. Based on the set of data mentioned previously, the task of the transformer was to correctly detect the anomaly image among the other $N-1$ images.

Fig. 1. A set of 10 images from the CIFAR100 dataset. Pictures one to nine (from the left) are images of trees, meanwhile, picture ten is an image of a volcano.

III. Proposed Solution

As mentioned previously, the problems were solved using transformers.

A transformer itself is a deep learning architecture that utilizes attention to understand the correlations in a sequence. The attention mechanism usually consists of four components. First is the query, which is a feature vector that describes what a model is looking for in a sequence. Second are the keys, which are feature vectors that roughly describe what an element is offering. Third are the values, which are feature vectors indicating the value of an element. Last is the score function, f_{attn}, which is a function that takes a query and key as input, and outputs a score. These four components are used to calculate a weighted average defined in the mathematical notation below.

$$\alpha_i = \frac{\exp\left(f_{attn}(key_i, query)\right)}{\sum_j \exp(\{_{attn}(key_i, query))},$$

$$out = \sum_i \alpha_i \cdot \text{value}_i.$$

The attention mechanism applied to transformers is called self-attention, or more specifically, scaled dot product attention. This mechanism utilizes the dot product as the score function and a scaling factor, d_k, to scale the score value. From this description, we can more specifically define mathematical formula for attention in transformers to the following equation.

$$\text{Attention}(Q, K, V) = \text{Softmax}(\frac{QK^T}{\sqrt{d_k}})V.$$

To further extend the attention mechanism, transformers allow multiple attentions to be used. This feature is called multi-head attention and it allows a model to attend to multiple aspects of sequence . It works by transforming a query, key and value into h sub-queries, sub-keys, and sub-values, which are then passed through h scaled dot product attentions, called heads. These heads are concatenated before being multiplied by a weight matrix to form multi-head attention.

$$Multihead(Q, K, V) = Concat(head_1, \ldots, head_h)W^O.$$

Based on the defined attention mechanism, transformers are built upon an encoder-decoder structure. The encoder part consists of N identical blocks arranged in a sequence, called encoder blocks. These encoder blocks consist of a single multi-head attention block, followed by a single feed-forward network block. The output of each of these blocks are added with their respective inputs through a residual connection and are normalized. For the the decoder part, the structure is similar, but it has an additional masked multi-head attention block that processes the output of an encoder. This additional block functions to prevent positions in the input sequence from attending to subsequent positions [1]. Fig. 2 best describes the architecture.

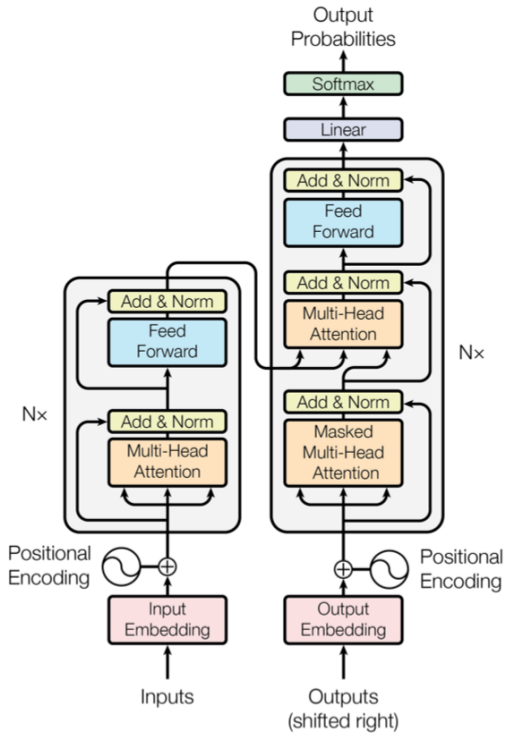


Fig. 2. The transformer architecture.

IV. Criteria for Assessing Solutions

The criteria used to assess the transformer's solution to a given problem is its accuracy in predicting the expected output. This accuracy metric is calculated slightly differently in the two experiments. In the sequence reversal experiment, accuracy is calculated as the percentage of predicted elements that are in the correct position. Meanwhile, in the set anomaly experiment, accuracy is calculated as the percentage of correctly identified anomalies.

V. Research Methodology

To test the performance of transformers in the two experiments, encoder-based transformer models were used. This was chosen to simplify the implementation of the architecture, and has been proven to work in some tasks [2] [3].

During training, the models used learning rate warm-up scheduling, which helps to improve the performance of transformers [4]. The specific learning rate warm-up method utilized in the models is called the cosine warm-up scheduler [1].

Because both experiments test two different data types with different properties, two different methodologies had to be used. To explain these methodologies, I will first explain about the sequence reversal experiment and then explain about the anomaly detection experiment.

In the sequence reversal task, the transformer model was tasked to reverse an input. However, due to the permutation-invariant property of transformers [5], the model cannot immediately differentiate positions in a sequence. For this reason, positional encoding was added to the input sequence before passing it to the transformer. The positional encoding consists of a sine and cosine function, defined in [1]. After positional encoding was added, the encoded sequences were passed through a transformer model with 32 hidden layers, 1 head, and 1 encoder block. Additionally, this model was trained with a learning rate of 0.0005, and a warm-up period of 50 epochs.

In the set anomaly task, the goal is to identify the anomaly image in a set. To do this, a CNN was used to extract high-level features of each image. After extracting these features, they were directly passed into the transformer model without positional encoding. The hyper-parameters of this model included 256 hidden layers, 4 heads, and 4 encoder blocks. On top of this, the model was trained with a learning rate to 0.0005, and a warm-up period of 100 epochs.

VI. Analysis and Interpretation

After conducting the experiments, I gathered the following results.

	Accuracy	
Experiment	Train	Test
Sequence Reversal	100%	100%
Anomaly Detection	96.35%	94.41%

TABLE I. Experiment results

Based on Table I, it appears that the transformer model was able to complete the two tasks with great accuracy. Specifically speaking, in the first task, the model correctly reversed 100% of the input sequences in both the train and test set. Meanwhile, for the anomaly detection task, the model was able to correctly identify 96.35% of the anomalies in the training set and 94.41% of the anomalies in the test set.

To understand how the model was able to achieve such high accuracy, I will show the attention maps of the model in both tasks.

First is the attention map for the sequence reversal task. In Appendix. A, we can see that the model determines the elements of the output sequence by attending to the element in the opposite index. For example, we can see that the element in index 0 attends to the element in index 15, the element in index 1 attends to an element in index 14, the element in index 2 attends to an element in index 13, and so on. This is similar to how us humans would usually approach this problem.

Next is the anomaly detection task. Fig. 3 shows an image set where the anomaly was identified correctly as the 10^{th} image, and Appendix. B shows its attention map. In the attention map, we can see that the multi-head attention layer focuses on the 10^{th} image in heads 1, 2, 3, and 4 of layer 2 and head 1 of layer 3. Meanwhile, in layer 4, all the heads completely ignore the 10^{th} image. This indicates that image 10 is the anomaly. In another example, Fig. 3 shows an image set where the model incorrectly identifies the 8^{th} image as the anomaly, and Appendix. C shows its attention map. From the attention map, we can see that the model's attention is pretty spread out, thus it hard to interpret. However, if we view the probabilities of each image, as seen in Table II, we can see that the model focuses on images 6, 8, and 10; image 10 being the anomaly. Despite focusing on the correct image, the model mistakenly predicts image 8 as the anomaly (due to it having the greatest probability). This error could be attributed to image 8 being the only image with a building in the background.

	Probabilities
Image 1	0.07%
Image 2	00.11%
Image 3	0.07%
Image 4	0.11%
Image 5	0.17%
Image 6	23.27%
Image 7	0.16%
Image 8	48.91%
Image 9	0.10 %
Image 10	27.04%

TABLE II. PROBABILITIES IN THE ANOMALY DETECTION TASK (INCORRECT IDENTIFICATION EXAMPLE)

Fig. 3. An image set of which the anomaly was correctly identified.

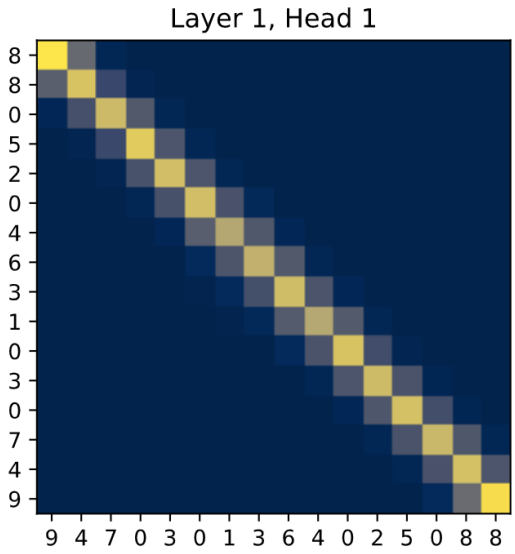
Fig. 4. An image set of which the anomaly was incorrectly identified.

VII. CONCLUSIONS AND RECOMMENDATIONS

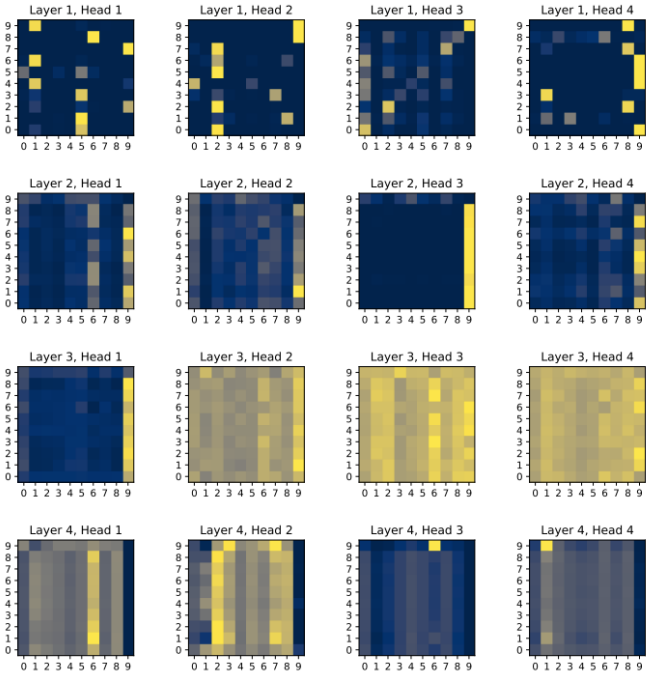
Based on the findings, it can be concluded that transformers perform well in both the sequence reversal task and anomaly detection task. This is further proof that transformers are capable in tasks outside of the NLP. Not only that, the findings have also shown that attention is capable of finding hidden correlations in data. By extending the attention mechanism to several heads, a transformer model is able to attend to several elements in a sequence, thus helping the model to better understand the hidden correlations.

Following this conclusion, I highly recommend using transformers in tasks related to sequence reversal and anomaly detection as they have demonstrated a high level of accuracy in my experiments.

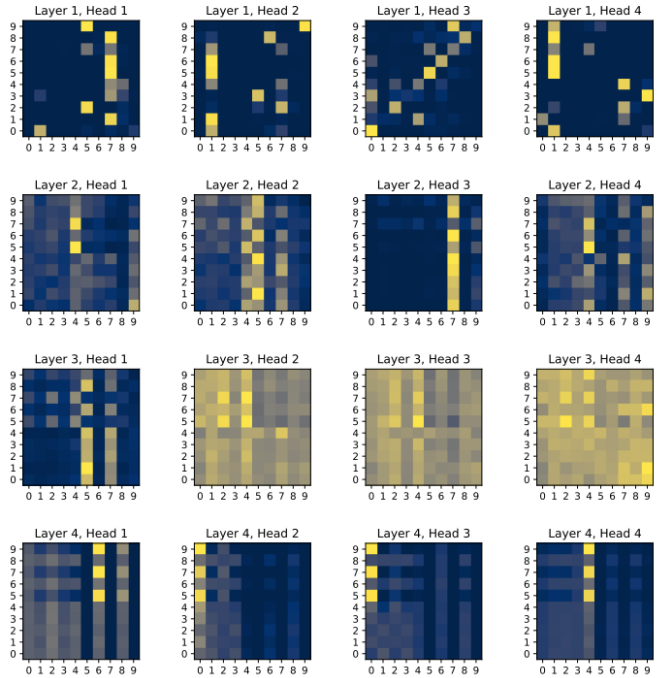
Appendix A
An example of an attention map in the sequence reversal task



Appendix B
An example of an attention map in the anomaly detection task (correct identification)



Appendix C
An example of an attention map in the anomaly detection task (incorrect identification)



References

[1] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I Polosukhin, "Attention is all you need", in Advances in Neural Information Processing Systems. 2017.

[2] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale", in the International Conference on Learning. 2021.

[3] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding", arXiv preprint arXiv:1810.04805. 2018.

[4] L. Liu, H. Jiang, P. He, W. Chen, X. Liu, J. Gao, and J. Han, "On the Variance of the Adaptive Learning Rate and Beyond", the International Conference on Learning. 2020.

[5] M. Naseer, K. Ranasinghe, S. Khan, M. Hayat, F. S. Khan, and M. Yang, "Intriguing Properties of Vision Transformers", in Neural Information Processing Systems. 2021.