CS F425 Deep Learning: Major Project

Department of Comp. Sc., BITS Pilani, Goa Campus

November 21, 2021

1 Logistics

1.1 Groups

This assignment is to be completed in groups. Please fill in your group details in this sheet. Maximum group size is 4, and Minimum group size is 2.

1.2 Deadline

11:59PM IST on Tuesday 7th December 2021.

1.3 Submission

The link for the submission will be put up on Google Classroom. Please make only one submission per team. The names, BITS ID and contributions of each team member must be included. You can submit either an .ipynb file or a link to a notebook hosted on Colab, Kaggle or other similar platforms. You can also submit a PDF file summarising your work. This is optional, however if you are only submitting a notebook, ensure that it is fully documented and contains the required plots / visualisations / diagrams etc..., you will be evaluated on the presentation of your work.

2 Problems

Select one of the following problems to work on:

2.1 Generative Audio Network

2.1.1 Background

A deep generative model is a special neural network structure that can be used to "generate" data instances. Based on how the model was constructed, it can be either used to generate pairs of (observation $\mathbf{x}$, target y) or only observation $\mathbf{x}$. In these cases, the generative model learns to construct a probability distribution $p(\mathbf{x}, y)$ or $p(\mathbf{x})$. The former model can also be used to generate samples conditionally, that is, conditioning the sample generation based on some target y.

Generative models have traditionally been used in fields involving image data and it's variants. But recent developments have led to generative models being utilised in fields

such as voice impersonation, music synthesis and even speech recognition.

2.1.2 Dataset

For this assignment, you are recommended to use the GTZAN Dataset available at Kaggle to train the generative model. This dataset contains music of 10 genres with 100 samples each of length 30 seconds.

You are also allowed to use alternative open access music datasets such as MusiCLEF or Magnatagatune. But processing alternative datasets and ensuring that only samples from the genre "classical" will be generated will be your responsibility.

2.1.3 Task

Your task is to create a generative model (either a GAN or a VAE) that would generate an audio clip based on an input parameter that specifies the genre of the audio clip. For testing purposes, we shall be considering samples of the genre "classical", as they are the easiest to generate[as they do not have any vocals.]

You will be evaluated based on the following (other than the general guidelines mentioned in Section 3):

- The audio sample generated. Specifically on parameters such as how much noise is present, the quality of the generated audio and most importantly the genre of the audio.
- In case proper audio samples aren't generated, the quality of code and whether it successfully executes, and rationale behind the code will be tested

2.1.4 Resources

The following resources should help in understanding relevant concepts.

- Blog site on how to classify the GTZAN dataset. Useful for data processing and visualization of the dataset.
- Previous work in the music synthesis using GANs.

2.2 Self- and Semi-Supervised Learning

2.2.1 Background

Supervised learning approaches are very good at learning from labelled data. However obtaining labels is often resource intensive, meaning that labelled data for a particular task maybe scarce. On the other hand, there is often an abundance of unlabelled data - just think of the millions of images openly available on the internet. Hence the questions: are there ways in which we can learn from this vast resource?

This might be counter intuitive at first, with the most obvious question being - without labels, what are we even learning? For example, with a dataset of labeled images of animals, in the usual supervised learning setting, the labels give meaning to what our

model is learning, e.g. we can learn an animal classifier. But when we have a dataset of animal images without any label, there is no way to learn such a classifier. Does this mean that we can't learn anything useful?

This is where the idea of learning representations comes in. While we don't have any labels to build a specific classifier, we can still model patterns and relationships within the dataset by learning useful representations. This is known as self-supervised representation learning.

The specific type of self-supervised learning we will be exploring is Contrastive Unsupervised Learning, which is based on a simple idea - representations of similar input data should be similar and representations of dissimilar data should be dissimilar. Since our representations will be real valued vectors in $z \in \mathbb{R}^n$, *similarity* can be simply thought of as distance under some metric between the vector (e.g. $||z_1 - z_2||$). At this stage, we need to answer two questions

1. How do we know which images are similar if we don't have labels?
2. Even if we knew which images are similar, how can we learn a model to produce the appropriate representation?

Most approaches tackle the first question by using data augmentation techniques to create multiple versions of the same image. All of these versions (from the same image) are treated as similar when training the model to produce appropriate representations. For example additional similar images can be generated from a single images by taking random crops, adding noise, flipping, rotating, distorting color etc...

Once the set of similar images are obtained, most approaches follow a similar pipeline. Lets say for an image x, you have obtained two augmented versions x_1 and x_2. We obtain embeddings of these by passing them through the model you want to train: $z_1 = f_\theta(x_1)$ and $z_2 = f_\theta(x_2)$). Our aim in contrastive representation learning is to make representations (or embeddings) of similar inputs similar so ideally z_1 and z_2 should be the same. We can train our model towards this by minimising an appropriate loss $\mathcal{L}(z_1, z_2)$ through gradient descent on θ. Different approaches use different losses and also add particularities to make this general pipeline work better.

Barlow Twins [2] is contrastive self-supervised representation learning approach that learns to make the cross-correlation matrix between two augmented versions of the input close to the identity. The goal is to keep the representation vectors of augmented versions of one sample similar, while minimizing the redundancy between them. An overview of the pipeline is given in Figure 1.

2.2.2 Dataset

For this assignment you will be using the STL-10 dataset [1]. It consists of two parts. The labeled part consists of ten classes of images with 500 training and 800 test images per class. The other part consists of 100000 unlabeled images. More information about the dataset as well as the links to download it can be found at this link.

2.2.3 Task

Your task is to:

1. Train a neural network (architecture details upto you) in a supervised manner on the training portion of the labeled part of STL-10.

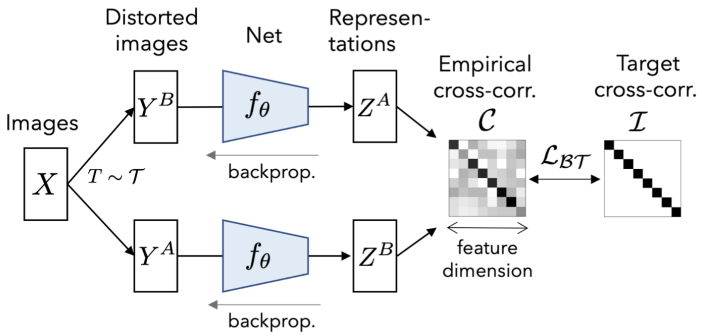

Figure 1: Barlow Twins Overview

2. Use the Barlow Twins method to train a neural network on a subset of the unlabelled part of STL-10 to produce appropriate embeddings. For this you should read through [2]. While you don't need to understand the complete paper, it will give you helpful context and it also has PyTorch style pseudocode which may help you write your own implementation. The size of the subset along with the particular architecture of the neural network is upto you.

3. Visualise how your representation space evolves during training in the self-supervised approach from Step 2. You are free to do this in any way you feel appropriate. The goal should be to convey the changes to the representation space in an intuitive way. (Hint: t-SNE).

4. Learn a linear classifier on top of the embedding network learnt in Step 2(whose weights should be frozen) using the training portion of the labeled part of STL-10 .

5. Having trained a fully supervised and a fully self-supervised model, consider a case where your training data consists of some mix of unlabeled and labeled data from STL-10. Formulate a way to combine both the supervised and self-supervised training procedures to make use of such a dataset to train a classifier.

6. Compare the performance of of the classifiers from Steps 1, 4 and 5.

7. Bonus: Study how the ratio of labeled and unlabeled data affects performance in Step 5.

2.2.4 Resources

The following resources should help in understanding relevant concepts. Please let us know if you find any useful resources not part of the list.

- Lilian Weng's blogs on Self-Supervised Learning and Contrastive Learning. The latter's section on Parallel Augmentation is especially relevant to the task and contains an explanation of Barlow Twins.

- Original Work on Barlow Twins. Note that while self-supervised learning may seem intimidating at first glance, the concepts involved are surprisingly simple. The paper contains pseudocode for the approach which should make the pipeline quite clear and provide a base for a PyTorch implementation.

- STL-10 Homepage.

3 Evaluation

This assignment counts for 20% (20 marks) towards your final grade. You will be evaluated based on the following:

- Quality of code and whether it successfully executes. A self-explanatory Python notebook will do the job. Of course, it has to run flawlessly :-)
- To what extent you have examined the central idea (through building models and experimentation.
- How you make use of theory taught in class to motivate and design your experiments.
- How well you present your results and analysis both in terms of technical depth and aesthetic.

All code, models and analysis must be your own. No stealing from GitHub. Any kind of plagiarism is strictly prohibited. If these rules are broken, no marks will be awarded to the entire team.

References

[1] Adam Coates, A. Ng, and Honglak Lee. An analysis of single-layer networks in unsupervised feature learning. In *AISTATS*, 2011.

[2] Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow twins: Self-supervised learning via redundancy reduction. In *ICML*, 2021.