ATML Assignment 0

Abdullah Mushtaq

Abstract

This assignment was done to brush up on my ML concepts regarding the new and upcoming technologies in the field - some of which have completely wiped out contemporaries (eg CNNs)

1. Task 2

1.1. Introduction

This section deals with Vision Image Transformers. For this task, I have imported a ViT model trained on ImageNet-21K.

1.2. Methodology

First, the ”google/vit-base-patch16-224” model was imported and set up. For this example, I used images of cats to test my model.

Model was set up with gradient updates disabled and images passed through it. The class label and index the model predicted were noted.

Next, the attention patches were extracted and presented as a heatmap overlaying the original image. For this task, I used the clcs token.

Finally, I masked the attention in two separate ways to see if the model could still predict the image labels. In one instance, a patch in the middle of a cat was masked, and in another, random masking all over the image was used.

1.3. Results

For this section, I will refer to a specific single example to illustrate my findings (as the model had similar results for all three images.

Firstly, the model was correctly able to predict both the class label as ”tabby cat”.

The heatmap generated, also gave a peculiar pattern - more on this later.

Finally, masking yielded interested results. While the model was able to give accurate prediction of the original image AND the center masked image; it was unable to predict that

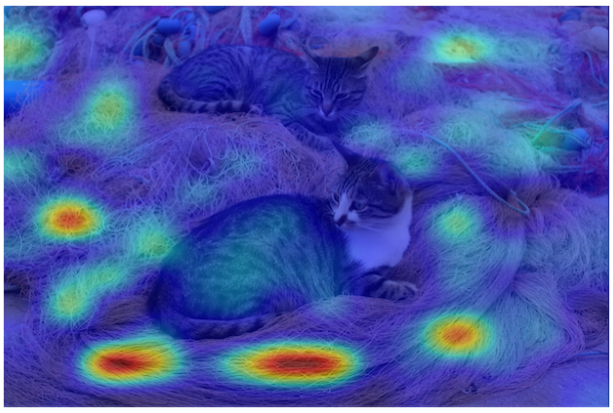


Figure 1. ViT Heatmap

Figure 2. Masked images

it was looking at an image of a tabby cat and instead labeled the image as ”crossword puzzle”

1.4. Discussion

The heatmap hotspots are focused around the edges of the cat the most. Some mildly bright areas are centered on the body of the cat as well. Interestingly, the ca’t head and neck recieve little attention.

We can interpret this as the model learning the meaning of ’cat’ from its shape, rather than feachers like ears, nose of ever fur and texture. Hence, why the bright spots center around the cat and not on it.

Since the model correctly identified the cat as a ’tabby cat’ we can interpret that the mildly glowing spots on the body of the cat were from the model using the color/texture/patterns on the fur to identify the species of cat.

This is precicely why the center masked image did not have a different label - the model focuses around the cat mostly and

therefore masking the center had no effect on the prediction.

On the other hand, random masking obscured the cat's outline - hence why a human would be able to identify the image, still, but the model is completely unable to.

1.5. Conclusion

With this experiment we have concluded how ViT transformers 'see' images and how they classify them.

ViTs classify and 'see' by mainly noting the edges of an object. That is how they are able to differentiate a cat from another animal. They also take some note of the body/texture/color as well to further accurately identify the image.

1.6. Contributions

Open AI

2. Task 4

2.1. Introduction

This section deals with VAEs.

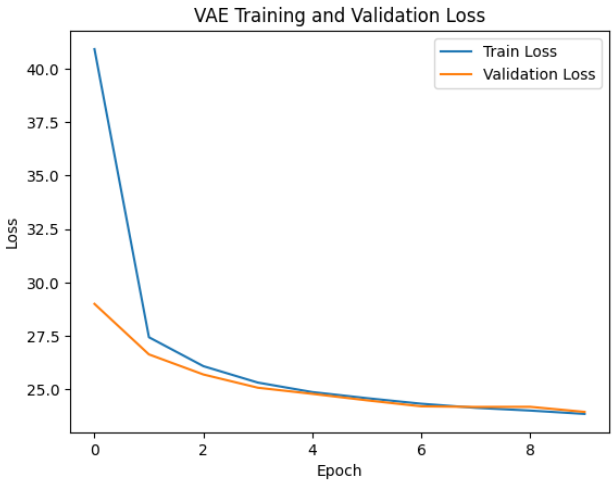
2.2. Methodology

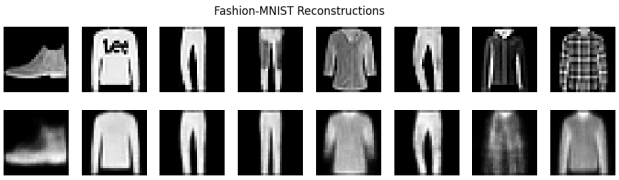
I used the provided architecture.py file and trained my model on the FashionMNIST dataset from torchvision.

The latent dimension was set at 20. The model was trained over 10 epochs.

Next, I tried to have the model decode and then reincode the images to test how well the encoder and decoder had learned.

2.3. Results





The model was able to learn very well. The recorded loss dropped from 40.9 to 23.9 over the course of 9 epochs.

Reconstructing the images yielded good results. The images were a bit more blurry than the original images.

2.4. Discussion

The model was able to learn well, as seen by the dropping loss values.

The decoder however, was still somewhat lacking as it made the reconstructed images a bit blurrier that they out to be.

2.5. Conclusion

VAE good.

References

3. Task 5

3.1. Introduction

This part explores the modularity gap in CLIP

3.2. Methodology

The STL-10 dataset from torchvision was downloaded and used.

Zero shot accuracy was evaluated using three different prompting techniques:

Plain

Photo (”a photo of x”)

Descriptive (”a centered photo of a small x”)

Next, I visualised the distributions of text and image imbeddings to identify the modularity gap

Lastly, I tried to align the modularities using orthogonal Procrustes transform.

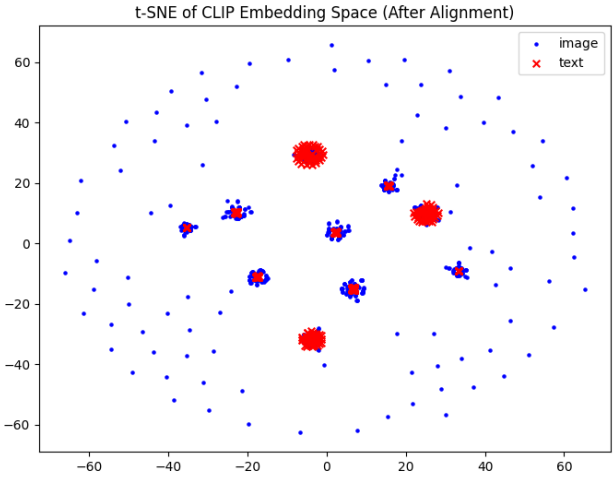


Figure 3. Enter Caption

3.3. Results

The zero shot accuracies came out as:

Plain: 0.963

Photo: 0.974

Descr.: 0.962

Indicating that the model predicted very well.

Next, we got a plot of the imbedding of the image and the text.

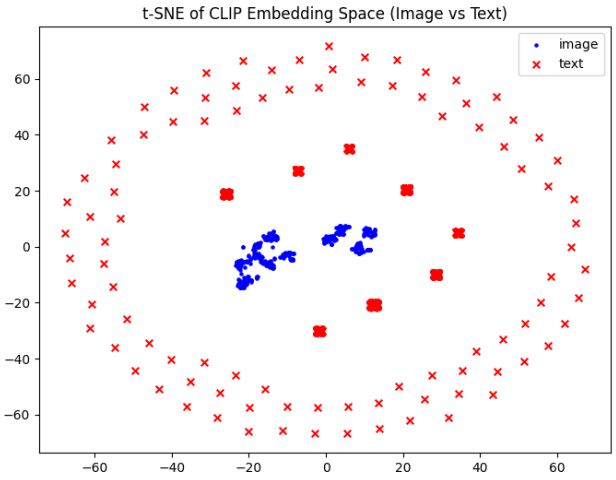


Figure 4. Enter Caption

We aligned it using orthogonal Procrustes transform and got an new Zero-shot accuracy after alignment of 0.977

3.4. Discussion

Through a plot of CLIP’s text and image imbeddings, we see that the imbedding of the image is withing the imbedding of the text. There is, however, a modularity gap, as can be seen in the image.

We used orthogonal Procrustes transform to reduce the modularity gap. The new plot of the embeddings shows this as quite successful

3.5. Conclusion

We were able to confirm that clip embeds the tax and images in the same space. By using the orthogonal Procrustes transform, we were able to further refuce the modularity gap between the embeddings of the text and the picture