Application of contrastive learning on 3D shapes point clouds

September 11, 2022

Matteo Gioia

Abstract

The necessity for large quantities of labelled data in deep learning encouraged the exploration of techniques to extract meaningful information with minimal supervision. **Contrastive learning** is one of these **self-supervised** techniques, where the core concept is to learn representations (embeddings) which can act as **pseudo-labels** for (simpler) downstream models. The goal of this project is exploring its **application to non euclidean data** in the form of 3d shapes.

1. Introduction

The necessity for large quantities of labelled data in deep learning encouraged the exploration of techniques to extract meaningful information with minimal supervision. **Contrastive learning** is one of these **self-supervised** techniques, where the core concept is to learn representations (embeddings) which can act as **pseudo-labels** for (simpler) downstream models.
Contrastive learning builds on top of previous self supervised techniques, such as **pretext task learning**, in which a model learns to extract features by solving a **pretext task**, like predicting the degree of rotation of an image (Fei-Fei Li). However, choosing the right pretext task is really hard and heavily impacts the quality of the representation learned.

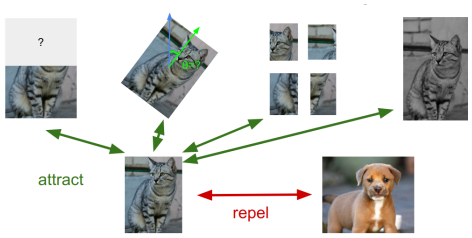

Figure 1. Basic idea behind contrastive losses (Fei-Fei Li)

Email: Matteo Gioia <gioia.1995989@studenti.uniroma1.it>.

Deep Learning and Applied AI 2022, Sapienza University of Rome, 2nd semester a.y. 2021/2022.

Contrastive learning tries to avoid this by defining a more ”general” pretext task (Fei-Fei Li): enforcing the model representations of original samples and their augmentations (**positives**) to be similar while also being ”dissimilar” from other samples and their augmentations (**negatives**). This makes contrastive learning a lot more flexible, as it allows to define multiple augmentations w.r.t. to which the model can be made, in a certain sense, invariant. Effectively, this result is achieved with the use of a **contrastive loss**. This process ultimately generates **pseudo-labels** that can be used for a different downstream task, such as classification. The pipeline of a contrastive learning regimen can be briefly described as follows:

1. **augmentation**: the original data is transformed with the chosen augmentations;
2. **contrastive training**: in this phase, the model generates the representations and is optimized w.r.t to the contrastive loss;
3. **downstream task**: the representations generated by the model are used for a different task (e.g. classification) to evaluate their quality.

Interestingly, advancements in research on contrastive techniques introduced new models that deviate from the original concept: some for instance use multiple encoders or memory banks, others do not need negative samples at all. (Jaiswal, 2021)

This project The goal of this project is to apply a contrastive learning regimen to 3D shapes transformed into point clouds, in order to compare the result with standard supervised models.

2. Related Work

Literature regarding contrastive learning on non euclidean data has grown significantly in the last few years, as is possible to see here `https://github.com/cshjin/GCL`. Most of it, however, is focused on applying contrastive learning on graphs and, while it could be argued that models like PointNet effectively elaborate a graph of

3D shapes (by sampling points), it is not clear whether the same augmentations used for graphs would still be effective. For instance, adding multiple nodes or edges could really deform the shape examined, creating for an augmentation similar to perturbing a graph.

3. Methodology

The setup for testing the efficacy of the contrastive regimen is as follows:

- **pre-processing**: each training dataset was modified to add 1 to 3 rotated versions of the original shape, in order to be used as positive/negative samples. The shapes were then sampled to create point clouds.
- **training**: PointNet network with 2 PPFNet (Deng et al., 2018) layers followed by a projection head of 2 FC layers and optimized with an INFONCE loss. This network uses 4D descriptors in order to process correctly the rotation.
- **testing**: 1 layer FC neural network, optimized with Cross Entropy loss.
- **environement**: Google Colab + Pytorch Geometric (Fey & Lenssen, 2019).

This model was tested on the *Geometric Shapes* and *ModelNet 10* (Z. Wu, 2015) datasets, although the latter had to be shrunk in order to be executed on Colab. Also note that for *Geometric Shapes* only 256 points allowed the model (when sampling the 3D shape) to reach convergence of the INFONCE loss, while 1024 were necessary to make it stable and quicker when using *ModelNet 10*.
For what regards the baseline used to compare results, the same PointNet architecture was used, optimized with a Cross entropy loss and thus modified with a final fully connected layer outputting the logits for each class.
Here `https://github.com/MatteoGioia/DLAI_Prj2022` is a link to the full jupyter notebook with the code, graphs and more.

4. Experimental Results

The following results are extracted from the best runs of each testing round (e.g. with 100 or 250 epochs) for both models.

Dataset	Contr. Pointnet	Sup. PointNet
Geometric Shapes	85%	92.5%
ModelNet10	60%	50.6%
ModelNet10 **10 labels**	44%	42%

Table 1. Final accuracy of each model

In the case of *Geometric Shapes*, the supervised model performed 7.5% better in its best run, although it's highly possible that it was just over fitting the relatively small dataset (more epochs lead to better accuracy each time, so it's possible the network was just memorising the dataset). For what regards *ModelNet 10*, instead, the contrastive model performed almost 10% better in its best run using all the possible training data, while only using 10 labels per class still yielded a 2% improvement. This could be the sign of a better generalization achieved by the model, which was able to generate more descriptive labels. However, when tuning the classifier for the contrastive model, its accuracy seemed to oscillate continuously between 50% and 60%, without stabilizing, as shown in Figure 2. This could mean that while the labels carry some extra information (since the InfoNCE loss was correctly minimized) the implemented architecture was not complex enough to take advantage of it.

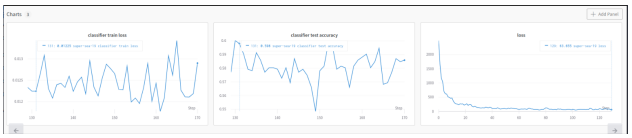

Figure 2. Accuracy of final classifier

5. Conclusions

Although the improvements shown in the experiments are minor, they still suggest that contrastive learning can be used to **potentially improve** the performance of already existing models. It would then be interesting to implement more complex architectures (w.r.t. the implemented one) and explore the results in a contrastive setting, even with different losses and (contrastive) architectures. Exploring more augmentations might also be considered, for example removing faces from a figure or deforming it partially, in order to build networks "invariant" to such modifications yet able to create "descriptive" labels.
In general, this project highlights how flexible this paradigm is, making it is possible to define augmentations "more simply" (w.r.t. to traditional pretext learning) and obtain a network that performs on par with supervised implementations or even better.

References

Deng, H., Birdal, T., and Ilic, S. Ppfnet: Global context aware local features for robust 3d point matching. *CoRR*, abs/1802.02669, 2018. URL `http://arxiv.org/abs/1802.02669`.

Fei-Fei Li, Jiajun Wu, R. G. Lecture 14: Self-supervised

learning (2022). http://cs231n.stanford.edu/slides/2022/lecture_14_jiajun.pdf.

Fey, M. and Lenssen, J. E. Fast Graph Representation Learning with PyTorch Geometric, 5 2019. URL https://github.com/pyg-team/pytorch_geometric.

Jaiswal, Ramesh Babu, Z. Z. e. a. A survey on contrastive self-supervised learning. *Technologies 2021*, 2021.

Z. Wu, S. Song, e. a. 3d shapenets: A deep representation for volumetric shapes. *Proceedings of 28th IEEE Conference on Computer Vision and Pattern Recognition*, 2015.