

Unsupervised Learning for Imganet

Divya Joseph

Zhuocheng Xu

Abstract:-

We propose an unsupervised task to achieve better generalisation. The task would be to focus on salient features in the image and help our models learn between what to focus on in data and what not to. This has been achieved by training on more samples in an unsupervised way.

Overview:-

With the advent of deep learning the question of what our models are learning is of utmost importance. Is it learning to mimic the data seen during training or it is learning important features from the data is a question we seek to answer. For example if we take a grayscale image of a flower our human eye would be able to still classify it correctly. It's said that we as humans learn and learn to generalize in an unsupervised fashion growing up. In the course of this project we will analyse if this could be true for deep learning models as well.

How we would generate data:-

We would be using Imagenet data for training. First, we generate a black-white version for each image X , then pass the transformed image through a colorizer (Zhang, etc. Colorful Image Colorization) to generate a batch of colorized images X' .

Usability:-

1. Using this during training would ensure we build a robust model and hope to improve classification score
2. As this task is unsupervised we would be using the existing data and no additional labeled data would be required during training.

How to train the data:-

Task:- (from image to representation - Encoder)

We would want to use a U-net and build good unsupervised representation from X . X' and X would be passed through encoders and the final representation formed at the end of it would need to be closer in the vector space. This would be the loss objective. Iterative training using multiple X' would help form unsupervised representation which focuses on salient features. This improves the model by avoiding being distracted by irrelevant features such as color in this project.

Task:- (from representation to label - Decoder)

We then train the model to use the representation to generate corresponding labels.

How to classify data:

When a new image comes, we first use encode to extract its representation and then use decoder to do the classification.