The Details Matter: Preventing Class Collapse in Supervised Contrastive Learning

Elouan Gardès Charles Monté

Abstract

Contrastive pre-training is a technique that aims at producing rich representations from data, that can then be used for a variety of downstream tasks later-defined. When dealing with supervised contrastive learning, class-collapse may occur, that is all labels are encoded to a very small region of the latent-space, leading to poor un-informative representations. While this is ok for classification based on the same labels, collapse prevents any other application of the learnt representations, such as sub-classes or strata recovery. In this work, we describe and analyze a method to deal with collapse proposed by *Fu et al. (2022)*. Our analysis ranges from theoretical understanding and extensions through experiments.

We discuss the theoretical underpinnings of the method, its implications, and the experimental results presented in the paper.

We show that the proposed L_{spread} loss function effectively prevents class collapse by promoting intra-class diversity in the embedding space. We get to improve the results of the paper with the use of pre-trained weights.

We also explore the limitations of the proposed approach and highlight inconsistencies in the results, mainly caused by difficult reproducibility.

We finish by proposing potential extensions to the method that could further enhance its performance and applicability.

Contents

1. Introduction

Supervised contrastive learning, a method derived from contrastive learning, has emerged as a powerful method for training deep neural networks over recent years, particularly in representation learning. Supervised contrastive learning enhances the contrastive learning process by leveraging class labels to further improve the quality of learned embeddings. The main objective is to push together embeddings from the same class while pulling apart embeddings from different classes.

However, despite its effectiveness, supervised contrastive learning faces a significant issue known as class collapse. Class collapse occurs when embeddings from all samples within the same class become indistinguishable, essentially collapsing into a single point in the embedding space. This phenomenon occurs because class collapse allows to minimize the contrastive loss but it results in the loss of important information that could have been used for downstream tasks. For instance, data from the same class may contain different semantic attributes, such as poses, textures, or backgrounds, which are not captured by the class label alone. When these intra-class variations, or "strata", are ignored, the model's ability to generalize to more complex tasks becomes limited.

To address the problem of class collapse, Fu et al. introduce a new supervised contrastive loss function called L_{spread}. The proposed approach aims to preserve intra-class variability by not only pulling apart embeddings from differ-

ent classes but also uniformly spreading apart embeddings within the same class. By doing so, the method discourages class collapse and maintains more detailed representations of the underlying strata. L_{spread} is directly based on the SupCon loss Lsc with an added term specifically made to avoid class collapse.

Not only does the paper provide theoretical evidence that the L_{spread} loss would prevent class collapse by promoting intra-class diversity in the embedding space, it also provides empirical evidence showing that this loss function leads to improved results in several challenging downstream tasks. The authors demonstrate that their approach not only produces higher-quality embeddings but also enhances overall model performance on multiple benchmarks, offering a significant step forward in the development of more robust and flexible contrastive learning techniques.

2. Motivations, related work and applications

We provide a few motivations for the original work studied here, as well as some introduction to relevant concepts for analysis and critiques.

2.1. Motivations

What makes a contrastive learning algorithm efficient is the quality of the learned embeddings. High quality embeddings will perform better on all the downstream tasks. While a majority of the tasks only revolve around inter-class variations, some specific tasks such as subgroup identification require the embeddings to also retain information about intra-class variation. This is where using methods to prevent class collapse become useful. It is also shown that embeddings with good strata information allow for a better understanding of the whole data repartition, which allows for instance to correct noisy labels with a better performance than normal embeddings.

2.2. Referenced work

SUPERVISED CONTRASTIVE LOSS (SUPCON)

The Supervised contrastive loss (SupCon) extends the principles of unsupervised contrastive learning by leveraging class labels to create positive and negative pairs. While SupCon has shown strong performance improvements over traditional supervised learning techniques, it suffers from class collapse.

SIMCLR

SimCLR *(Chen et al., 2020)* is a popular unsupervised contrastive learning method that does not rely on labels but instead uses data augmentations to generate positive pairs. While SimCLR has demonstrated impressive results, its lack of label supervision can make it less effective than its supervised counter-parts when we do have labels at our disposal. SimCLR may not capture class-specific information as effectively as supervised approaches like SupCon or the proposed L_{spread}. The key difference is that SimCLR only ensures that augmented versions of the same image are mapped closely, without explicitly enforcing separation between different classes.

INFONCE LOSS

The InfoNCE loss *(van den Oord et al., 2018)*, commonly used in contrastive learning methods like SimCLR, is designed to maximize the mutual information between different views of the same data point while pushing apart representations of different points. In the context of supervised contrastive learning, a variation of InfoNCE can be applied to encourage separation between individual samples within the same class, preventing class collapse. The authors of the studied paper build on this idea by incorporating InfoNCE into their L_{spread} loss, which adds a repelling force between points of the same class to ensure that embeddings are more spread out. This helps retain intra-class diversity, which is critical for downstream applications such as fine-grained classification and robustness to noisy data.

Applications

The proposed L_{spread} loss function has broad applications across various domains, particularly in tasks that require fine-grained classification, robustness to noisy data, and transfer learning. By preventing class collapse and preserving intra-class diversity, L_{spread} can enhance the quality of learned embeddings and improve model performance on challenging downstream tasks. Some specific applications include:

PRETRAINING:

L_{spread} can be used to pretrain models on large-scale datasets, enabling them to learn more robust and detailed representations of the data. Training supervised contrastive models with L_{spread} produces embeddings that transfer well across tasks by capturing not only the class-specific information but also finer-grained features that may be relevant for the downstream tasks. For example, embeddings trained on a dataset of animal images may be fine-tuned to a specific task of classifying different species or even different poses of the same species. This capacity is especially well demonstrated on the coarse-to-fine classification tasks described in the experiments section of this report.

ROBUSTNESS OF THE EMBEDDINGS

In real-world datasets, classes may contain noisy,mislabeled or unlabeled samples. L_{spread} helps prevent this by encour-

aging the embedding space to give a better representation of the data points, not only thanks to their class, but also thanks to their intrinsic features. A good example of this robustness, outside of the examples on noisy data, is the capacity of such models to discover unlabeled strata, which is a really useful feature when building minimal coresets.

CORESET CONSTRUCTION

Coreset construction refers to selecting a small, representative subset of the training data that can be used to train models efficiently without significantly compromising performance. The implementation of the L_{spread} loss plays a role in this by identifying which samples in the embedding space are most representative of the overall dataset. This can be particularly useful in scenarios where computational resources are limited, as it allows for more efficient training while maintaining model accuracy. In the paper, the authors demonstrate how their proposed Lspread loss can be used to identify important examples for coreset construction, ensuring that the selected points capture the essential intra-class variations needed for optimal training.

3. The proposed method

3.1. Theoretical overview of the L_{spread} loss

The proposed L_{spread} loss function is designed to prevent class collapse by promoting intra-class diversity in the embedding space. The loss function is based on the SupCon loss, however, the SupCon can lead class collapse, as it only focuses on inter-class separability. To address this limitation, the authors introduce InfoNCE loss term using labels, to have repelling force between points of the same class, ensuring that L_{spread} embeddings are more spread out, capturing the underlying strata and preserving intra-class diversity. The two components mathematically work like this:

- L_{attract} is the component directly based on the SupCon loss. It uses a variation of the loss that encourages class separation in embedding space. This term would be the standard loss in a normal Supervised Contrastive Learning problem. It is mathematically defined just below where $P(i, B)$ represents all the points in B (the batch) with the same class as the anchor point x_i while $N(i, B)$ represents all the points in B with a different class from x_i.

$$L_{\text{attract}}(f, x_i, B) = -\frac{1}{|P(i,B)|} \times \sum_{p \in P(i,B)} \log \frac{\exp(\sigma(x_i, x_p))}{\exp(\sigma(x_i, x_p)) + \sum_{a \in N(i,B)} \exp(\sigma(x_i, x_a))}$$

- L_{repel}, is the component inspired by the InfoNCELoss. It is in fact an adaptation of the InfoNCE loss that works inside of a class. It in a sense creates a self-supervised contrastive problem inside of each class, which encourages the points within a class to be spread apart and to stick with points that are very similar, encouraging strata to be together. It is defined below where x_i^{aug} is an augmentation of x_i and $P(i, B)$ represents a set of augmentations of the x_i anchor.

$$L_{\text{repel}}(f, x_i, B) = -\log \frac{\exp(\sigma(x_i, x_i^{\text{aug}}))}{\sum_{p \in P(i,B)} \exp(\sigma(x_i, x_p))}$$

The loss is finally formulated as a weighted loss of the two components above:

$$L_{\text{spread}} = \alpha L_{\text{attract}} + (1 - \alpha) L_{\text{repel}}$$

Where α is a hyperparameter that controls the amount of regularization applied. The optimal α value allows the embeddings to capture both inter-class separation (through L_{attract}) and intra-class spread (through L_{repel}), leading to more detailed and robust representations of the data.

3.2. Theoretical fundations

The theoretical development and analysis of the proposed method relies on 3 main assumptions made in the paper:

1. The encoder output space is restricted to the unit hypersphere $\mathbb{S}^{(d-1)}$. This is made to simplify the geometry of the embeddings, which is helpful for training as the distance between between points can be easily controlled using cosine similarity but also during the latent space analysis, as it encourages uniform spread of the embeddings and ensures that they are bounded. This assumption is common practice and leads to overall good mathematical fundations and empirical results while not really limiting the flexibility of the encoder: it is really rare to find real-world which has a distribution that cannot be constrained to the unit hypersphere, which means that the small loss in flexibility is compensated by an easier and helpful representation of the latent space.

2. The embedding dimension d is larger than the number of classes ($K \leq d + 1$). This assumption directly comes from *(Graf et al., 2023)* assumptions to be able to recover optimal embedding geometry in supervised contrastive learning tasks. The only restriction that this assumption creates appears when tackling tasks with a very large number of classes, such as ImageNet *(Deng et al., 2009)*. Ensuring this condition requires a much higher embedding dimension to avoid class overlap but restricts the adaptability of the method as it may require

models with an higher number of parameters, which necessit larger training capacities. Overall, as a majority of the SOAT encoder models have an embedding dimension of over 2048, this assumption is realistic and works well, but can restrict the reproducibility of the results. A downside of this assumption we have personnally noticed is not only linked to this assumption but also an empirical observation: not having a large enough batch size causes problems in the loss computation, causing the loss to reach infinite values. This is a problem that can be solved by increasing the batch size, but with an embedding dimension of 2048, the batch size cannot be increased too much because of memory constraints.

3. The encoder is "infinitely powerful", meaning it can learn any mapping on the embedding space. This assumption is the most useful on the theoretical side as it ensure no limitations on the expressiveness of the model, which makes the optimization problem easier and ensures the model will perfectly learn the desired embedding distribution given data and time. However, this is the less realistic of the three assumptions and the one that makes the experiment and the theory diverge a lot. It ignores the main practical issues encountered during AI training such as overfitting, computational efficiency or data poverty which are significant in real-word scenarios and that we encountered during our experiments. What saves in a way this assumption is the good results obtained at the end of the paper, allowing the empirical results to show that the theory goes in the right direction.

Even if restrictive and idealized for some, those assumptions are common throughout the literature and allow for a good theoretical base for further demonstrations and experiments.

3.3. Theoretical implications and demonstrations

The main theoretical implications and demonstrations appearing in the paper are based solely on a thought experiment about how subsampling the dataset can reveal information on the strata and their repartition in the embedding space. Repeating the thought experiment and the arguments would be redundant, but the main idea is that by subsampling the dataset, training an encoder on this subsampled set and analyzing the behaviour of strata, we can derive general ideas about the average distance between two strata. Three cases arise based on the presence of both the stratas in the subsampled dataset:

1. **Both strata are present:** The encoder is trained on both, and the distance between them depends on the embedding geometry, collapsing to zero under certain conditions on α.
2. **One stratum is present:** The distance depends on the out-of-distribution performance and is influenced by differences in distribution and model capacity.
3. **Neither stratum is present:** The distance is bounded by the total variation distance between the distributions of the strata.

Two key insights emerge from these cases:

1. Common strata (frequently appearing in the data) cluster closely as the distance between them depends on the optimal asymptotic distribution, which if $\alpha = 0$ is 0.
2. Rarer or more distinct strata are more widely separated from the common strata, as common and rare strata are separated based on the difficulty of the respective out-of-distribution tasks.

One main criticism that can be drawn from this thought experiment is, once again, the idealistic aspect of the theoretical assumptions made. In case 1 for instance, it is considered that the encoder is trained on infinite data from the two strata. However, it is to keep in mind that those assumptions are made to simplify the problem and try to evaluate the method in the best possible conditions. The empirical results are the ones that will show if the method is efficient or not in real-world scenarios.

From those insights emerge two main theoretical implications:

1. For the coarse-to-fine tasks, it is preferable to have a very large distance between strata, as it reduces the upper bound on the generalization error.
2. For the general problem, the distance between strata of the same class should not necessarily be maximized, but rather optimized while making sure to maximize the distance between strata of different classes.

Both those results results provide good insight on where α should be set in the L_{spread} loss regarding the task that is done by the model. They also show that strata separation is a key aspect of the model's performance and that the L_{spread} loss is a good way to ensure that the strata are well separated in the embedding space.

4. Experiments, reproducibility and extensions

The authors propose a wide variety of experiments to show the extent of the benefits introduced by their new loss. These benefits range from better downstream tasks performance on challenging unbalanced and strata-rich datasets such

as Waterbirds *(Sagawa et al., 2020)*, to richer embedding spaces and more robust retrieval of fine-grained elements in coarse classes.

For this work, we propose to reproduce and extend on their experiments on the particularly challenging Waterbird data. In particular, our aim is to provide insights into the author's loss behavior as well as some specific trade-offs that it introduces. We also provide an improvement to reach even better downstream performances using their technique by starting from checkpoints pretrained on ImageNet.

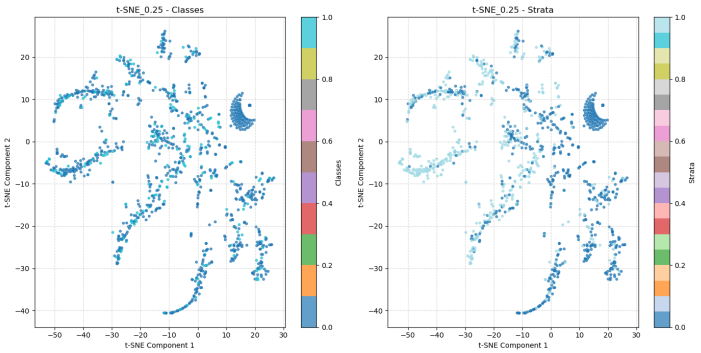
We also comment on the poor reproducibility that the authors allow by lacking to provide substantial or even sufficient details about their experimental settings, let-alone code for specific datasets they introduce. We illustrate the consequences of such poor reproducibility in a spectacular failure of ours to even attempt at extending their results on their CIFAR100-Coarse-U custom dataset.

For all subsequent tests, we place ourselves in the exact experimental settings suggested by the author, that is same model, same technique, same data, same hyperparameters for early reproducibility tests. When no information is given on a crucial part of this setting, we opt for the most straightforward and standard approach.

4.1. Latent-spaces and intuition

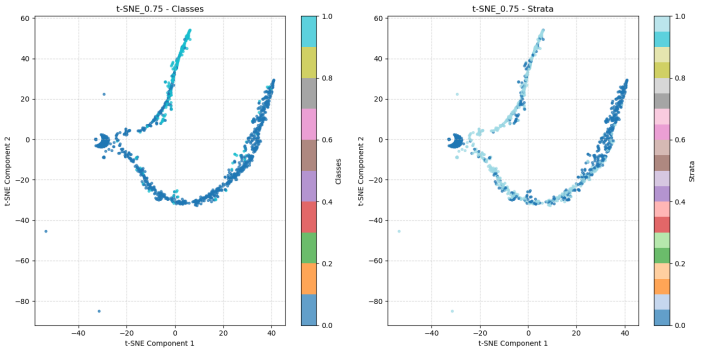
We were able to fully reproduce the author's results on the Waterbird dataset. This dataset is comprised of two classes (land bird, water bird, LB and WB for short) and two strata (land background strata, water background strata, LS and WS for short). The training set is vastly unbalanced in that very few data-points for LB/WS and WB/LS are present. Despite that, strong performance on these samples is critical as the validation and test sets are much more balanced.

In a typical supervised contrastive training setting, class-collapse occur as the encoder learns that WB samples are nearly always associated with WS, and conversely LB samples are nearly always associated with LS. What ends-up happening is thus that the model learns to classify stratum instead of classes, which leads to terrible representations unusable for downstream tasks. Here is, for instance, the latent-space t-SNE-visualization *(van der Maaten & Hinton, 2008)* for very low values of α, i.e for a criterion very close of that of classic supervised-constrastive learning, in the validation set:



We notice how the encoder is unable to produce clusters in terms of class, but it manages to separate stratum rather nicely. What was sufficient cluster classes in the training set (highly correlated stratum) becomes useless on the balanced (in terms of class/stratum combos) validation set. We thus need strong regularization to ensure the encoder does not rely that much on the background strata.

Here now is what we obtain when we resort to much larger regularization, i.e we ask the model to have some intra-class diversity in the latent-space:



This time, we obtain nice clusters in terms of classes, the encoder has learnt to see beyond strata on the training set and is able to consider the right class information on the validation set ; but it still has some good sense of stratum.

We experimented with even higher alphas (close to 1), and since it leads to only asking the model to separate points from the same class, it teaches the model to basically encode samples at random.

4.2. Extension and results

Beyond these high-level considerations on the effectiveness of the proposed method, we wanted to try and improve the author's method simply by testing whether their technique also works if we take a pre-trained model as our starting point. We thus keep the author's setting, but use IMAGENET1K_V2 weights from torchvision *(Marcel et al., 2024)* as a starting point for our ResNet50 encoder *(He et al., 2015)*, and pre-train it with the author's loss at different val-

ues for α. The results we obtain on the validation set are compiled at the top of next page in 1

We compare short-training (S-T, 8 epochs) with long-traning (L-T, 12 epochs) and notice that pre-training benefits from longer trainings whereas baseline struggles with over-fitting in our tests. We provide the validation loss, as well as validation accuracies when using the same idea of linear-probing setting at the author (though we had to guess what prediction-head to use, batch-size, epochs, optimizer, learning-rate and pretty much everything!). We also introduce a straight-forward worst-class accuracy (W-C Acc) metric, simply the performance on the most challenging class/strata combination, in place of the complicated and un-intuitive worst-group robustness used by the author and which relies on a non-practical pipeline of further training and careful cherry-picked adjustments. We want real usable adaptation to new data where prior knowledge of domain-shifts might be lacking and would be the very reason we resort to such regularization techniques.

We demonstrate that resorting to a pre-trained encoder significantly improves downstream accuracies. The delta to classic random initialization is biggest when looking at W-C accuracy. We even thought that we missed something given how un-effective the technique seems to be when looking at these bottom results, but it does work and proves to be reliably better than classic loss on our pre-training results as larger regularization leads to higher raw accuracy as well as W-C accuracy, meaning the latent-space is actually much richer. This leads us to wonder how the authors were able to reach such high W-C Robustness metrics, this is probably hidden in the complicated pipeline that this misleading metric relies on, but absolutely no implementation details or code being given means that we are let to guess.

To finish on experiments on Waterbirds, we want to show some loss-plots from both training and validation. This is motivated by the fact that raw loss seems particularly un-correlated to latent-space richness or downstream performance, as demonstrated in our table.

Here are two plots showcasing the same thing:

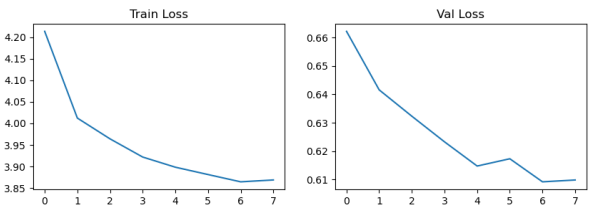


Figure 1. Alpha = 0.25, pre-trained version.

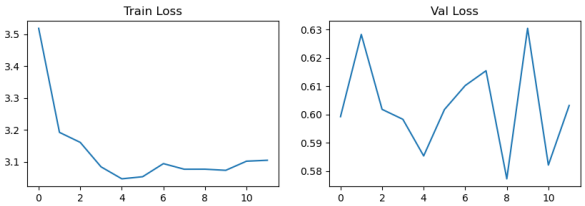


Figure 2. Alpha = 0.75, pre-trained version.

We choose these plots as the first one corresponds to the very best validation loss obtained, and the second one to the very best downstream performances. Despite having low loss, we have very poor performances in the first case, and despite having what seems like complete overfitting and lack of stability in the second case, we reach the very best performances and richest latent-spaces.

Beyond calling to caution when using pre-training losses (especially if we had a self-supervised setting where we would have to partly rely on that!), we are not fully able to give conjectures on this observation. We believe that the way the loss is balanced means that the raw value has little interest when comparing values of α: the sub-tasks induced by the loss are not equally hard and equally punished. This would explain the lack of correlation between validation and downstream accuracy between α values. But even for a given training with a given α, take that last loss plot, the best validation loss corresponds to a model much worse than that obtained at the end of training, despite the loss hike. This effect is one that we are not able to grasp in this work.

4.3. Other experiments

We tried further improvements on the proposed technique. In particular, we tried to adjust α during training, in two ways. First, we tried to schedule its value by testing two opposite hypothesis:

- The model may need stronger regularization at the beginning to avoid collapse. It may then benefit from fully using class information to refine clusters. This leads to increasing α during training.
- The model may need to have a good sense of classes first to retrieve interesting features. It may then benefit from strong regularization to avoid collapse and teach it to look for other information. This leads to decreasing alpha during training.

Both strategies actually performed similarly, but maybe more importantly both strategies failed at improving over the best α using for the whole training. This suggests that there are no obvious phases during training, our model needs

	α = 0.25	α = 0.5, S-T	α = 0.5, L-T	α = 0.75, L-T
Loss	**11.4**	12.2	12.2	12.7
Acc	77	86	87.2	**88**
W-C Acc	0	61.1	45	**70**
Loss	11.9	13.4	13	14.6
Acc	70	57	65	57
W-C Acc	4	7	3	4

Table 1. Top: pretrained base. Bottom: random-init base.

to pay attention to both objectives at all times.

Finally, we experimented on CIFAR100-Coarse-U, a dataset introduced by the authors and obtained by manually unbalancing subclasses from the original CIFAR100-Coarse (extension from CIFAR100 - *Krizhevsky (2009)* inside coarse classes. The subclasses acting as stratum are thus unbalanced like were stratum in Waterbirds, only in the training set.

We were unable to reproduce any results from the authors. We highlight the lack of details regarding training and implementation for these experiments: on top of not giving epochs, optimizer, learning rates, batch-sizes (crucial for contrastive learning), prediction-head or anything of the kind, the authors also failed to give the crucial information of the balance between the loss they introduce and the cross-entropy loss-head used specifically on top of the transformer used in the experiments for this dataset. We of course tried many combos for these hyper-parameters but were not able to come even close to any result given in the original paper.

Unlike the original authors, we provide all our code for reproducibility as well as for potential mistakes in our implementations at the following github repo.

5. Limitations and potential extensions

5.1. Limitations

Other than the reproducibility issues that we have encountered during our experiments, some limitations of the proposed method can be identified:

THEORETICAL ASSUMPTIONS

The theoretical assumptions made in the paper, while useful for simplifying the problem and deriving general insights, may not always hold in practice and even block the use of the loss in certain fields. For instance when looking at medical imaging, a field that would hugely benefit from the use of such a loss, the class balance hypothesis is incompatible with the data, as class imbalance is the norm in medical datasets.

DEPENDENCE ON THE α PARAMETER

One limitation of the proposed method lies in its reliance on the hyperparameter α, which balances the attraction and repulsion forces within the L_{spread} loss. Although the authors provide empirical results, the choice of α is likely problem-specific and requires fine-tuning for optimal performance as shown during our experiment. This dependence on a heuristic parameter limits the method's generalizability across diverse datasets or tasks. Future work could explore adaptive methods to automatically adjust α based on data characteristics. However, while we tried to find such methods, we were not able to automatically tune the α parameter during training.

ROBUSTNESS OF THE STRATA CONCEPT

The concept of strata, while useful for understanding the distribution of data in the embedding space, may be too subjective and difficult to define in practice. Strata are defined as groups of data points that share similar characteristics, such as poses, textures, or backgrounds. However, the boundaries between strata are not always clear-cut, and the definition of what constitutes a stratum may vary depending on the dataset or task. However, this limitation is closer to the thought experiment than to the empirical results themselves. The results show that strata add value in the embedding space, but the concept of strata in itself brings discussions on the subjectivity of the concept.

5.2. Potential extensions

The main extension to the paper we explored was the use of pretrained weights. We have shown that using pretrained weights can greatly improve the performance of the model. This extension is particularly useful but is not the only one that could be done. Other potential extensions include:

EXPLORATION TO MORE COMPLEX IMAGE DATASETS

As discussed just above, while the medical field would greatly benefit from the use of the L_{spread} loss, the class balance hypothesis is incompatible with the data. However, the use of the loss on more complex image datasets could

be a great extension of the paper. The results obtained on the coarse-to-fine classification tasks show that the loss is efficient on complex tasks, and the use of the loss on more complex datasets could be a great extension of the paper. We briefly thought about what could be a great way to simply test the L_{spread} loss on medical imaging datasets, and found some Medical MNIST-type (Yang et al., 2023) datasets that could be used to test the loss. This extension could be a great way to show the efficiency of the loss on more complex datasets.

UNDERSTANDING THE DISTRIBUTION OF α

While it has been shown above that scheduling alpha during training does not yield good results, we did not try to implement other methods that could maybe help the training process to use the loss the most efficiently possible by finding the optimal alpha value alone. This would avoid manual tuning of the hyperparameter and could be a great extension of the paper as it would reduce the training time and the need for human intervention during the training process.

6. Conclusion

In this work, we have provided a critical analysis of a paper proposed by*Fu et al. (2022)*. We have discussed the theoretical underpinnings of the proposed L_{spread} loss function, its implications, and the experimental results presented in the paper. We showed that the L_{spread} loss effectively prevents class collapse by promoting intra-class diversity in the embedding space, leading to improved results on several challenging downstream tasks. We also explored the limitations of the proposed approach, including reproducibility issues, theoretical assumptions, and the dependence on the α parameter. Finally, we propose potential extensions to the method that could further enhance its performance and applicability to more diverse and complex data settings.

References

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. A simple framework for contrastive learning of visual representations. In III, H. D. and Singh, A. (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 1597–1607. PMLR, 13–18 Jul 2020.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255, 2009. doi: 10.1109/CVPR.2009.5206848.

Fu, D. Y., Chen, M. F., Zhang, M., Fatahalian, K., and Ré, C. The details matter: Preventing class collapse in supervised contrastive learning. *Computer Sciences amp; Mathematics Forum*, 3(1), 2022.

Graf, F., Hofer, C. D., Niethammer, M., and Kwitt, R. Dissecting supervised contrastive learning, 2023.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition, 2015. URL https://arxiv.org/abs/1512.03385.

Krizhevsky, A. Learning multiple layers of features from tiny images. 2009. URL https://api.semanticscholar.org/CorpusID:18268744.

Marcel, F., Rodriguez, N., and Contributors. Torchvision: Resnet50 model. https://pytorch.org/vision/main/models/generated/torchvision.models.resnet50.html#torchvision.models.resnet50, 2024. Accessed: 2024-12-04.

Sagawa, S., Koh, P. W., Hashimoto, T. B., and Liang, P. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization, 2020. URL https://arxiv.org/abs/1911.08731.

van den Oord, A., Li, Y., and Vinyals, O. Representation learning with contrastive predictive coding. *CoRR*, abs/1807.03748, 2018.

van der Maaten, L. and Hinton, G. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(86):2579–2605, 2008. URL http://jmlr.org/papers/v9/vandermaaten08a.html.

Yang, J., Shi, R., Wei, D., Liu, Z., Zhao, L., Ke, B., Pfister, H., and Ni, B. Medmnist v2-a large-scale lightweight benchmark for 2d and 3d biomedical image classification. *Scientific Data*, 10(1):41, 2023.