Data Free Model Extraction

Experiment

Based on the paper Data-Free Model Extraction https://arxiv.org/abs/2011.14779 by Truong et al which was for images in 10 classes, we use a generative model to generate examples of videos of 400 classes from which the student can learn. The student acts as a discriminator trying to get similar predictions as the predictions of the teacher when the generated image is given as input to both. Generator and Student are alternately learned. The backpropagation from the prediction loss (KLDiv or L1) was done by gradient approximation through the teacher.

Intuition

If we input random noise through the teacher, it gives mostly the same labels. We need a way to generate images that give different labels and are difficult to classify which is possible using a GAN based approach. The generator tries to generate images that will be difficult for the student to classify. This is encouraged by the L1 or KL Divergence loss function which the optimizer tries to maximize when learning the generator parameters. The student on the other hand tries to minimize the loss to better fit the teacher's predictions. Since the teacher is a blackbox, we cannot backpropagate through it. For that, we do gradient approximation using the Forward Differences method.

Methodology

We used the Data Free Model Extraction (DFME) paper's repository and modified it to work for videos. The Generator was replaced with Video Gan. We first conducted experiments before going to video experiments where we used the DFME with the Image Generator and tried to predict the class assuming it to be a single frame for the video for Swin-T and an image for the student MobileNet and tried to train for the 400 classes of Kinetics-400. We used various combinations of learning rate for student and generator and number of approximation points for the gradient approximator and changing the ratio between steps of generator and student

How to run

1. Install all requirements ```pip install -q -r requirements.txt```
2. Download the weights of swin-t or movinet
3. Modify the config/params.yaml file to use the options as desired and then ```python train_threat.py```
OR
Create a new config file to use the options as desired and then ```python train_threat.py --config path-to-config-yaml```

Properly set the path of datasets and model checkpoint path in the config.