Vision Transformer (ViT)

Summary of the paper –

The paper was published in 2020 and at that time transformers were still new and were mostly used in NLPs and they still are not that famous in the vision sector. Many other authors tried to converge vision with transforms, but they could not find the correct implementation. They would either make small modifications to the CNN architecture or replace the convolution section, and therefore they could gain higher accuracy. Applying self-attention section in vision was one of the hardest parts as the other authors tried different ways but could not find the one where they would get the most accuracy. Also, transformers need to train on high number of images to understand the underlying pattern. To provide an image input to the transformer the authors converted the image into patches and these patches are treated like the tokens in NLP. The patches along with the positional embeddings are sent as an input to the transformer encoder. Learnable 'classification tokens' are also attached to the patches. For the designing, the authors tried to create their model as similar as the original transformer model. Unlike the prior works in self-attention for computer vision, the authors do not introduce image-specific inductive biases into the architecture apart from the initial patch extraction step. The ViT leverages the standard Multi-Head Self-Attention by treating projected image patches as Query, Key, and Value vectors to calculate the dynamic relevance between all parts of the image. After self-attention aggregates global information, a Multi-Layer Perceptron (MLP) block consisting of two dense layers locally and non-linearly transforms the aggregated features for each patch embedding independently. The first layer typically expands the dimension of the input by a large factor (e.g., a factor of 4). This expansion allows the model to project the features into a much richer, higher-dimensional space where complex non-linear interactions can be learned. The Gaussian Error Linear Unit (GELU) function is applied after the expansion, introducing the necessary non-linearity for the network to learn complex patterns. The second layer contracts the dimension back down to the original size of the patch embedding. Finally, after passing through all encoder blocks, the CLS (classification) token is extracted as the feature representation of the entire image and is passed to a simple MLP classification head to generate the final class prediction.

Inductive biases - Inductive bias can be defined as the set of assumptions or biases that a learning algorithm employs to make predictions on unseen data based on its training data. These assumptions are inherent in the algorithm's design and serve as a foundation for learning and generalization.

My approach-

Firstly, I used CIFAR-100 dataset for all my model variations. I had first created the ViT model using PyTorch from scratch so I had to create the multi-head attention, MLP layer and Transformer Block and then I combined the 3 of them in the ViT model. I could not train the model for more than epochs due to hardware limitations and this goes for all the model variations. And as I used only 1 epoch and created the model from scratch the accuracy was quite low. Then I created the model using Transfer Learning in pytorch where I used the pre-trained model 'vit_tiny_patch16_224' and only adding the classification head. The accuracy was still low but better than the scratch model. Then the created the model again using Transfer Learning but this time using Tensorflow. But, due to tensorflow compatibility issue I could not use pre-trained ViT model and I had to choose MobileNetV3 model. This was the most accurate model. Then I created the GUI using streamlit on the PyTorch TransferLearning model.

My understanding of ViT model-

ViT was one of the first models to implement Image classification using Transformers. The image is first converted into small size patches which are then concatenated with the positional embeddings and an external layer for classification, and then the addition of this is given as an input for an encoder transformer. The encoder model consists of multi-head self-attention layers and multi-layer perceptron (MLP). The attention layer finds the important features in the images. Then the MLP section enriches the features by converting the patch embeddings to a higher dimension and then uses the GELU activation layer to introduce non-linearity to understand the complex patterns and then the embeddings are shrunk to their previous dimension. Then the classification head layer classifies the image.