Artificial Neural Networks Project

Scene Classification

Artificial Neural Networks

April 18, 2025

1 Overview

The goal of this final project is to assess your knowledge and approach for three widely–used pipelines for representation learning on images:

(1) Vanilla supervised learning (cross-entropy).

(2) SimCLR [3]: self-supervised contrastive pre-training + linear probe.

(3) Supervised Contrastive Learning (SupCon) [4]: label-aware contrastive pre-training + linear probe.

SimCLR is a self-supervised learning method that learns image representations by maximizing agreement between different augmented views of the same image, without using labels. It uses a contrastive loss over a large batch of examples, relying on data augmentations like cropping, color jitter, and blur to generate positive pairs.

Supervised Contrastive Learning extends SimCLR by incorporating label information into the contrastive loss. In addition to contrasting different views of the same sample, it also pulls together all samples of the same class. This encourages intra-class compactness and inter-class separation in the learned embedding space. You can see a comparison of these method in Figure 1.

For further details over each method you can refer to their respective papers cited in this document.

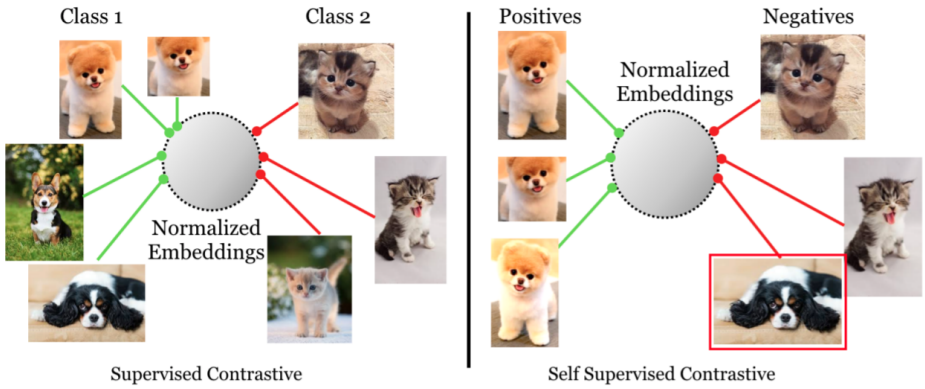

Figure 1: SimCLR and SupCon contrastive learning.

2 15-Scene Dataset

The dataset contains 15 categories of different scenes [1]. The categories are *office*, *kitchen*, *living room*, *bedroom*, *store*, *industrial*, *tall building*, *inside cite*, *street*, *highway*, *coast*, *open country*, *mountain*, *forest*, and *suburb*. The dataset has been divided into two parts train and test. Each part has equally 15 different classes of scenes. The train set is used during the training process in order to "teach" the model how to classify images. The validation set is used to evaluate the model after each epoch, it is not seen by the model during training. You will find the dataset in the BlackBoard platform, located at *Artificial Neural Network, Project-Self-Supervised Scene Classification > 15-Scene.zip*.

3 Implementation

- To ensure a fair comparison, use a single ResNet-18 architecture as the backbone across all training strategies.
- For both SimCLR and SupCon, employ a single linear layer on top of the encoder for classification (i.e., linear probing).
- Include `embedding.py` script to visualize the test embeddings in 2D space, apply the t-SNE (t-distributed Stochastic Neighbor Embedding) [2]technique.
- The implementation must be done in `.py` files (not Jupyter notebooks), with clear and informative comments throughout the code.

- Include an `evaluation.py` script that outputs classification accuracy for all methods when executed.

4 Deliverables

Submit a **PDF report** that addresses the following points:

1. Create a table summarizing the training settings for each model, including learning rate, optimizer, batch size, number of epochs, and the number of fully connected layers. Justify the choice of each setting.
2. Use t-SNE to visualize how test embeddings from different classes are distributed in 2D space. For SimCLR and SupCon, use the embeddings obtained before the linear classification layer. Interpret and justify your observations.
3. Compare the performance across the three approaches (Supervised, SimCLR, and SupCon). Answer the following: Which model achieves the highest classification accuracy? What are the advantages of using SimCLR or SupCon over supervised training?
4. Describe any strategies you used to address underfitting or overfitting issues. Provide evidence to support your claims (e.g., plots, accuracy curves, etc.).

Submit both your code and PDF report in a zipped folder named as $Lastname\_FirstName\_StudentID.zip$ **before 31st of May**.

References

[1] Svetlana Lazebnik, Cordelia Schmid, and Jean Ponce. "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories". In: *2006 IEEE computer society conference on computer vision and pattern recognition (CVPR'06)*. Vol. 2. IEEE. 2006, pp. 2169–2178.

[2] Laurens van der Maaten and Geoffrey Hinton. "Visualizing Data using t-SNE". In: *Journal of Machine Learning Research* 9.86 (2008), pp. 2579–2605. URL: `http://jmlr.org/papers/v9/vandermaaten08a.html`.

[3] Ting Chen et al. "A simple framework for contrastive learning of visual representations". In: *International conference on machine learning*. PmLR. 2020, pp. 1597–1607.

[4] Prannay Khosla et al. "Supervised contrastive learning". In: *Advances in neural information processing systems* 33 (2020), pp. 18661–18673.