MOMENTUM CONTRAST FOR UNSUPERVISED VISUAL REPRESENTATION LEARNING

Contrastive learning (CL):

An image is fed into two types of encoder (basically a Convolutional Neural Network). This encoder performs a set of random data augmentation on the image and provides different segments of the same image. The segments of the same image form the true pair (i.e., input and the target (xi, yi)). CL tries to learn from the similarities and differences of the image.

Usually, there will be a dictionary which contains all the augmented images of a dataset or a batch. When an image is fed into the neural network, it looks into the dictionary and tries to identify the other segment of the image. This is done by learning the similarities and difference between the different segments of various images.

The dictionary consists of keys (i.e) pieces of images, sampled from the data (images) and is represented by an encoder. The unsupervised learning algorithm tries to match the query with its matching key. A query matches with a key only when it is an encoded image of the same image. Designing a large and consistent dictionary will enable the network to learn faster from the differences between the images.

In other words, contrastive learning is a way of building a discrete dictionary on high dimensional continuous inputs such as images.

Contrastive loss:
It measures the similarities between the two samples. The target of the input can vary on the fly and can be represented as data representation computed by a network. Generally, it is the sum over one positive and K negative samples in the current mini batch.

The value of this function will be low when the query matches the positive key and dissimilar to all other negative keys.

Momentum Contrast (MoCo):

The dictionary should be designed in such a way that it improves the performance and makes the model robust.

MoCo introduces a dynamic dictionary with a queue and moving averaged encoder.The encoder is a CNN model which feeds the dictionary with augmented images (different representations of an image).

The dictionary as a queue:

- Contains the sampled data of the current and the previous mini batches.
- The encoded representations of the current mini batch are enqueued and the oldest mini batch are dequeued.

- The size of the dictionary is not limited by the mini batch size and this makes it large.
- Large dictionary helps in learning good features as it covers a rich set of negative samples.

Dynamic dictionary means:
- The keys are randomly sampled from the images.

Slowly progressing key encoder:
- As the dictionary size is large, it is difficult to update the key encoder with backpropagation. Since, the gradients must propagate through all the samples in the dictionary.
- One solution to this, to copy the query encoder to the key encoder by eliminating its gradients. But this solution makes the dictionary inconsistent as the encoder is rapidly changing.
- A momentum update to this method was introduced.

MoCo does not require a customized architecture for a pretext task which makes it easier to transfer features to a number of downstream tasks.

Disadvantages:
End-to-end Mechanism:
- Two different encoders for key and query
- The dictionary size is limited by the mini batch size
- Large mini batch size is limited by the GPU memory size
- In some recent pretext tasks the dictionary size can be made large by multiple positions but it requires special network designs.
- When we use a larger mini batch size, we have to use a linear learning rate scaling mechanism without which the accuracy drops by 2%.

Memory Bank:
- A memory bank consists of samples of all the data.
- A dictionary for every mini batch is drawn randomly from the memory bank and the sample is updated in the memory bank every time it is seen by the encoder.
- This makes the method less consistent as it has to keep track of every sample and the momentum update is on every sample not the whole encoder.

Batch Normalization:
- Leaks information between samples and prevents the model from learning good representations.
- This is prevented by shuffling the BN. The sample order in the current mini batch is shuffled before distributing it to the GPU and they are shuffled back after encoding.

A SIMPLE FRAMEWORK FOR CONTRASTIVE LEARNING OF VISUAL REPRESENTATION (SimCLR)

Components of the framework:

1. Strong data augmentation
2. Learnable non linear transformation between the representation and the contrastive loss
3. Normalized embeddings and an adjustable temperature parameter for the contrastive loss function
4. Larger batch sizes and longer training
5. Deeper and wider networks.

A stochastic data augmentation method:

1. Random cropping followed by resizing
2. Random color distortion
3. Random Gaussian Blur

Composition of data augmentation operations:

- Perform the each data augmentation method individually or in pairs.
- Apply the data augmentation to one branch of the framework to understand the effects of the individual data augmentation.
- One pair of augmentation stands out: random cropping followed by random color distortion.
- Stronger data augmentation improves the linear evaluation of learned unsupervised models.

- A Resnet base encoder is used to extract representations from the augmented data samples
- A MLP (multi layer perceptron) is used between the representation and contrastive loss inorder to amplify the invariant features i.e to improve the quality of the representation.
- Loss function used is NT-Xent loss (Normalized temperature scaled cross entropy loss)

Global BN:

- Aggregating BN mean and variance over all devices during training
- Shuffling
- Replacing BN with layer norm

Evaluation Protocol:

- Linear evaluation protocol where a linear classifier is trained on the top of a frozen base network.

Cons:

- The projection head loses some information about the transformation (color or orientation of the objects) applied that may be useful for the downstream tasks.
- The h contains more information about the transformation compared to the g(h).