

LORAS AND GSA-NETS FOR OCR

AMELIE SCHREIBER

Abstract. The following is a merging of two theories, that of LoRAs or "Low Rank Adaptations" introduced in [Hu et. al.], and that of group equivariant self-attention in transformers or other neural networks that include attention. We prove three things. First, LoRAs do not disrupt translation equivariance when included in translation equivariant attention with relative positional encodings. Second, LoRAs do not disrupt the lifting self-attention layers described in [RC]. And third, LoRAs do not disrupt equivariance of group self-attention layers as defined in [RC]. We also include a correction to the proof of equivariance of group self-attention by fixing the definition of the relative positional encodings defined in Proposition 5.2 and its proof in [RC].

Contents

1. Introduction	1
2. What is...a LoRA?	2
3. What is...Equivariance?	4
4. Including LoRAs Does Not Disrupt Translation Equivariance	5
5. LoRAs and Lifting Self-Attention	9
6. LoRAs and Group Self-Attention	11
7. Appendix	13
8. Conclusions	15
References	17

1. Introduction

Optical Character Recognition (OCR) has been a crucial technology in digitizing printed text, enabling us to convert scanned images of printed text into machine-encoded text. Traditional OCR techniques have relied on various methods, including feature extraction, segmentation, and pattern recognition, to detect and identify characters in an image. However, these methods have struggled with complex texts and poor-quality images, necessitating the development of more sophisticated models.

The current state-of-the-art in OCR technology on English is the Transformer OCR (TrOCR) model, introduced in [Li et. al.]. We also note the excellent performance of the CNN-LSTM based model in [MK], introduced for Hebrew written in Rashi script. This model leverages the strengths of both Vision Transformers (ViT) and large language models.

Date: May 24, 2023.

2020 Mathematics Subject Classification. Primary 68T07, 68T45, Secondary 68T50 68T10,

The ViT component is responsible for extracting features from images of text, which are then fed into a large language model for decoding into machine-encoded text. This combination enables TrOCR to handle a wide range of text images and has resulted in unprecedented performance in the OCR field.

In this paper, we propose a novel approach to further improve the performance and efficiency of OCR models using Low-Rank Adaptation (LoRA). LoRA is a method that significantly reduces the cost of fine-tuning large models. Instead of retraining all model parameters, which is computationally expensive, LoRA freezes the pre-trained model weights and injects trainable rank decomposition matrices into each layer of the architecture. This approach can reduce the number of trainable parameters by several orders of magnitude, making the fine-tuning process much more efficient.

LoRA can be applied to both the language model and vision parts of the TrOCR model, making it a versatile tool for model fine-tuning. Importantly, LoRA prevents catastrophic forgetting, a phenomenon where a deep learning model quickly forgets previously learned information when new data is introduced. This feature of LoRA is particularly valuable for large models that are trained on a diverse set of tasks. Furthermore, LoRA modularizes large models, making it easier to adapt them to multiple downstream tasks, such as transcribing and translating text in different languages and from different time periods.

Our approach also incorporates the concept of equivariant neural networks. Equivariance in a neural network refers to the property where the output of the network changes in the same way as the input. For instance, if the input image is rotated, the output feature map also rotates in the same way. This property is beneficial for handling variations in data, such as different orientations and positions of text in OCR tasks.

We propose the use of group self-attention networks (GSA-Nets), a type of equivariant transformer neural network, in place of the vision transformer part of the TrOCR model. GSA-Nets are designed to be equivariant to arbitrary symmetry groups, improving the model’s ability to capture semantic and syntactic information in images of text. The equivariance property of GSA-Nets enhances generalizability, allowing the model to better handle unseen variations in the data.

Furthermore, GSA-Nets improve parameter efficiency, meaning that they can achieve comparable or better performance than non-equivariant models with fewer parameters. This efficiency can lead to faster training times and less computational resources needed. Moreover, since GSA-Nets are naturally robust to transformations such as rotations and translations, they require significantly less data augmentation, a commonly used technique in training deep learning models that involves artificially expanding the training dataset by creating modified versions of existing data.

In this paper, we propose a novel approach that integrates LoRA and GSA-Nets to enhance the performance and efficiency of the TrOCR model. We hypothesize that by using GSA-Nets in place of the vision part of the TrOCR model and LoRAs for fine-tuning we can achieve better, state-of-the-art results.

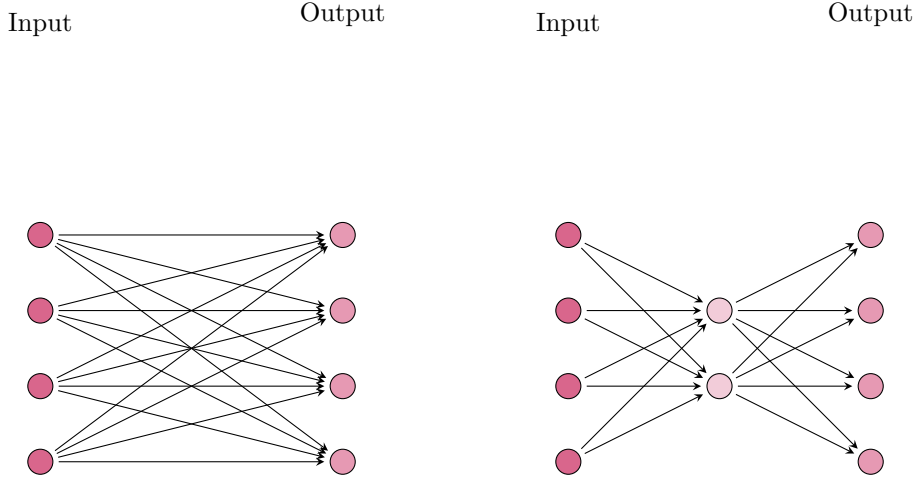
2. What is...a LoRA?

In the realm of deep learning, the concept of Low Rank Adaptations (LoRAs) was first introduced by [Hu et. al.]. These LoRAs provide an efficient alternative to the traditional fine-tuning of neural networks. The process begins by freezing the pre-existing weights of a layer in the neural network. For instance, in the context of a transformer’s attention mechanism, this could involve freezing the weights of the query, key, or value matrices—often represented as W_Q , W_K , and W_V .

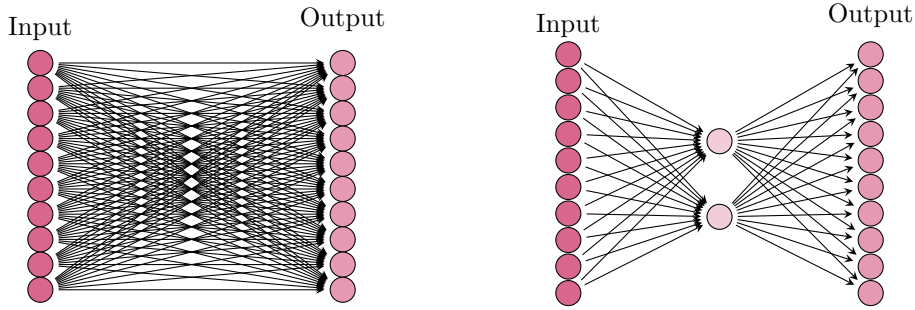
Following this, a LoRA layer is introduced to one or more of these pre-trained weight matrices. If we consider W to be a frozen weight matrix, the LoRA layer would take the form of $W + \Delta W$, wherein $\Delta W = BA$ constitutes the LoRA. Typically, these are low-rank decompositions, with $A \in \mathbb{R}^{r \times d_{in}}$ and $B \in \mathbb{R}^{d_{out} \times r}$, where the original weight matrix is $W \in \mathbb{R}^{d_{out} \times d_{in}}$. It is common for r to be significantly less than $\min\{d_{in}, d_{out}\}$.

The application of LoRAs only provides significant benefits when r is much smaller than the input and output dimension. Nevertheless, we can still opt for a smaller r and implement a LoRA in lieu of conventional fine-tuning. Empirical evidence suggests that in many cases, selecting $r = 4$ or $r = 8$ is sufficient—even for large weight matrices such as the query, key, and value matrices of a transformer’s attention mechanism.

Let’s now explore a scenario where the application of a LoRA does not yield any substantial benefits in terms of reducing the number of parameters:



Here, we see that the number of parameters for the LoRA layer $\Delta W = BA$ is the same as the original layer W , where we have $4 \times 2 \times 2 = 16$ parameters for the LoRA (on the right), and $4 \times 4 = 16$ parameters for the original frozen weight matrix (on the left). Next, let’s look at an example that gives us 40% the parameters of the frozen weight matrix:



Here we see the original (frozen) weight matrix has 10^2 parameters, and the LoRA has only $10 \times 2 \times 2 = 40$ parameters. In most cases, we have that the rank (ideally this is the number of neurons in the middle layer of the LoRA) of the frozen matrix is much smaller

than the input and output dimensions, and there is in fact a drastic reduction in parameter count. As an example, we might have an input and output dimension of say 100, in which case the weight matrix has $100^2 = 10,000$ parameters. However, the rank of this matrix is very often much lower than 100. In practice, it was shown that choosing $r = 4$ for the query, key, and value matrices is often more than sufficient for a LoRA as the middle dimension. In this case, we would get $100 \times 4 \times 2 = 800$ parameters in the LoRA, which is less than one tenth the original parameter count. Once we have such a LoRA in place, we can train it on some downstream task, and then add the LoRA weight matrix BA to the original (frozen) weight matrix W to obtain a model that performs well on this new task. Now, let us look at how LoRAs can be added into a translation equivariant attention mechanism without any loss of translation equivariance.

3. What is...Equivariance?

What is equivariance in the context of deep learning, and why is it useful and important? Equivariance, in the context of machine learning and specifically vision transformers, refers to a property of a function (in this case, the transformer) where the form of the output changes in the same way the input does. Formally, for a group G , we say that a function between two vector spaces $f : V_1 \rightarrow V_2$ is equivariant if

$$f(\rho_1(g)x) = \rho_2(g)f(x), \quad \forall g \in G$$

To understand this, consider the following scenario: Imagine you have an image, and you apply a transformation to it, like rotation or translation. An equivariant function, when applied to this transformed image, will produce an output that is the same as if you had applied the transformation to the output of the function on the original image. In other words, the function "respects" the transformation and applies it equivalently to its outputs.

This concept is important in vision transformers because it's crucial for these models to understand images regardless of how they're transformed. For instance, a picture of a cat remains a picture of a cat no matter how it's rotated or translated. So, a vision transformer that is equivariant to these transformations can recognize a cat in any orientation.

The concept of equivariance in vision transformers comes from the broader field of group theory in mathematics, where a function is said to be equivariant with respect to a group if it "respects" the group's operations. In this context, the group would be the set of all possible image transformations, and the operations would be the ways those transformations can be combined (e.g., performing two rotations in sequence).

However, it's important to note that typical Vision Transformers (ViTs) are not naturally equivariant, unlike some other types of models like Convolutional Neural Networks (CNNs) that exhibit translational equivariance due to their design. However, they often exhibit learned equivariance and can be more equivariant than CNNs once trained. Designing transformers to be equivariant to certain transformations is an active area of research. For instance, researchers have proposed various methods to make transformers equivariant or invariant (where the output doesn't change despite transformations to the input) to different types of transformations.

4. Including LoRAs Does Not Disrupt Translation Equivariance

In this section we explore the intersection of two major developments in transformer architectures - Low-Rank Adaptations (LoRAs) and Group Self-Attention Networks (GSA-Nets). Both of these innovations aim to enhance the efficiency and effectiveness of transformer models, but they each approach the problem from different angles. While LoRAs focus on reducing the computational and storage demands of fine-tuning large-scale models, GSA-Nets aim to improve model performance by incorporating symmetry groups into self-attention mechanisms.

The introduction of LoRAs, which freeze pre-trained model weights and inject trainable rank decomposition matrices into each layer of the Transformer architecture, greatly reduces the number of trainable parameters for downstream tasks. However, the question arises: does the inclusion of LoRAs affect the translation equivariance of GSA-Nets? Translation equivariance is a key property of neural networks, which describes the fact that if a pattern is translated, its numerical descriptors are also translated but not modified. This property is central to the functioning of GSA-Nets.

This section thus delves into the compatibility of these two advancements in transformer architecture. It seeks to examine whether the inclusion of LoRAs disrupts the translation equivariance property that is crucial to the performance of GSA-Nets. By exploring this question, we can gain a deeper understanding of how different innovations in transformer architecture can be effectively combined, allowing us to develop more efficient and powerful deep learning models.

In the following proof, we follow notation very similar to [RC]. Suppose we are given an input matrix $X \in \mathbb{R}^{d_{in} \times d}$, with columns representing the embedding vectors of d tokens. We can formulate the self-attention matrix in a transformer with relative positional encoding as

$$(1) \quad A_{i,j} = (W_Q X)(W_K(X + P_{x(j)-x(i)}))^T$$

We leave out the scaling factor $1/\sqrt{d_k}$ for simplicity, but it can easily be included without disrupting any of our arguments or proofs. It will be more beneficial for our purposes to formulate self-attention in a function theoretic way. In particular, we can view the input matrix X as a vector valued function $f : S \rightarrow \mathbb{R}^{d_{in}}$, that is $f \in L_{\mathbb{R}^{d_{in}}}(S)$, for the index set $S = \{1, 2, \dots, d\}$. We then view the query and key matrices as maps $\varphi_{qry} : L_{\mathbb{R}^{d_{in}}}(S) \rightarrow L_{\mathbb{R}^{d_k}}(S)$ and $\varphi_{key} : L_{\mathbb{R}^{d_{in}}}(S) \rightarrow L_{\mathbb{R}^{d_k}}(S)$. There is also a value function $\varphi_{val} : L_{\mathbb{R}^{d_{in}}}(S) \rightarrow L_{\mathbb{R}^{d_v}}(S)$. With these in hand, we can express the attention map with positional encoding as

$$(2) \quad A_{i,j} = \alpha[f](i, j) = \langle \varphi_{qry}(f(i)), \varphi_{key}(f(j) + \rho(i, j)) \rangle$$

Here, we have written $\rho(i, j)$ for the positional encoding. The map $\alpha[f] : S \times S \rightarrow \mathbb{R}$ maps pairs of elements $i, j \in S$ to the attention score of j relative to i . We can then write the attention mechanism as

$$(3) \quad \zeta[f](i) = \sum_{j \in S} \sigma_j(\alpha[f](i, j)) \varphi_{val}(f(j))$$

Next, we would like to include a LoRA for the query, key, and value maps. We will formulate this as

$$(4) \quad \Delta \varphi_{qry}(f(i)) = (\varphi_{qry}^A \circ \varphi_{qry}^B)(f(i)) = \varphi_{qry}^B(\varphi_{qry}^A(f(i)))$$

$$(5) \quad \Delta\varphi_{key}(f(i)) = (\varphi_{key}^A \circ \varphi_{key}^B)(f(i)) = \varphi_{key}^B(\varphi_{key}^A(f(i)))$$

$$(6) \quad \Delta\varphi_{val}(f(i)) = (\varphi_{val}^A \circ \varphi_{val}^B)(f(i)) = \varphi_{val}^B(\varphi_{val}^A(f(i)))$$

Here, we have

$$(7) \quad \varphi_{qry}^A : L_{\mathbb{R}^{d_k}}(S) \rightarrow L_{\mathbb{R}^{r(A)}}(S) \quad \varphi_{qry}^B : L_{\mathbb{R}^{r(A)}}(S) \rightarrow L_{\mathbb{R}^{d_k}}(S)$$

$$(8) \quad \varphi_{key}^A : L_{\mathbb{R}^{d_k}}(S) \rightarrow L_{\mathbb{R}^{r(A)}}(S) \quad \varphi_{key}^B : L_{\mathbb{R}^{r(A)}}(S) \rightarrow L_{\mathbb{R}^{d_k}}(S)$$

$$(9) \quad \varphi_{val}^A : L_{\mathbb{R}^{d_v}}(S) \rightarrow L_{\mathbb{R}^{r(A)}}(S) \quad \varphi_{val}^B : L_{\mathbb{R}^{r(A)}}(S) \rightarrow L_{\mathbb{R}^{d_v}}(S)$$

Next, including this in the attention mechanism, we get

$$(10) \quad \alpha^{LoRA}[f](i, j) = \langle \varphi_{qry}(f(i)) + \Delta\varphi_{qry}(f(i)), \varphi_{key}(f(j) + \rho(i, j)) + \Delta\varphi_{key}(f(j) + \rho(i, j)) \rangle$$

and then

$$(11) \quad \zeta^{LoRA}[f](i) = \sum_{j \in S} \sigma_j \left(\alpha^{LoRA}[f](i, j) (\varphi_{val}(f(j)) + \Delta\varphi_{val}(f(j))) \right)$$

Now, in order to have translation equivariance of a LoRA multihead self-attention with relative positional encoding, we need the following equation to hold,

$$(12) \quad m_{LoRA}^r[L_y[f], \rho](i) = L_y[m_{LoRA}^r[f, \rho]](i)$$

where $L_y[f](i) = f(x^{-1}(x(i) - y))$. The LoRA multihead self-attention with relative positional encodings on $L_y[f]$ is given by

$$\begin{aligned}
& m_{LoRA}^r[L_y[f], \rho](i) \\
&= \varphi_{out} \left(\bigcup_{head \in [H]} \sum_{j \in N(i)} \sigma_j \left(\left\langle \varphi_{qry}^{head}(L_y[f](i)) + \Delta \varphi_{qry}^{head}(f(i)), \right. \right. \right. \\
&\quad \left. \left. \varphi_{key}^{head}(L_y[f](j) + \rho(i, j)) + \Delta \varphi_{key}^{head}(f(j) + \rho(i, j)) \right\rangle \right) \left(\varphi_{val}^{head}(L_y[f](j)) + \Delta \varphi_{val}^{head}(f(j)) \right) \Big) \\
&= \varphi_{out} \left(\bigcup_{head \in [H]} \sum_{j \in N(i)} \sigma_j \left(\left\langle \varphi_{qry}^{head}(f(x^{-1}(x(i) - y))) + \Delta \varphi_{qry}^{head}(f(x^{-1}(x(i) - y))), \right. \right. \right. \\
&\quad \left. \left. \varphi_{key}^{head}(f(x^{-1}(x(j) - y)) + \rho(i, j)) + \Delta \varphi_{key}^{head}(f(x^{-1}(x(j) - y)) + \rho(i, j)) \right\rangle \right) \left(\varphi_{val}^{head}(f(x^{-1}(x(j) - y))) + \Delta \varphi_{val}^{head}(f(x^{-1}(x(j) - y))) \right) \Big) \\
&= \varphi_{out} \left(\bigcup_{head \in [H]} \sum_{x^{-1}(x(\bar{j})+y) \in N(x^{-1}(x(\bar{i})+y))} \sigma_{x^{-1}(x(\bar{j})+y)} \left(\left\langle \varphi_{qry}^{head}(f(\bar{i})) + \Delta \varphi_{qry}^{head}(f(\bar{i})), \right. \right. \right. \\
&\quad \left. \left. \varphi_{key}^{head}(f(\bar{j}) + \rho(x^{-1}(x(\bar{i}) + y), x^{-1}(x(\bar{j}) + y))) + \Delta \varphi_{key}^{head}(f(\bar{j}) + \rho(x^{-1}(x(\bar{i}) + y), x^{-1}(x(\bar{j}) + y))) \right\rangle \right) \left(\varphi_{val}^{head}(f(\bar{j})) + \Delta \varphi_{val}^{head}(f(\bar{j})) \right) \Big) \\
&\quad \times \left(\varphi_{val}^{head}(f(\bar{j})) + \Delta \varphi_{val}^{head}(f(\bar{j})) \right) \Big)
\end{aligned}$$

Here we have used the substitution $\bar{i} = x^{-1}(x(i) - y) \implies i = x^{-1}(x(\bar{i}) + y)$ and $\bar{j} = x^{-1}(x(j) - y) \implies j = x^{-1}(x(\bar{j}) + y)$. We can further reduce the equations using the definition of $\rho(i, j) = \rho^P(x(j) - x(i))$:

$$\begin{aligned}
&= \varphi_{out} \left(\bigcup_{head \in [H]} \sum_{x^{-1}(x(\bar{j})+y) \in N(x^{-1}(x(\bar{i})+y))} \sigma_{x^{-1}(x(\bar{j})+y)} \left(\left\langle \varphi_{qry}^{head}(f(\bar{i})) + \Delta \varphi_{qry}^{head}(f(\bar{i})), \right. \right. \right. \\
&\quad \left. \left. \varphi_{key}^{head}(f(\bar{j}) + \rho^P(x(\bar{j}) + y - (x(\bar{i}) + y))) + \Delta \varphi_{key}^{head}(f(\bar{j}) + \rho^P(x(\bar{j}) + y - (x(\bar{i}) + y))) \right\rangle \right) \left(\varphi_{val}^{head}(f(\bar{j})) + \Delta \varphi_{val}^{head}(f(\bar{j})) \right) \Big) \\
&= \varphi_{out} \left(\bigcup_{head \in [H]} \sum_{x^{-1}(x(\bar{j})+y) \in N(x^{-1}(x(\bar{i})+y))} \sigma_{x^{-1}(x(\bar{j})+y)} \left(\left\langle \varphi_{qry}^{head}(f(\bar{i})) + \Delta \varphi_{qry}^{head}(f(\bar{i})), \right. \right. \right. \\
&\quad \left. \left. \varphi_{key}^{head}(f(\bar{j}) + \rho^P(x(\bar{j}) - x(\bar{i}))) + \Delta \varphi_{key}^{head}(f(\bar{j}) + \rho^P(x(\bar{j}) - x(\bar{i}))) \right\rangle \right) \left(\varphi_{val}^{head}(f(\bar{j})) + \Delta \varphi_{val}^{head}(f(\bar{j})) \right) \Big) \\
&= \varphi_{out} \left(\bigcup_{head \in [H]} \sum_{x^{-1}(x(\bar{j})+y) \in N(x^{-1}(x(\bar{i})+y))} \sigma_{x^{-1}(x(\bar{j})+y)} \left(\left\langle \varphi_{qry}^{head}(f(\bar{i})) + \Delta \varphi_{qry}^{head}(f(\bar{i})), \right. \right. \right. \\
&\quad \left. \left. \varphi_{key}^{head}(f(\bar{j}) + \rho(\bar{i}, \bar{j})) + \Delta \varphi_{key}^{head}(f(\bar{j}) + \rho(\bar{i}, \bar{j})) \right\rangle \right) \left(\varphi_{val}^{head}(f(\bar{j})) + \Delta \varphi_{val}^{head}(f(\bar{j})) \right) \Big)
\end{aligned}$$

For any translation $y \in \mathbb{R}^{d_{in}}$, where d_{in} is the dimension of $f(i)$ and $f(j)$, the summation remains the same, so we have:

$$(13) \quad \sum_{x^{-1}(x(\bar{j})+y) \in N(x^{-1}(x(\bar{i})+y))} [\bullet] = \sum_{x^{-1}(x(\bar{j})) \in N(x^{-1}(x(\bar{i})))} [\bullet] = \sum_{\bar{j} \in N(\bar{i})} [\bullet]$$

As we can see, $m_{LoRA}^r[L_y[f], \rho](i) = L_y[m_{LoRA}^r[f, \rho]](i)$. We can thus conclude that relative positional encodings, coupled with LoRAs for the query, key, and value weight matrices gives a translation equivariant multihead self-attention mechanism. In particular, addition of LoRAs in a translation equivariant model with relative positional encodings does not disrupt the translation equivariance, which means we can further reduce the expression as

$$\begin{aligned} & m_{LoRA}^r[L_y[f], \rho](i) \\ &= \varphi_{out} \left(\bigcup_{head \in [H]} \sum_{x^{-1}(x(\bar{j})+y) \in N(x^{-1}(x(\bar{i})+y))} \sigma_{x^{-1}(x(\bar{j})+y)} \left(\left\langle \varphi_{qry}^{head}(f(\bar{i})) + \Delta \varphi_{qry}^{head}(f(\bar{i})), \right. \right. \right. \\ & \quad \left. \left. \left. \varphi_{key}^{head}(f(\bar{j}) + \rho(\bar{i}, \bar{j})) + \Delta \varphi_{key}^{head}(f(\bar{j}) + \rho(\bar{i}, \bar{j})) \right\rangle \left(\varphi_{val}^{head}(f(\bar{j})) + \Delta \varphi_{val}^{head}(f(\bar{j})) \right) \right) \right) \\ &= m_{LoRA}^r[f, \rho](\bar{i}) \\ &= m_{LoRA}^r[f, \rho](x^{-1}(x(i) - y)) \\ &= L_y[m_{LoRA}^r[f, \rho]](i) \end{aligned}$$

To reiterate in slightly different notation, for translation equivariance of a LoRA multi-head self-attention with relative positional encoding, the following equation needs to hold,

$$(14) \quad m_{LoRA}^r[L_y[f], \rho](i) = L_y[m_{LoRA}^r[f, \rho]](i)$$

where $L_y[f](i) = f(x^{-1}(x(i) - y))$. The LoRA multihead self-attention with relative positional encodings on $L_y[f]$ is:

$$\begin{aligned} & m_{LoRA}^r[L_y[f], \rho](i) \\ &= \varphi_{out} \left(\bigcup_{head \in [H]} \sum_{j \in N(i)} \frac{\exp \left\langle \varphi_{qry}^{head}(L_y[f](i)) + \Delta \varphi_{qry}^{head}(f(i)), \varphi_{key}^{head}(L_y[f](j) + \rho(i, j)) + \Delta \varphi_{key}^{head}(f(j) + \rho(i, j)) \right\rangle}{\sum_{k \in N(i)} \exp \left\langle \varphi_{qry}^{head}(L_y[f](i)) + \Delta \varphi_{qry}^{head}(f(i)), \varphi_{key}^{head}(L_y[f](k) + \rho(i, k)) + \Delta \varphi_{key}^{head}(f(k) + \rho(i, k)) \right\rangle} \right. \\ & \quad \left. \times \left(\varphi_{val}^{head}(L_y[f](j)) + \Delta \varphi_{val}^{head}(f(j)) \right) \right) \end{aligned}$$

We use the substitution $\bar{i} = x^{-1}(x(i) - y) \implies i = x^{-1}(x(\bar{i}) + y)$ and $\bar{j} = x^{-1}(x(j) - y) \implies j = x^{-1}(x(\bar{j}) + y)$, and the definition of $\rho(i, j) = \rho^P(x(j) - x(i))$:

$$\begin{aligned}
& m_{LoRA}^r[L_y[f], \rho](i) \\
&= \varphi_{out} \left(\bigcup_{head \in [H]} \sum_{\bar{j} \in N(\bar{i})} \frac{\exp \langle \varphi_{qry}^{head}(f(\bar{i})) + \Delta \varphi_{qry}^{head}(f(\bar{i})), \varphi_{key}^{head}(f(\bar{j}) + \rho(\bar{i}, \bar{j})) + \Delta \varphi_{key}^{head}(f(\bar{j}) + \rho(\bar{i}, \bar{j})) \rangle}{\sum_{\bar{k} \in N(\bar{i})} \exp \langle \varphi_{qry}^{head}(f(\bar{i})) + \Delta \varphi_{qry}^{head}(f(\bar{i})), \varphi_{key}^{head}(f(\bar{k}) + \rho(\bar{i}, \bar{k})) + \Delta \varphi_{key}^{head}(f(\bar{k}) + \rho(\bar{i}, \bar{k})) \rangle} \right. \\
&\quad \left. \times \left(\varphi_{val}^{head}(f(\bar{j})) + \Delta \varphi_{val}^{head}(f(\bar{j})) \right) \right) \\
&= m_{LoRA}^r[f, \rho](\bar{i}) \\
&= m_{LoRA}^r[f, \rho](x^{-1}(x(i) - y)) \\
&= L_y[m_{LoRA}^r[f, \rho]](i)
\end{aligned}$$

The last equations conclude that relative positional encodings, coupled with LoRAs for the query, key, and value weight matrices, result in a translation equivariant multihead self-attention mechanism. More specifically, addition of LoRAs in a translation equivariant model with relative positional encodings does not disrupt the translation equivariance.

5. LoRAs and Lifting Self-Attention

Next, we study the lifting self-attention layers defined in [RC]. For the lifting self-attention layer to be G -equivariant, we must have that $m_{G\uparrow}^r[L_g[f], \rho](i, h) = L_g[m_{G\uparrow}^r[f, \rho]](i, h)$. Consider a g -transformed input signal $L_g[f](i) = L_y L_{h_3}[f](i) = f(x^{-1}(h_3^{-1}(x(i) - y)))$. Here $g = (y, h_3) \in \mathbb{R}^{d_{in}} \rtimes \mathcal{H}$. The lifting self-attention layer with LoRAs, applied to $L_g[f]$ is:

$$\begin{aligned}
& m_{G\uparrow}^{r, LoRA}[L_y L_{h_3}[f], \rho](i, h) \\
&= \varphi_{out} \left(\bigcup_{head \in [H]} \sum_{j \in N(i)} \sigma_j \left(\langle \varphi_{qry}^{head}(L_y L_{h_3}[f](i)) + \Delta \varphi_{qry}^{head}(L_y L_{h_3}[f](i)), \varphi_{key}^{head}(L_y L_{h_3}[f](j) \right. \right. \\
&\quad \left. \left. + L_h[\rho](i, j)) + \Delta \varphi_{key}^{head}(L_y L_{h_3}[f](j) + L_h[\rho](i, j)) \rangle \right) \langle \varphi_{val}^{head}(L_y L_{h_3}[f](j)) + \Delta \varphi_{val}^{head}(L_y L_{h_3}[f](j)) \rangle \right) \\
&= \varphi_{out} \left(\bigcup_{head \in [H]} \sum_{j \in N(i)} \sigma_j \left(\langle \varphi_{qry}^{head}(f(x^{-1}(h_3^{-1}(x(i) - y)))) + \Delta \varphi_{qry}^{head}(f(x^{-1}(h_3^{-1}(x(i) - y)))) \right. \right. \\
&\quad \left. \left. + L_h[\rho](i, j) + \Delta \varphi_{key}^{head}(f(x^{-1}(h_3^{-1}(x(j) - y))) + L_h[\rho](i, j)) \rangle \right) \right. \\
&\quad \left. \times \langle \varphi_{val}^{head}(f(x^{-1}(h_3^{-1}(x(j) - y)))) + \Delta \varphi_{val}^{head}(f(x^{-1}(h_3^{-1}(x(j) - y)))) \rangle \right) \\
&= \varphi_{out} \left(\bigcup_{head \in [H]} \sum_{x^{-1}(h_3^{-1}x(\bar{j})+y) \in N(x^{-1}(h_3^{-1}x(\bar{i})+y))} \sigma_j \left(\langle \varphi_{qry}^{head}(f(\bar{i})) + \Delta \varphi_{qry}^{head}(f(\bar{i})), \varphi_{key}^{head}(f(\bar{j}) \right. \right. \\
&\quad \left. \left. + L_h[\rho](x^{-1}(h_3x(\bar{i}) + y), x^{-1}(h_3x(\bar{j}) + y)) + \Delta \varphi_{key}^{head}(f(\bar{j}) + L_h[\rho](x^{-1}(h_3x(\bar{i}) + y), x^{-1}(h_3x(\bar{j}) + y))) \rangle \right) \right. \\
&\quad \left. \times \langle \varphi_{val}^{head}(f(\bar{j})) + \Delta \varphi_{val}^{head}(f(\bar{j})) \rangle \right)
\end{aligned}$$

Here we have used $\bar{i} = x^{-1}(h_3^{-1}(x(i)-y)) \implies i = x^{-1}(h_3x(\bar{i})+y)$ and $\bar{j} = x^{-1}(h_3^{-1}(x(j)-y)) \implies j = x^{-1}(h_3x(\bar{j})+y)$. By using the definition of $\rho(i, j)$ we can further reduce the expression:

$$\begin{aligned}
&= \varphi_{out} \left(\bigcup_{head \in [H]} \sum_{x^{-1}(h_3^{-1}x(\bar{j})+y) \in N(x^{-1}(h_3^{-1}x(\bar{i})+y))} \sigma_{x^{-1}(h_3^{-1}x(\bar{j})+y)} \left(\langle \varphi_{qry}^{head} f(\bar{i}) + \Delta \varphi_{qry}^{head} f(\bar{i}), \varphi_{key}^{head} f(\bar{j}) \right. \right. \\
&\quad \left. \left. + \rho^P(h^{-1}(h_3^{-1}x(\bar{j})+y) - h^{-1}(h_3x(\bar{i})+y)) \rangle + \Delta \varphi_{key}^{head} f(\bar{j}) + \rho^P(h^{-1}(h_3^{-1}x(\bar{j})+y) - h^{-1}(h_3x(\bar{i})+y)) \rangle \right) \right) \\
&\quad \times (\varphi_{val}^{head}(f(\bar{j})) + \Delta \varphi_{val}^{head}(f(\bar{j}))) \Big) \\
&= \varphi_{out} \left(\bigcup_{h \in [H]} \sum_{x^{-1}(h_3^{-1}x(\bar{j})+y) \in N(x^{-1}(h_3^{-1}x(\bar{i})+y))} \sigma_{x^{-1}(h_3^{-1}x(\bar{j})+y)} \left(\langle \varphi_{qry}^{head} f(\bar{i}) + \Delta \varphi_{qry}^{head} f(\bar{i}), \varphi_{key}^{head} f(\bar{j}) \right. \right. \\
&\quad \left. \left. + \rho^P(h^{-1}h_3(x(\bar{j}) - x(\bar{i}))) \rangle + \Delta \varphi_{key}^{head} f(\bar{j}) + \rho^P(h^{-1}h_3(x(\bar{j}) - x(\bar{i}))) \rangle \right) \right) \\
&\quad \times (\varphi_{val}^{head}(f(\bar{j})) + \Delta \varphi_{val}^{head}(f(\bar{j}))) \Big) \\
&= \varphi_{out} \left(\bigcup_{h \in [H]} \sum_{x^{-1}(h_3^{-1}x(\bar{j})+y) \in N(x^{-1}(h_3^{-1}x(\bar{i})+y))} \sigma_{x^{-1}(h_3^{-1}x(\bar{j})+y)} \left(\langle \varphi_{qry}^{head} f(\bar{i}), \varphi_{key}^{head} f(\bar{j}) \right. \right. \\
&\quad \left. \left. + L_{h_3^{-1}h}(\bar{i}, \bar{j}) + \Delta \varphi_{key}^{head} f(\bar{j}) + L_{h_3^{-1}h}(\bar{i}, \bar{j}) \rangle \right) \right) \\
&\quad \times (\varphi_{val}^{head}(f(\bar{j})) + \Delta \varphi_{val}^{head}(f(\bar{j}))) \Big)
\end{aligned}$$

Since for unimodular (or compact) groups the area of summation remains equal for any $g \in G$, we have:

$$(15) \quad \sum_{x^{-1}(h_3x(\bar{j})+y) \in N(x^{-1}(h_3x(\bar{i})+y))} [\bullet] = \sum_{x^{-1}(h_3x(\bar{j})) \in N(x^{-1}(h_3x(\bar{i})))} [\bullet] = \sum_{x^{-1}(x(\bar{j})) \in N(x(\bar{i}))} [\bullet] = \sum_{\bar{j} \in N(\bar{i})} [\bullet]$$

As a result, we can further reduce the expression above as:

$$\begin{aligned}
&m_{G_{\uparrow}}^{r,LoRA}[L_\gamma L_{h_3}[f], \rho](i, h) \\
&= \varphi_{out} \left(\bigcup_{head \in [H]} \sum_{\bar{j} \in N(\bar{i})} \sigma_{\bar{j}} \left(\langle \varphi_{qry}^{head} f(\bar{i}) + \Delta \varphi_{qry}^{head} f(\bar{i}), \varphi_{key}^{head} f(\bar{j}) + L_{h_3^{-1}h}[\rho](\bar{i}, \bar{j}) + \Delta \varphi_{key}^{head} f(\bar{j}) + L_{h_3^{-1}h}[\rho](\bar{i}, \bar{j}) \rangle \right) \right) \\
&\quad \times (\varphi_{val}^{head}(f(\bar{j})) + \Delta \varphi_{val}^{head}(f(\bar{j}))) \Big) \\
&= m_{G_{\uparrow}}^{r,LoRA}[r, \rho](\bar{i}, h_3^{-1}h) \\
&= m_{G_{\uparrow}}^{r,LoRA}(x^{-1}(h_3^{-1}(x(i)-y)), h_3^{-1}h) \\
&= L_\gamma L_{h_3}[m_{G_{\uparrow}}^{r,LoRA}[f, \rho]](i, h)
\end{aligned}$$

So, we see that adding LoRAs does not disrupt equivariance of the lifting self-attention layer.

6. LoRAs and Group Self-Attention

Next, we can also show that the inclusion of LoRAs does not disrupt equivariance of group self-attention.

$$\begin{aligned}
& m_{G, \text{LoRA}}^r[L_y L_{h_3}[f], \rho](i, h_1) \\
&= \varphi_{out} \left(\bigcup_{head \in [H]} \sum_{h \in \mathcal{H}} \sum_{(j, h_2) \in N(i, h_1)} \sigma_{(j, h_2)} \left(\langle \varphi_{qry}^{head} L_y L_{h_3}[f](i, h_1) + \Delta \varphi_{qry}^{head} L_y L_{h_3}[f](i, h_1), \varphi_{key}^{head} (L_y L_{h_3}[f](j, h_2) \right. \right. \\
&\quad \left. \left. + L_h[\rho]((i, h_1), (j, h_2))) \rangle + \Delta \varphi_{key}^{head} (L_y L_{h_3}[f](j, h_2) + L_h[\rho]((i, h_1), (j, h_2))) \right) \right) \\
&\quad \times (\varphi_{val}^{head} (L_y L_{h_3}[f](j, h_2)) + \Delta \varphi_{val}^{head} (L_y L_{h_3}[f](j, h_2))) \\
&= \varphi_{out} \left(\bigcup_{head \in [H]} \sum_{h \in \mathcal{H}} \sum_{(j, h_2) \in N(i, h_1)} \sigma_{(j, h_2)} \left(\langle \varphi_{qry}^{head} f(x^{-1}(h_3^{-1}(x(i) - y)), h_3^{-1}h_1) + \Delta \varphi_{qry}^{head} f(x^{-1}(h_3^{-1}(x(i) - y)), h_3^{-1}h_1), \right. \right. \\
&\quad \left. \left. \varphi_{key}^{head} (f(x^{-1}(h_3^{-1}(x(i) - y)), h_3^{-1}h_2)) \right. \right. \\
&\quad \left. \left. + L_h[\rho]((i, h_1), (j, h_2))) \rangle + \Delta \varphi_{key}^{head} (f(x^{-1}(h_3^{-1}(x(i) - y)), h_3^{-1}h_2) + L_h[\rho]((i, h_1), (j, h_2))) \right) \right) \\
&\quad (\varphi_{val}^{head} (f(x^{-1}(h_3^{-1}(x(i) - y)), h_3^{-1}h_2)) + \Delta \varphi_{val}^{head} (f(x^{-1}(h_3^{-1}(x(i) - y)), h_3^{-1}h_2))) \\
&= \varphi_{out} \left(\bigcup_{head \in [H]} \sum_{h_3 h \in \mathcal{H}} \sum_{(x^{-1}(h_3 x(\bar{j}) + y), h_3 h'_2) \in N((x^{-1}(h_3 x(\bar{i}) + y), h_3 h'_1))} \sigma_{(x^{-1}(h_3 x(\bar{j}) + y), h_3 h'_2)} \left(\langle \varphi_{qry}^{head} f(\bar{i}, h'_1) + \Delta \varphi_{qry}^{head} f(\bar{i}, h'_1), \varphi_{key}^{head} (f(\bar{j}, h'_2) \right. \right. \\
&\quad \left. \left. + L_h[\rho]((x^{-1}(h_3^{-1}x(\bar{i}) + y), h_3 h'_1), (x^{-1}(h_3^{-1}x(\bar{j}) + y), h_3 h'_2))) \right. \right. \\
&\quad \left. \left. + \Delta \varphi_{key}^{head} (f(\bar{j}, h'_2) + L_h[\rho]((x^{-1}(h_3^{-1}x(\bar{i}) + y), h_3 h'_1), (x^{-1}(h_3^{-1}x(\bar{j}) + y), h_3 h'_2))) \right) \right) \\
&\quad \times (\varphi_{val}^{head} (f(\bar{j}, h'_2)) + \Delta \varphi_{val}^{head} (f(\bar{j}, h'_2)))
\end{aligned}$$

Next, we use the definition of $L_h[\rho]((i, h_1), (j, h_2)) = \rho^P(h^{-1}(x(j) - x(i)), h_1^{-1}h_2)$ to reduce the equations further:

$$\begin{aligned}
&= \varphi_{out} \left(\bigcup_{head \in [H]} \sum_{h_3 h \in \mathcal{H}} \sum_{(x^{-1}(h_3 x(\bar{j})+y), h_3 h'_2) \in N((x^{-1}(h_3 x(\bar{i})+y), h_3 h'_1))} \sigma_{(x^{-1}(h_3 x(\bar{j})+y), h_3 h'_2)} \left(\langle \varphi_{qry}^{head} f(\bar{i}, h'_1) + \Delta \varphi_{qry}^{head} f(\bar{i}, h'_1), \varphi_{key}^{head}(f(\bar{j}, h'_2)) \right. \right. \\
&\quad \left. \left. + \rho^P(h^{-1}(h_3 x(\bar{j}) + y - (h_3 x(\bar{i}) + y)), (h_3 h'_1)^{-1}(h_3 h'_2)) \right. \right. \\
&\quad \left. \left. + \Delta \varphi_{key}^{head}(f(\bar{j}, h'_2)) + \rho^P(h^{-1}(h_3 x(\bar{j}) + y - (h_3 x(\bar{i}) + y)), (h_3 h'_1)^{-1}(h_3 h'_2)) \rangle \right) \right) \\
&\quad \times (\varphi_{val}^{head}(f(\bar{j}, h'_2)) + \Delta \varphi_{val}^{head}(f(\bar{j}, h'_2))) \Big) \\
&= \varphi_{out} \left(\bigcup_{head \in [H]} \sum_{h_3 h \in \mathcal{H}} \sum_{(x^{-1}(h_3 x(\bar{j})+y), h_3 h'_2) \in N((x^{-1}(h_3 x(\bar{i})+y), h_3 h'_1))} \sigma_{(x^{-1}(h_3 x(\bar{j})+y), h_3 h'_2)} \left(\langle \varphi_{qry}^{head} f(\bar{i}, h'_1) + \Delta \varphi_{qry}^{head} f(\bar{i}, h'_1), \varphi_{key}^{head}(f(\bar{j}, h'_2)) \right. \right. \\
&\quad \left. \left. + \rho^P(h^{-1}(h_3 x(\bar{j}) - h_3 x(\bar{i})), h'_1{}^{-1} h'_2) + \Delta \varphi_{key}^{head}(f(\bar{j}, h'_2)) + \rho^P(h^{-1}(h_3 x(\bar{j}) - h_3 x(\bar{i})), h'_1{}^{-1} h'_2) \rangle \right) \right) \\
&\quad \times (\varphi_{val}^{head}(f(\bar{j}, h'_2)) + \Delta \varphi_{val}^{head}(f(\bar{j}, h'_2))) \Big) \\
&= \varphi_{out} \left(\bigcup_{head \in [H]} \sum_{h_3 h \in \mathcal{H}} \sum_{(x^{-1}(h_3 x(\bar{j})+y), h_3 h'_2) \in N((x^{-1}(h_3 x(\bar{i})+y), h_3 h'_1))} \sigma_{(x^{-1}(h_3 x(\bar{j})+y), h_3 h'_2)} \left(\langle \varphi_{qry}^{head} f(\bar{i}, h'_1) + \Delta \varphi_{qry}^{head} f(\bar{i}, h'_1), \varphi_{key}^{head}(f(\bar{j}, h'_2)) \right. \right. \\
&\quad \left. \left. + L_{h_3^{-1}h}[\rho](\bar{i}, h'_1), (\bar{j}, h'_2)) + \Delta \varphi_{key}^{head}(f(\bar{j}, h'_2)) + L_{h_3^{-1}h}[\rho](\bar{i}, h'_1), (\bar{j}, h'_2) \rangle \right) (\varphi_{val}^{head}(f(\bar{j}, h'_2)) + \Delta \varphi_{val}^{head}(f(\bar{j}, h'_2))) \right)
\end{aligned}$$

Now, since G is assumed unimodular (or compact) we have that the following summations are equal,

$$(16) \quad \sum_{(x^{-1}(h_3 x(\bar{j})+y), h_3 h'_2) \in N((x^{-1}(h_3 x(\bar{i})+y), h_3 h'_1))} [\bullet] = \sum_{(x^{-1}(h_3 x(\bar{j})), h_3 h'_2) \in N((x^{-1}(h_3 x(\bar{i})), h_3 h'_1))} [\bullet] = \sum_{(x^{-1}(x(\bar{j})), h'_2) \in N(x^{-1}(x(\bar{i})), h'_1)} [\bullet] = \sum_{(\bar{j}, h'_2) \in N(\bar{i}, h'_1)} [\bullet]$$

We also have that $\sum_{h_3 h \in \mathcal{H}} [\bullet] = \sum_{h \in \mathcal{H}} [\bullet]$. Finally, we make the following observations,

$$\begin{aligned}
&m_{G, LoRA}^r[L_y L_{h_3}[f], \rho](i, h_1) \\
&= \varphi_{out} \left(\bigcup_{head \in [H]} \sum_{h \in \mathcal{H}} \sum_{(\bar{j}, h'_2) \in N(\bar{i}, h'_1)} \sigma_{(\bar{j}, h'_2)} \left(\langle \varphi_{qry}^{head}(f(\bar{i}, h'_1)) + \Delta \varphi_{qry}^{head}(f(\bar{i}, h'_1)), \varphi_{key}^{head}(f(\bar{j}, h'_2)) + \right. \right. \\
&\quad \left. \left. L_{h_3^{-1}h}[\rho](\bar{i}, h'_1), (\bar{j}, h'_2)) + \Delta \varphi_{key}^{head}(f(\bar{j}, h'_2)) + L_{h_3^{-1}h}[\rho](\bar{i}, h'_1), (\bar{j}, h'_2) \rangle \right) (\varphi_{val}^{head}(f(\bar{j}, h'_2)) + \Delta \varphi_{val}^{head}(f(\bar{j}, h'_2))) \right) \\
&= m_{G, LoRA}^r[f, \rho](\bar{i}, h_3^{-1} h_1) \\
&= m_{G, LoRA}^r[f, \rho](x^{-1}(h_3^{-1}(x(i) - y)), h_3^{-1} h_1) \\
&= L_y L_{h_3}[m_{G, LoRA}^r[f, \rho]](i, h_1)
\end{aligned}$$

So, we see that in fact $m_{G, LoRA}^r[L_y L_{h_3}[f], \rho](i, h) = L_y L_{h_3}[m_{G, LoRA}^r[f, \rho]](i, h)$, and LoRAs do not disrupt equivariance of group self-attention.

We can see for the following setup, that the same arguments work if we include a LoRA $\Delta\varphi_{out}$ for φ_{out} . In particular, including such a LoRA does not disrupt the equivariance of the group self-attention layers.

$$\begin{aligned}
m_{G,LoRA}^r[L_y L_{h_3}[f], \rho](i, h_1) = & \varphi_{out} \left(\bigcup_{head \in [H]} \sum_{h \in \mathcal{H}} \sum_{(j, h_2) \in N(i, h_1)} \sigma_{(j, h_2)} \left(\langle \varphi_{qry}^{head} L_y L_{h_3}[f](i, h_1) + \Delta\varphi_{qry}^{head} L_y L_{h_3}[f](i, h_1), \right. \right. \\
& \varphi_{key}^{head}(L_y L_{h_3}[f](j, h_2) \\
& \left. \left. + L_h[\rho]((i, h_1), (j, h_2))) + \Delta\varphi_{key}^{head}(L_y L_{h_3}[f](j, h_2) + L_h[\rho]((i, h_1), (j, h_2))) \rangle \right) \\
& (\varphi_{val}^{head}(L_y L_{h_3}[f](j, h_2)) + \Delta\varphi_{val}^{head}(L_y L_{h_3}[f](j, h_2))) \Big) \\
& + \Delta\varphi_{out} \left(\bigcup_{head \in [H]} \sum_{h \in \mathcal{H}} \sum_{(j, h_2) \in N(i, h_1)} \sigma_{(j, h_2)} \left(\langle \varphi_{qry}^{head} L_y L_{h_3}[f](i, h_1) + \Delta\varphi_{qry}^{head} L_y L_{h_3}[f](i, h_1), \right. \right. \\
& \varphi_{key}^{head}(L_y L_{h_3}[f](j, h_2) \\
& \left. \left. + L_h[\rho]((i, h_1), (j, h_2))) + \Delta\varphi_{key}^{head}(L_y L_{h_3}[f](j, h_2) + L_h[\rho]((i, h_1), (j, h_2))) \rangle \right) \\
& (\varphi_{val}^{head}(L_y L_{h_3}[f](j, h_2)) + \Delta\varphi_{val}^{head}(L_y L_{h_3}[f](j, h_2))) \Big)
\end{aligned}$$

Note, a similar argument holds for translation equivariance, and for lifting self-attention layers as well. Thus, we may include LoRAs $\Delta\varphi_{out}$ in translation equivariant models, and in the lifting self-attention layers as well without disrupting equivariance.

7. Appendix

In [RC] there are a few calculation errors in the proof of the equivariance of group self-attention layers. Below is an explanation of why the proof of the equivariance property does not work and how to fix it. Let $L_g[f](i, h_1) = L_y L_{h_3}[f](i, h_1) = f(x^{-1}(h_3^{-1}(x(i) - y)), h_3^{-1}h_1)$ be a g -transformed input signal, where $g = (y, h_3) \in G = \mathbb{R}^{d_{in}} \rtimes \mathcal{H}$. The group self-attention operation on $L_g[f]$ is given by:

$$\begin{aligned}
m_G^r[L_y L_{h_3}[f], \rho](i, h_1) &= \varphi_{out} \left(\bigcup_{head \in [H]} \sum_{h \in \mathcal{H}} \sum_{(j, h_2) \in N(i, h_1)} \sigma_{(j, h_2)} \left(\langle \varphi_{qry}^{head}(L_y L_{h_3}[f](i, h_1)), \varphi_{key}^{head}(L_y L_{h_3}[f](j, h_2)) \right. \right. \\
&\quad \left. \left. + L_h[\rho]((i, h_1), (j, h_2))) \right) \varphi_{val}^{head}(L_y L_{h_3}[f](j, h_2)) \right) \\
&= \varphi_{out} \left(\bigcup_{head \in [H]} \sum_{h \in \mathcal{H}} \sum_{(j, h_2) \in N(i, h_1)} \sigma_{(j, h_2)} \left(\langle \varphi_{qry}^{head}(f(x^{-1}(h_3^{-1}(x(i) - y))), h_3^{-1}h_1), \right. \right. \\
&\quad \varphi_{key}^{head}(f(x^{-1}(h_3^{-1}(x(i) - y))), h_3^{-1}h_2) \\
&\quad \left. \left. + L_h[\rho]((i, h_1), (j, h_2))) \right) \varphi_{val}^{head}(f(x^{-1}(h_3^{-1}(x(i) - y))), h_3^{-1}h_2)) \right) \\
&= \varphi_{out} \left(\bigcup_{head \in [H]} \sum_{h_3 h \in \mathcal{H}} \sum_{(x^{-1}(h_3 x(\bar{j}) + y), h_3 h'_2) \in N((x^{-1}(h_3 x(\bar{i}) + y), h_3 h'_1))} \sigma_{(x^{-1}(h_3 x(\bar{j}) + y), h_3 h'_2)} \left(\langle \varphi_{qry}^{head}(f(\bar{i}, h'_1)), \varphi_{key}^{head}(f(\bar{j}, h'_2)) \right. \right. \\
&\quad \left. \left. + L_h[\rho]((x^{-1}(h_3^{-1}x(\bar{i}) + y), h_3 h'_1), (x^{-1}(h_3^{-1}x(\bar{j}) + y), h_3 h'_2))) \right) \varphi_{val}^{head}(f(\bar{j}, h'_2)) \right)
\end{aligned}$$

Here, we have used $\bar{i} = x^{-1}(h_3^{-1}(x(i) - y)) \implies i = x^{-1}(h_3^{-1}x(\bar{i}) + y)$, and $\bar{j} = x^{-1}(h_3^{-1}(x(j) - y)) \implies j = x^{-1}(h_3^{-1}x(\bar{j}) + y)$, and $h'_1 = h_3^{-1}h_1$ and $h'_2 = h_3^{-1}h_2$. By using the definition of $L_h \rho((i, h_1), (j, h_2)) = \rho^P(h^{-1}(x(j) - x(i)), h^{-1}h_1^{-1}h_2)$ we can further reduce the equations:

$$\begin{aligned}
&= \varphi_{out} \left(\bigcup_{head \in [H]} \sum_{h_3 h \in \mathcal{H}} \sum_{(x^{-1}(h_3 x(\bar{j}) + y), h_3 h'_2) \in N((x^{-1}(h_3 x(\bar{i}) + y), h_3 h'_1))} \sigma_{(x^{-1}(h_3 x(\bar{j}) + y), h_3 h'_2)} \left(\langle \varphi_{qry}^{head}(f(\bar{i}, h'_1)), \varphi_{key}^{head}(f(\bar{j}, h'_2)) \right. \right. \\
&\quad \left. \left. + \rho^P(h^{-1}(h_3 x(\bar{j}) + y - (h_3 x(\bar{i}) + y)), h^{-1}(h_3 h'_1)^{-1}(h_3 h'_2)) \right) \varphi_{val}^{head}(f(\bar{j}, h'_2)) \right) \\
&= \varphi_{out} \left(\bigcup_{head \in [H]} \sum_{h_3 h \in \mathcal{H}} \sum_{(x^{-1}(h_3 x(\bar{j}) + y), h_3 h'_2) \in N((x^{-1}(h_3 x(\bar{i}) + y), h_3 h'_1))} \sigma_{(x^{-1}(h_3 x(\bar{j}) + y), h_3 h'_2)} \left(\langle \varphi_{qry}^{head}(f(\bar{i}, h'_1)), \varphi_{key}^{head}(f(\bar{j}, h'_2)) \right. \right. \\
&\quad \left. \left. + \rho^P(h^{-1}(h_3 x(\bar{j}) - h_3 x(\bar{i})), h^{-1}h_1^{-1}h'_2)) \right) \varphi_{val}^{head}(f(\bar{j}, h'_2)) \right) \quad \text{notice the lack of a factor of } h_3^{-1} \text{ in front of } h_1'^{-1}h'_2 \text{ here} \\
&\neq \varphi_{out} \left(\bigcup_{head \in [H]} \sum_{h_3 h \in \mathcal{H}} \sum_{(x^{-1}(h_3 x(\bar{j}) + y), h_3 h'_2) \in N((x^{-1}(h_3 x(\bar{i}) + y), h_3 h'_1))} \sigma_{(x^{-1}(h_3 x(\bar{j}) + y), h_3 h'_2)} \left(\langle \varphi_{qry}^{head}(f(\bar{i}, h'_1)), \varphi_{key}^{head}(f(\bar{j}, h'_2)) \right. \right. \\
&\quad \left. \left. + L_{h_3^{-1}h}[\rho](\bar{i}, h'_1), (\bar{j}, h'_2)) \right) \varphi_{val}^{head}(f(\bar{j}, h'_2)) \right)
\end{aligned}$$

So, we see that $m_G^r[L_y L_{h_3}[f], \rho](i, h) \neq L_y L_{h_3}[m_G^r[f, \rho]](i, h)$. It is also said in [RC] that the equivariance of group self-attention is due to the relative positional encoding being ****invariant**** to the group action. That is, it is claimed to follow from $L_g[\rho](i, j) = \rho(i, j)$. This however is false. In particular, the positional encoding $\rho(i, j)$ is not used in the proof, it is the positional encoding $\rho((i, h_1), (j, h_2))$ that is used. Furthermore, we see that it is not G -invariant unless $x(j) - x(i)$ is an H -invariant vector.

$$\begin{aligned}
L_g[\rho]((i, h_1), (j, h_2)) &= L_y L_h[\rho]((i, h_1), (j, h_2)) \\
&= L_h[L_y[\rho]]((i, h_1), (j, h_2)) \\
&= L_h[\rho]((x^{-1}(x(i) - y), h_1), (x^{-1}(x(j) - y), h_2)) \\
&= \rho((x^{-1}(h^{-1}(x(i) - y)), h^{-1}h_1), (x^{-1}(h^{-1}(x(j) - y)), h^{-1}h_2)) \\
&= \rho^P(h^{-1}(x(j) - y) - h^{-1}(x(i) - y), (h^{-1}h_1)^{-1}(h^{-1}h_2)) \\
&= \rho^P(h^{-1}(x(j) - x(i)), (h^{-1}h_1)^{-1}(h^{-1}h_2)) \\
&= \rho^P(h^{-1}(x(j) - x(i)), h_1^{-1}h h^{-1}h_2) \\
&= \rho^P(h^{-1}(x(j) - x(i)), h_1^{-1}h_2) \\
&\neq \rho((i, h_1), (j, h_2)) \quad \text{unless } h^{-1}(x(j) - x(i)) = x(j) - x(i)
\end{aligned}$$

This means that group self-attention is not group equivariant, as is. The other proofs in the paper do in fact work, however. Moreover, if we redefine $L_h[\rho]((i, h_1), (j, h_2))$ as $\rho^P(h^{-1}(x(j) - x(i)), h_1^{-1}h_2)$ we see that equivariance does in fact hold, and that the positional encoding is \mathcal{H} -invariant in the group component.

8. Conclusions

8.1. LoRAs and Translation Equivariance with Relative Positional Encodings. The Low-Rank Adaptations (LoRAs) present a unique opportunity to retain the translation equivariance of the transformer models when relative positional encodings are utilized. Relative positional encodings provide the ability to capture the relative positions of elements in the sequence, enabling the model to retain translation equivariance. This means the response of the model does not change when the input is shifted or translated. In the context of OCR, this is especially crucial, as the position of text within an image can vary greatly, and being able to recognize text regardless of its position is key to successful OCR.

LoRAs, with their low-rank adaptations, can maintain this translation equivariance. By operating in a low-rank subspace of the model's parameters, they enable efficient and effective adaptation of the model to specific tasks or domains without disrupting the inherent properties of the model, including its translation equivariance. Through this, LoRAs allow for the adaptation of models with relative positional encodings, enhancing their performance on tasks such as OCR while retaining their beneficial inductive biases.

8.2. Group Equivariant Self-Attention in Vision Transformers. Group equivariant self-attention is a recent development in the field of machine learning, specifically in the design of attention mechanisms for transformers. Understanding its usefulness requires a comprehension of the underlying concepts it builds upon:

- (1) **Self-Attention:** In transformers, the model computes a weighted sum of all inputs, where the weights are determined by the similarity of each input to all others. This mechanism, known as self-attention, allows the model to capture dependencies between inputs irrespective of their distance in the input sequence.
- (2) **Equivariance:** An equivariant system in machine learning and pattern recognition is one where the output changes in the same way as the input. For instance, if an image is rotated, an equivariant system will produce a correspondingly rotated output. This contrasts with invariance, where the output remains unchanged despite alterations to the input.

- (3) Group Equivariance: This concept extends equivariance. Here, the "group" refers to a mathematical group, a set of elements coupled with an operation that combines any two elements to form a third. In this context, the group can be viewed as a set of transformations (e.g., rotations, translations, reflections) that can be applied to the input. A system is group equivariant if its output changes in the same way as the input for any transformation in the group.

A group equivariant self-attention mechanism, therefore, maintains the property of equivariance with respect to a group of transformations when computing the attention weights. So, why is this mechanism useful in vision transformers?

- (1) Enhanced Spatial Understanding: By being equivariant to transformations, the model can better comprehend spatial relationships in the input data. This is particularly useful in vision tasks where the same object can appear in different positions, orientations, or scales in the image.
- (2) Data Efficiency: Equivariance can render the model more data-efficient, as it needs fewer examples to learn that certain patterns are meaningful regardless of their position or orientation in the image.
- (3) Robustness: The model can be more robust to changes in the input data, such as changes in the viewpoint, lighting, or object position by being insensitive to certain transformations.
- (4) Interpretability: Equivariance can increase model interpretability, as it allows us to reason about how the model's output will change in response to specific transformations of the input.

It is important to note that these benefits depend on the specific group of transformations to which the model is equivariant. The choice of group should ideally mirror the types of transformations that the input data can undergo in the problem domain. For example, in vision tasks, it is often beneficial to consider groups that include rotations and translations, as these are common transformations that images can undergo.

8.3. LoRAs for Specific Scripts and Time Periods. LoRAs offer a unique advantage when adapting models to recognize specific scripts or text from specific time periods. Given the vast diversity of written scripts and the evolution of language and writing styles over time, it is a challenging task to develop models that can accurately recognize and interpret such text.

Through the use of LoRAs, models can be fine-tuned in a low-rank subspace of their parameters, enabling them to adapt to the unique characteristics of different scripts or historical texts. This adaptation occurs without disrupting the inherent properties of the models, such as their translation equivariance when relative positional encodings are used. Thus, LoRAs provide an efficient and effective method to adapt OCR models to specific scripts and time periods, enhancing their performance on these specialized tasks.

8.4. Final Remarks. The advent of group equivariant self-attention (GSA) in transformer networks, or GSA-Nets, has ushered in a new era of enhanced spatial understanding, data efficiency, robustness, and interpretability in vision tasks. In particular, their use as vision encoders in Transformer OCR (TrOCR) systems can lead to significantly improved performance. As GSA-Nets maintain the property of equivariance to a group of transformations, they are able to comprehend the spatial relationships in the input data more effectively. This is particularly beneficial in OCR tasks, where the same text can appear in different positions, orientations, or scales within an image. Additionally, the robustness of GSA-Nets

to changes in the input data, such as changes in the viewpoint, lighting, or object position, can lead to enhanced performance in diverse and challenging real-world OCR tasks.

Simultaneously, the use of Low-Rank Adaptations (LoRAs) in the vision and language transformer components of TrOCR systems can provide further benefits. LoRAs have the unique ability to retain the translation equivariance of the transformer models when relative positional encodings are utilized, a property that is critical to successful OCR. Moreover, LoRAs can efficiently and effectively adapt the model to specific tasks or domains, making them particularly well-suited for recognizing specific scripts, handwriting styles, or text from specific time periods.

For instance, in historical document digitization, texts can vary greatly in style and appearance based on the time period, the geographical region, and even the individual writer. Traditional OCR models often struggle with such diversity and variability in the data. However, with LoRAs, models can be fine-tuned in a low-rank subspace of their parameters, enabling them to adapt to these unique characteristics without disrupting the inherent properties of the models. LoRAs also provide a cost effective and efficient way of doing this. Thus, the use of LoRAs in TrOCR systems can significantly improve their performance on these specialized tasks, leading to more accurate and reliable digitization of historical documents.

In conclusion, the use of GSA-Nets as vision encoders and LoRAs in the vision and language transformer components of TrOCR systems can lead to substantial improvements in their performance, particularly in specialized OCR tasks involving specific scripts, handwriting styles, or time periods. Therefore, it is highly recommended to incorporate these techniques into TrOCR systems for improved OCR performance.

References

- [Hu et. al.] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, LoRA: Low-Rank Adaptation of Large Language Models, <https://arxiv.org/abs/2106.09685>
- [Li et. al.] Minghao Li, Tengchao Lv, Jingye Chen, Lei Cui, Yijuan Lu, Dinei Florencio, Cha Zhang, Zhoujun Li, Furu Wei, TrOCR: Transformer-based Optical Character Recognition with Pre-trained Models, <https://arxiv.org/abs/2109.10282>
- [MK] Shahar Mahpod, Yosi Keller, Auto-ML Deep Learning for Rashi Scripts OCR, <https://arxiv.org/abs/1811.01290>
- [RC] David W. Romero, Jean-Baptiste Cordonnier, Group Equivariant Stand-Alone Self-Attention For Vision, <https://arxiv.org/abs/2010.00977>

Email address: amelie.schreiber.math@gmail.com