CNN With Attention Mechanism for Cifar-10 Classification

Group 18:Ouxiang Sun,Marc Perales Salomo,David Ademola,Firas Marzouk,Dandi Harmanto

***Abstract*—In this project, we are going to perform an image classification task on the Cifar-10 dataset using deep learning. Since Transformer has now become the de-facto standard in classification tasks, we design a hybrid neural network model that combines a convolutional neural network with an attention mechanism. The model consists of three modules: a convolutional module, a self-attention based encoder and a residual module. These hybrid model introduce desirable properties of convolutional neural networks (CNNs) to the Transformer architecture (shift, scale, and distortion invariance) while maintaining the merits of Transformers. Our error rate on the Cifar-10 classification task is 18%.**

I. Introduction

For this task, in detail, it is necessary to design a machine learning model, use the model to train on the training set of cifar-10, and evaluate the accuracy of the model on the test set after the model is trained. The test set cannot be used during the training process. For image classification tasks, Krizhevsky[1] first used a deep neural network for modeling. Residual neural network was proposed by HeKaiming in [2], which enabled the training of deep neural network with more layers, and achieved SOTA on Imagenet that year. Transformers[3] have recently dominated a wide range of tasks in natural language processing (NLP). The Vision Transformer[4] is the first computer vision model to rely exclusively on the Transformer architecture to obtain competitive image classification performance at large scale.

Despite the success of vision Transformers at large scale, the performance is still below similarly sized convolutional neural network (CNN) counterparts (e.g., ResNets[2]) when trained on smaller amounts of data. One possible reason may be that ViT lacks certain desirable properties inherently built into the CNN architecture that make CNNs uniquely suited to solve vision tasks. In this project, we hypothesize that convolutions can be strategically introduced to the transformer structure to improve performance and robustness. To verify our hypothesises, we designed a hybrid architecture combining convolutional neural network and attention mechanism.

II. Method

In this section we will introduce our network architecture in detail. The model consists of two identical two-convolution, one-attention, one-residual modules. We will introduce each sub-module in modules.

Convolution Module

Each convolution module contains a layer of convolutional neural network, which needs to specify the number of input channels and output channels. By default, pooling is not used. The size of the convolution kernel is set to 3, and the padding is set to 1, so only the number of channels is changed after the convolution operation. Then take a BatchNorm layer and a ReLU activation function. If pooling is set to True, there will be a maxpooling layer after the activation function to reduce the size of the feature map. This can be formulated as:

$$X = Conv2d(X) \tag{1}$$
$$X = BatchNorm2d(X) \tag{2}$$
$$X = ReLU(X) \tag{3}$$
$$X = MaxPooling2d(X), if pooling = True \tag{4}$$

Attention Module

The general attention mechanism makes use of three main components, namely the queries, the keys, and the values. Each query vector is matched against a database of keys to compute a score value. This matching operation is computed as the dot product of the specific query under consideration with each key vector:

$$e_{q,ki} = q \cdot ki \tag{5}$$

The scores are passed through a softmax operation to generate the weights:

$$\alpha_{q,ki} = softmax(e_{q,ki}) \tag{6}$$

The generalized attention is then computed by a weighted sum of the value vectors, where each value vector is paired with a corresponding key:

$$Attention(Q, K, V) = \sum_i \alpha_{q,ki} \cdot vi \tag{7}$$

When using the attention mechanism for global modeling in this model, first converting the three-dimensional image features into a two-dimensional sequence, then use a multi-head self-attention mechanism to process. Finally, the two-dimensional features are reshaped to obtain three-dimensional image features. This can be formulated as:

$$X = X.Reshape(HW, C) \tag{8}$$
$$X = Linear(X) \tag{9}$$
$$Q, K, V = XW_q, XW_k, XW_v \tag{10}$$
$$X = Attention(Q, K, V) + X \tag{11}$$
$$X = Linear(X) \tag{12}$$
$$X = X.Reshape(C, H, W) \tag{13}$$

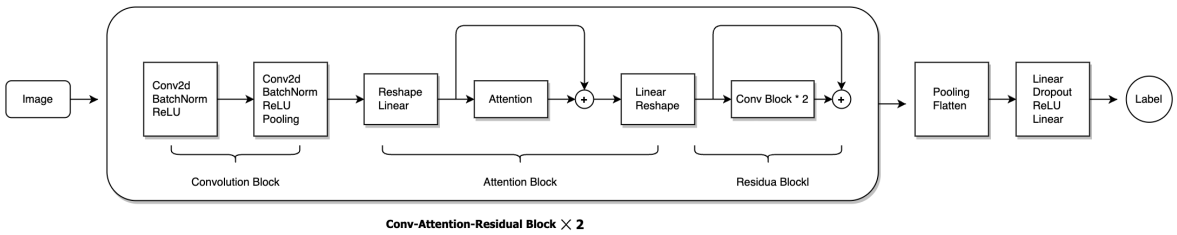

Fig. 1. This is a picture.

Residual Module

The residual connection module adopts the operation of the convolution module, and its output dimension is the same as the input dimension, and does not use the pooling operation. Each residual module contains two convolution modules, and the dimensions and channels of the feature are not changed after the residual module. This can be formulated as:

$$X^{'} = X \tag{14}$$

$$X = Conv2d(X) \tag{15}$$

$$X = Conv2d(X) \tag{16}$$

$$X = X + X^{'} \tag{17}$$

Loss Function & optimization

We choose to use SGD to optimize our model and upgrade its parameters. For Loss Function, we choose to use Cross-Entropy Loss, in the context of an optimization algorithm, the function used to evaluate a candidate solution is referred to as the objective function.

$$Loss = -\frac{1}{N}\sum_{i}\sum_{c=1}^{M} y_{ic} log(p_{ic})$$

M:number of image types, y_{ic}:symbolic function, take 1 if the true class of sample i is equal to c, otherwise take 0, $p_{ic}$the predicted probability that the observed sample i belongs to the category c.

The "gradient" in gradient descent refers to an error gradient. The model with a given set of weights is used to make predictions and the error for those predictions is calculated. The gradient descent algorithm seeks to change the weights so that the next evaluation reduces the error, meaning the optimization algorithm is navigating down the gradient (or slope) of error.

III. Result & Evaluation

For evaluation metrics, we choose to use average cross-entropy loss and model accuracy. On the test set, we only use the accuracy rate. The specific method is to assign the label with the highest probability to the image to be predicted, and then calculate the proportion of the number of test images with correct labels to the total number.

We conduct three sets of experiments, first, we train only with the architecture of convolutional neural network and residual connections, in this case the accuracy on the test set is 73%, or the error rate is 27%. Then we added the multi-head self-attention mechanism to the convolutional layer. After training, the architecture effect of adding the self-attention layer became worse, with only 60% accuracy. Later, we added an embedding layer before and after the self-attention layer, and the accuracy rate reached 81.6%.The test set error rate is 18.4%.

Hyper-Parameters: epochs 12, weight decay 0.0001, learning rate 0.008. The Python Environment: Python 3.8, Torch 1.11.0, torchvision 0.12.0. Hardware: Intel(R) Xeon(R) Gold 6330 CPU @ 2.00GHz, GPU RTX 3090 @ 24G. About 20 minutes to train and test.

IV. Conclusion

In this work, we have presented a detailed study of introducing Vision Transformer architecture into Convolution to merge the benefits of Transformers with the benefits of CNNs for image classification tasks. Experiments demonstrate that the hybrid architecture achieve superior performance.This model also has some shortcomings. We found during training that after adding the Transformer structure, the model is prone to gradient explosion and becomes sensitive to the learning rate.

Future Work: In this project, we combine the convolutional neural network with the encoder structure of the Transformer. In the future, we will explore the network architecture of Transformer as the main structure, and use convolutional layers to help Transformer try to solve problems that cannot generalize on small-scale datasets.

References

[1] Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." Advances in neural information processing systems 25 (2012).

[2] He, Kaiming, et al. "Deep residual learning for image recognition." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.

[3] Vaswani, Ashish, et al. "Attention is all you need." Advances in neural information processing systems 30 (2017).

[4] Dosovitskiy, Alexey, et al. "An image is worth 16x16 words: Transformers for image recognition at scale." arXiv preprint arXiv:2010.11929 (2020).