Rethinking preventing class-collapsing in metric learning with margin-based losses

Elad Levi[1], Tete Xiao[2], Xiaolong Wang [3], Trevor Darrell[1,2]
[1]Nexar, [2]UC Berkeley, [3]UC San Diego

Abstract

Metric learning seeks perceptual embeddings where visually similar instances are close and dissimilar instances are apart, but learned representations can be sub-optimal when the distribution of intra-class samples is diverse and distinct sub-clusters are present. Although theoretically with optimal assumptions, margin-based losses such as the triplet loss and margin loss have a diverse family of solutions. We theoretically prove and empirically show that under reasonable noise assumptions, margin-based losses tend to project all samples of a class with various modes onto a single point in the embedding space, resulting in class collapse that usually renders the space ill-sorted for classification or retrieval. To address this problem, we propose a simple modification to the embedding losses such that each sample selects its nearest same-class counterpart in a batch as the positive element in the tuple. This allows for the presence of multiple sub-clusters within each class. The adaptation can be integrated into a wide range of metric learning losses. Our method demonstrates clear benefits on various fine-grained image retrieval datasets over a variety of existing losses; qualitative retrieval results show that samples with similar visual patterns are indeed closer in the embedding space.

1. Introduction

Metric learning aims to learn an embedding function to lower dimensional space, in which semantic similarity translates to neighborhood relations in the embedding space [22]. Deep metric learning approaches achieve promising results in a large variety of tasks such as face identification [5, 44, 43], zero-shot learning [9], image retrieval [14, 10] and fine-grained recognition [47].

In this work we investigate the family of losses which optimize for an embedding representation that enforces that all modes of intra-class appearance variation project to a single point in embedding space. Learning such an embedding is very challenging when classes have a diverse appearance. This happens especially in real-world scenarios where the class consists of multiple modes with diverse visual appearance. Pushing all these modes to a single point in the embedding space requires the network to memorize the relations between the different class modes, which could reduce the generalization capabilities of the network and result in sub-par performance.

Recently researchers observed that this phenomena, where all modes of class appearance "collapse" to the same center, occurs in case of the classification SoftMax loss [30]. They proposed a multi-center approach, where multiple centers for each class are used with the SoftMax loss to capture the hidden distribution of the data to solve this issue. In the metric learning field, a positive sampling method has been proposed [53] with respect to the N-pair loss [41] in order to relax the constraints on the intra-class relations. For margin-based losses such as the triplet loss [3] and margin loss [51], it was believed that they might offer some relief from class collapsing [47, 51]. From a theoretic perspective, we prove that with optimal assumptions on the hypothesis space and the training procedure, it is indeed true that the margin-based losses have a minimal solutions without class collapsing. However, we formulate a noisy framework and prove that with modest noise assumptions on the labels, the margin-based losses yet suffer from class collapse and the easy positive sampling method proposed in [53] allow more diverse solutions. Adding noise to the labels allow modelling both the aleatoric and the approximation uncertainties of the neural network, therefore it batters represent the training process on real-world datasets with fixed restricted network architecture.

We complement our theoretical study with an extensive empirical study, which demonstrates the class-collapsing phenomena on real-world datasets, and show that the easy positive sampling method is able to create a more diverse embedding which results in better generalization performances. These findings suggest that the noisy environment framework better fits the training dynamic of neural networks in real-world use cases.

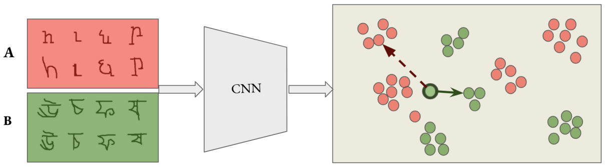

Figure 1: Given an anchor (circle with dark ring), our approach samples the closest positive example in the embedding space as the positive element. This results in pushing the anchor only towards the closest element direction (green arrow), which allows the embedding to have multiple clusters for each class.

2. Related Work

Sampling methods. Designing a good sampling strategy is a key element in deep metric learning. Researchers have been proposed sampling methods when sampling both the negative examples as well as the positive pairs. For negative samples, studies have focused on sampling hard negatives to make training more efficient [39, 35, 48, 28, 29]. Recently, it has been shown that increasing the negative examples in training can significantly help unsupervised representation learning with contrastive losses [12, 52, 4]. Besides negative examples, methods for sampling hard positive examples have been developed in classification and detection tasks [21, 38, 1, 6, 40, 50]. The central idea is to perform better augmentation to improve the generalization in testing [6]. In contrast, Arandjelovic et al. [1], proposed to perform a positive sampling by assigning the near instance from the same class as the positive instance. As the positive training set is highly noisy in their setting, this method leads to features invariant to different perspectives. Different from this approach, we use this method in a clean setting, where the purpose is to get the opposite result of maintaining the inner-class modalities in the embedding space. Using easy positive sampling has been also proposed with respect to the N-pair loss [53] in order to relax the constraints of the loss on the intra-class relations. From a theoretic perspective, we prove that in a clean setting this relaxation is redundant for other popular metric losses like the triplet loss [3] and margin loss [51]. We formulate the noisy-environment setting and prove that in this case the triplet and margin losses also suffer from class-collapsing and using an easy positive sampling method optimizes for solutions without class-collapsing. We also provide an empirical study that supports the theoretic analysis.

Model uncertainty. There are three types of sources for uncertainty: epistemic, aleatoric and approximation [8]. The epistemic uncertainty describes the lack of knowledge of the model, the approximation uncertainty describes the model limitation to fit the data, and the aleatoric uncertainty accounts for the stochastic nature of the data. While the epistemic uncertainty is relevant only in regions of the feature space where there is a lack of data, both the approximation and the aleatoric uncertainties are relevant also in regions where there is labelled data. In this work, we model the approximation and the aleatoric uncertainties, by adding noise to the labels. This noise can stand for the data stochasticity in the aleatoric uncertainty case, or the results of the Bayes optimal model within the hypothesis space in case of the approximation uncertainty. The approximation uncertainty in deep neural networks is considered to be negligible [7]. However, we prove that even a small amount of noise results in a degeneration of the family of optimal solutions in case of the margin-based losses.

Learning with noisy labels is a practical problem when applied to the real world [36, 27, 37, 33, 16, 17, 23], especially when training with large-scale data [42]. One line of work applies a data-driven curriculum learning approach where the data that are most likely labelled correctly are used for learning in the beginning, and then harder data is taken into learning during a later phase [16]. Researchers have also tried on to apply the loss only on the easiest top k-elements in the batch, determined by the lowest current loss [37]. Inspired by these the easy positive sampling method focuses on selecting only the top easiest positive relations in the batch.

Beyond memorization. Deep networks are shown to be extremely easy to memorize and over-fit to the training data [54, 31, 32]. For example, it is shown the network can be trained with randomly assigned labels on the ImageNet data, and obtain 100% training accuracy if augmentations are not adopted. Moreover, even the CIFAR-10 classifier performs well in the validation set, it is shown that it does not really generalize to new collected data which is visually similar to the training and validation set [31]. In this paper, we show that when allowing the network the freedom not to have to learn inner-class relation between different class modes, we can achieve much better generalization, and the representation can be applied in a zero-shot setting.

3. Preliminaries

Let $X = \{x_1, .., x_n\}$ be a set of samples with labels $y_i \in \{1, .., m\}$. The objective of metric learning is to learn an embedding $f(\cdot, \theta) \to \mathbb{R}^k$, in which the neighbourhood of each sample in the embedding space contains samples only from the same class. One of the common approaches for metric learning is using embedding losses in which at each iteration, samples from the same class and samples from different classes are chosen according to same sampling heuristic. The objective of the loss is to push away projections of samples from different classes, and pull closer projections of samples from a same class. In this section, we introduce a few popular embedding losses.

Notation: Let $x_i, x_j \in X$, define: $D^f_{x_i,x_j} = \| f(x_i) -$

$f(x_j)\|\ ^2$. In cases where there is no ambiguity we omit f and simply write D_{x_i,x_j} . We also define the function $\delta_{x_i,x_j} = \begin{cases} 1 & \text{if } y_i = y_j \\ 0 & \text{otherwise} \end{cases}$. Lastly, for every $a \in \mathbb{R}$, denote $(a)_+ := max(a, 0)$.

The Contrastive loss [11] takes tuples of samples embeddings. It pushes tuples of samples from different classes apart and pulls tuples of samples from the same class together.

$$\mathcal{L}^f_{con}(x_i, x_j) = \delta_{x_i,x_j} \cdot D^f_{x_i,x_j} + (1 - \delta_{x_i,x_j}) \cdot (\alpha - D^f_{x_i,x_j})_+$$

Here α is the margin parameter which defines the desired minimal distance between samples from different classes.

While the Contrastive loss imposes a constraint on a pair of samples, the Triplet loss [3] functions on a triplet of samples. Given a triplet $x_a, x_p, x_n \in X$, let

$$h^f(x_a, x_p, x_n) = (D^f_{x_a,x_p} - D^f_{x_p,x_n} + \alpha)_+$$

the triplet loss is defined by

$$\mathcal{L}^f_{trip}(x_a, x_p, x_n) = \delta_{x_a,x_p} \cdot (1 - \delta_{x_a,x_n}) \cdot h^f(x_a, x_p, x_n)$$

The Margin loss [51] aims to exploit the flexibility of Triplet loss while maintaining the computational efficiency of the Contrastive loss. This is done by adding a variable (β_x for $x \in X$) which determines the boundary between positive and negative pairs; given an anchor $t \in X$, let

$$g(z_1, z_2) = (z_1 - z_2 + \alpha)_+$$

the loss is defined by

$$\mathcal{L}^{f,\beta}_{mar}(t, x) = \delta_{t,x} \cdot g(D^f_{t,x}, \beta_t) + (1 - \delta_{t,x}) \cdot g(\beta_t, D^f_{t,x})$$

4. Class-collapsing

The contrastive loss objective is to pull all the samples with the same class to a single point in the embedding space. We call this the *Class-collapsing* property. Formally, an embedding $f : X \to \mathbb{R}^m$ has the class-collapsing property, if there exists a label y and a point $p \in \mathbb{R}^m$ such that $\{f(x_i)| \quad y_i = y\} = \{p\}$.

4.1. Embedding losses optimal solution

It is easy to see that an embedding function f that minimizes:

$$\mathbb{O}_{con}(f) = \frac{1}{n^2}\left(\sum_{x_i,x_j \in X} \mathcal{L}^f_{con}(x_i, x_j)\right)$$

has the class-collapsing property with respect to all classes. However, this is not necessarily true for the Triplet loss and the Margin loss.

For simplification for the rest of this subsection we will assume there are only two classes. Let $A \subset X$ be a subset of elements such that all the elements in A belongs to one class and all the element in A^c belong to the other class.

Recall some basic set definitions.

Definition 1. For all sets $Y, Z \subset \mathbb{R}^m$ define:

1. The diameter of Y is defined by:

$$diam(Y) = \sup\{\|y - z\| \ |y, z \in Y\}$$

2. The distance between Y and Z is:

$$\|Y - Z\| = \inf\{\|y - z\| \ |y \in Y, z \in Z\}$$

It is easy to see that if $f : X \to \mathbb{R}^m$ is an embedding, such that $diam(f(A)) < 2 \cdot \alpha + \|f(A) - f(B)\|$, then:

$$\mathbb{O}_{trip}(f) = \frac{1}{n^3}\left(\sum_{x_i,x_j,x_k \in X} \mathcal{L}^f_{trip}(x_i, x_j, x_k)\right) = 0.$$

Moreover, fixing $\beta_{x_i} = \alpha$ for every $x_i \in X$, then:

$$\mathbb{O}_{margin}(f, \beta) = \frac{1}{n^2}\left(\sum_{x_i,x_j \in X} \mathcal{L}^{f,\beta}_{margin}(x_i, x_j)\right) = 0.$$

It can be seen that indeed, the family of embeddings which induce the global-minimum with respect to the Triplet loss and the Margin loss, is rich and diverse. However, as we will prove in the next subsection, this does not remain true in a noisy environment scenario.

4.2. Noisy environment analysis

For simplicity we will also discuss in this section the binary case of two labels, however this could be extended easily to the multi-label case.

The noisy environment scenario can be formulated by adding uncertainty to the label class. More formally, let $Y = \{Y_1, .., Y_n\}$ be a set of independent binary random variables. Let $A_1, .., A_t \subset X$, $0.5 < p < 1$ such that: $|A_j| = \frac{n}{t}$ and

$$\mathbb{P}(Y_i = k) = \begin{cases} p & x_i \in A_k \\ q' := \frac{1-p}{t-1} & x_i \notin A_k \end{cases}$$

We can also reformulate δ as a binary random variable such that:

$$\overline{\delta}_{Y_i,Y_j} := \mathbf{1}_{Y_i = Y_j}$$

The loss with respect to embedding f is a random variable and the objective is to minimize its expectation

$$\mathbb{E}\mathcal{L}^f_{tr}(x_i,x_j,x_k) = \mathbb{E}\left(\bar{\delta}_{Y_i,Y_j}\cdot(1-\bar{\delta}_{Y_i,Y_k})\right)\cdot h^f(x_i,x_j,x_k)$$

Therefore, we are searching for an embedding function which minimize

$$\mathbb{E}\mathbb{O}_{trip}(f) = \frac{1}{n^3}\sum_{x_i,x_j,x_k\in X}\mathbb{E}\mathcal{L}^f_{trip}(x_i,x_j,x_k)$$

In Appendix A we will prove the following two theorems.

Theorem 1. *Let $f : X \to \mathbb{R}^m$ be an embedding, which minimize $\mathbb{E}\mathbb{O}_{trip}(f)$, then f has the class-collapsing property with respect to all classes.*

Similarly, we can define:

$$\mathbb{E}\mathcal{L}^f_{mar}(x_i,x_j) = \mathbb{E}\bar{\delta}_{Y_i,Y_j}\cdot(D^f_{x_i,x_j}-\beta_{x_i}+\alpha)_+$$
$$+\,\mathbb{E}(1-\bar{\delta}_{Y_i,Y_j})\cdot(\beta_{x_i}-D^f_{x_i,x_j}+\alpha)_+$$

Theorem 2. *Let $f : OX \to \mathbb{R}^m$ be an embedding, which minimize*

$$\mathbb{E}\mathbb{O}_{margin}(f,\beta) = \frac{1}{n^2}\sum_{x_i,x_j\in X}\mathbb{E}\mathcal{L}^f_{margin}(x_i,x_j),$$

then f has the class-collapsing property with respect to all classes.

In conclusion, although theoretically in clean environments the Triplet loss and Margin loss should allow more flexible embedding solutions, this does not remain true when noise is considered. On a real-world data, where mislabeling and ambiguity can be usually be found, the optimal solution with respect to both these losses becomes degenerate.

4.3. Easy Positive Sampling (EPS)

Using standard embedding losses for metric learning can result in an embedding space in which visually diverse samples from the same class are all concentrated in a single location in the embedding space. Since the standard evaluation and prediction method for image retrieval tasks are typically based on properties of the K-nearest neighbours in the embedding space, the class-collapsing property is a side-effect which is not necessarily in order to get optimal results. In the next section, we will show experimental results, which support the assumption that complete class-collapsing can hurt the generalization capability of the network.

To address the class-collapsing issue we propose a simple method for sampling, which results in weakening the objective penalty on the inner-class relations, by applying the loss only on the closest positive sample. Formally we define the EPS sampling in the following way; given a mini-batch with N samples, for each sample a, let C_a be the set of elements from the same class as a in the mini-batch, we choose the positive sample p_a to be

$$\arg\min_{t\in C_a}(\|f(t)-f(a)\|)$$

For negative samples n_a we can choose according to various options. In this paper we use the following methods: **(a)** Choosing randomly from all the elements which are not in C_a. **(b)** Using distance sampling [51]. **(c)** semi-hard sampling [35],**(d)** MS hard-mining sampling [49]. We then apply the loss on the triplets (a, p_a, n_a). Using such sampling changes the loss objective such that instead of pulling all samples in the mini-batch from the same class to be close to the anchor, it only pulls the closest sample to the anchor (with respect to the embedding space) in the mini-batch, see Figure 1.

In Appendix B, we formalize this method in the noisy environment framework. We prove (Claim 1,2) that every embedding which has the class collapsing property is *not* a minimal solution with respect to both the margin and the triplet loss with the easy positive sampling. Furthermore, in Claim 3,4 we prove that the objective of the losses with EPS on tuples/triplets is to push away every element (including positive elements), that is not in the k-closest elements to the anchor, where k is determined by the noise level p. Therefore, if we apply the EPS method on a mini-batch which has small number of positive elements from each modality, in such case adding the EPS to the losses not only relaxes the constraints on the embedding, allowing the embedding to have multiple inner-clusters. It also optimizes the embedding to have this form.

5. Experiments

We test the EPS method on image retrieval and clustering datasets. We evaluate the image retrieval quality based on the recall@k metric [15] , and the clustering quality by using the normalized mutual information score (NMI) [25]. The NMI measures the quality of clustering alignments between the clusters induced by the ground-truth labels and clusters induced by applying clustering algorithm on the embedding space. The common practice to choose the NMI clusters is by using K-means algorithm on the embedding space, with K equal to the number of classes. However, this prevents from the measurement capturing more diverse solutions in which homogeneous clusters appear only when using larger amount of clusters. Regular NMI prefers solutions with class-collapsing. Therefore, we increase the number of clusters in the NMI evaluation (denote it by NMI+) we also report the regular NMI score.

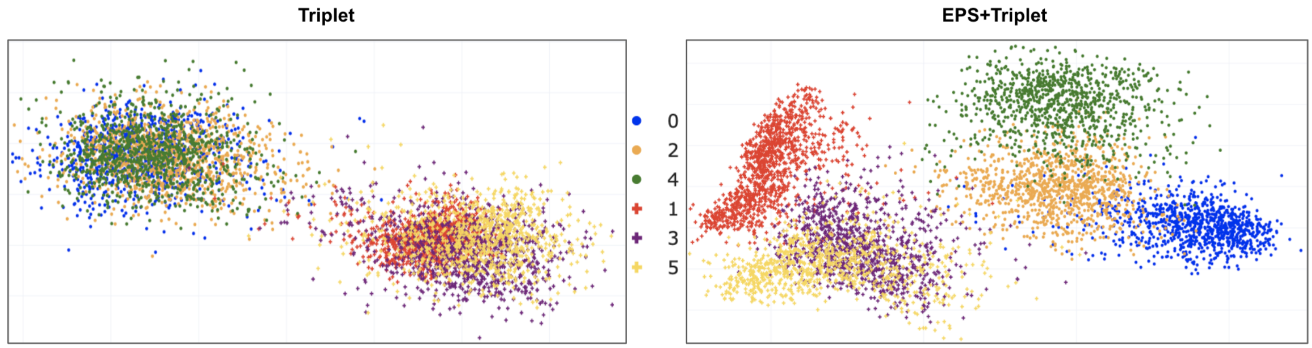

Figure 2: Embedding examples from the MNIST validation set, after training using only even/odd labels. Different colors indicate different digits. **Left:** Using Triplet-loss, class collapsing pushes all intra-class digits to overlapping clusters. **Right:** With EPS, different digits form separate clusters. Retrieval or classification using the odd-vs-even task/metric is more effectively implemented using the embedding on the right, even though the embedding on the left is learned with a loss that more strictly optimizes for the task.

	MNIST Train Digits		MNIST Test Digits	
	Triplet	Trip+EPS	Triplet	Trip+EPS
R@1	42.0	**65.8**	35.2	**42.3**
R@5	87.5	**93.6**	80.9	**83.9**
R@10	96.6	**97.4**	93.3	**93.6**

Table 1: Recall@k evaluated on MNIST dataset. The train classes are digits 0-5 and the test classes are digits 6-9

5.1. MNIST Even/Odd Example

To demonstrate the class-collapsing phenomena, we take the MNIST dataset [20], and split the digits according to odd and even. From a visual perspective this is an arbitrary separation. We took the first 6 digits for training and left the remaining 4 digits for testing. We used a simple shallow architecture which results in an embedding function from the image space to $\mathbb{R}^2$ (For implementation details see Appendix C).

We train the network using the triplet loss. We compare the EPS method to random sampling of positive examples (the regular loss). As can be seen in Figure 2, the regular training without EPS suffers from class-collapsing. Training with EPS creates a richer embedding in which there is a clear separation not only between the two-classes, but also between different digits from the same class. As expected, the class-collapsing embedding preforms worse on the test data with the unseen digits, see Table 1.

5.2. Fine-grained Recognition Evaluation

We compare the EPS approach to previous popular sampling methods and losses. The evaluation is conducted on standard benchmarks for zero-shot learning and image retrieval following the common splitting and evaluation practice [51, 26, 2]. We build our implementation on top of the framework of [34], which allows us to have a fair comparison between all the tested methods with an embedding of fix size (128). For more implementation details and consistency of the results, see Appendix C.

5.2.1 Datasets

We evaluate our model on the following datasets.

- **Cars-196** [18], which contains 16,185 images of 196 car models. We follow the split in [51], using 98 classes for training and 98 classes for testing.

- **CUB200-2011** [46], which contains 11,788 images of 200 bird species. We also follow [51], using 100 classes for training and 100 for testing.

- **Omniglot** [19], which contains 1623 handwritten characters from 50 alphabet. In our experiments we only use the alphabets labels during the training process, i.e, all the characters from the same alphabet has the same class. We follow the split in [19] using 30 alphabets for training and 20 for testing.

5.2.2 Architecture details

We use an embedding of size 128, and an input size of 224×224 for the first two datasets, and 80×80 for the Omniglot dataset. For all the experiments we used the original bounding boxes without cropping around the object box. As a backbone for the embedding, we use ResNet50 [13] with pretrained weights on imagenet. The backbone is followed by a global average pooling and a linear layer which reduces

model	Cars-196					CUB-200				
	R@1	R@2	R@4	NMI	NMI+	R@1	R@2	R@4	NMI	NMI+
Trip. + SH [35]	51.5	63.8	73.5	53.4	-	42.6	55.0	66.4	55.4	-
Trip. + SH†	76.1	84.4	90.0	65.1	68.5	61.5	73.4	82.5	66.2	68.1
ProxyNCA [26]	73.2	82.4	86.4	64.9	-	49.2	61.9	67.9	64.9	-
ProxyNCA†	77.1	85.2	91.2	65.6	68.9	63.1	74.8	83.8	67.2	68.7
Dist-Margin [51]	79.6	86.5	91.9	**69.1**	70.4	63.6	74.4	83.1	**69.0**	68.7
MS [49]	77.3	85.3	90.5	-	-	57.4	69.8	80.0	-	-
MS†	81.2	89.1	93.5	60.5	71.1	62.3	73.3	82.1	59.8	68.0
EPS + Trip. + SH	78.3	85.9	91.4	59.8	69.8	61.8	73.6	82.4	62.4	68.0
EPS + Dist-Margin	**83.6**	**89.5**	**93.6**	67.3	**72.4**	**64.7**	**75.2**	**84.3**	68.2	**69.4**
EPS + MS	82.9	89.4	93.2	60.0	72.0	63.3	74.2	82.5	61.2	68.2

Table 2: Recall@k and NMI performance on Cars196 and CUB200- 2011. NMI+ indicate the NMI measurement when using 10 (number of classes) clusters. The EPS method improves in all cases. †: Our re-implemented version with the same embedding dimension.

model	Omniglot-letters					Omniglot-languages				
	R@1	R@2	R@4	R@8	NMI	R@1	R@2	R@4	R@8	NMI+
Trip. + SH [35]	49.4	60.0	69.2	76.9	66.2	71.0	80.2	87.6	92.4	38.7
ProxyNCA [26]	49.1	60.4	70.9	78.9	69.0	73.0	82.1	88.8	93.5	43.3
Dist-Margin [51]	49.4	61.1	70.1	79.2	68.9	73.2	82.3	89.1	94.0	43.5
MS [49]	57.7	68.5	77.3	83.8	69.2	78.8	86.4	92.0	95.4	46.0
EPS + Trip. + SH	**68.4**	**79.3**	**86.9**	**92.1**	**79.6**	**85.2**	**91.1**	**94.9**	**97.3**	**52.6**
EPS + Dist-Margin	66.2	76.7	84.8	90.3	77.9	83.0	89.4	93.6	96.4	50.7
EPS + MS	68.7	79.1	86.9	92.2	77.3	86.2	91.7	94.9	97.2	53.8

Table 3: Recall@k and NMI performance on Omniglot dataset. In both cases the training was done with only language labels. **Right:** evaluation on language labels. **Left:** evaluation on letter labels. NMI+ indicate the NMI measurement when using 30*(number of classes) clusters. The EPS method improves in both cases.

the dimension to the embedding size. Optimization is performed using Adam with a learning rate of 10^{-5}, and the other parameters set to default values from

5.2.3 Results

We tested the EPS method with 3 different losses: Triplet [3], Margin [51] and Multi-Similarity (MS) [49]. For the Margin loss experiment, we combine the EPS with distance sampling [51]; this could be done because the distance sampling only constrains on the negative samples, where our method only constrains on the positive samples. We set the margin $\alpha = 0.2$ and initialized $\beta = 1.2$ as in [51]. For the Triplet we combine EPS with semi-hard sampling [35] by fixing the positive according to EPS and then using semi-hard sampling for choosing the negative examples. For the MS loss we replace the positive hard-mining method with EPS and use the same hard-negative method. We use the same hyper-paremeters as in [49] $\alpha = 2, \lambda = 1, \beta = 50$.

Results are summarized in Tables 2 and 3. We can see that EPS achieves the best performances. It is important to note that in the baseline models, when using Semi-hard sampling, the sampling strategy was done also on the positive part as suggest in the original papers. We see that replacing the semi-hard positive sampling with easy-positive sampling, improve results in all the experiments. The improvement gain becomes larger as the dataset classes can be partitioned more naturally to a small number of sub-clusters which are visually homogeneous. In Cars196 dataset it is the car viewpoint, where in Omniglot it is the letters in each language. As can be seen in Table 3, using EPS on the Omniglot dataset result in creating an embedding in which in most cases the nearest neighbor in the embedding consists of element of the same letter, although the network was trained without these labels. In Figure 3 we can see a qualitatively comparison of CARS16 models results. EPS seems to create more homogeneous neighbourhood relationships with respect to the the viewpoint of the car. More results and comparisons can be find in Appendix C.

5.2.4 Positive batch size effect

An important hyperparameter in our sampling method is the number of positive batch samples, from which we select the closest one in the embedding space to the anchor. If the class is visually diverse and the number of positive samples in batch is low, than with high probability the set of all the

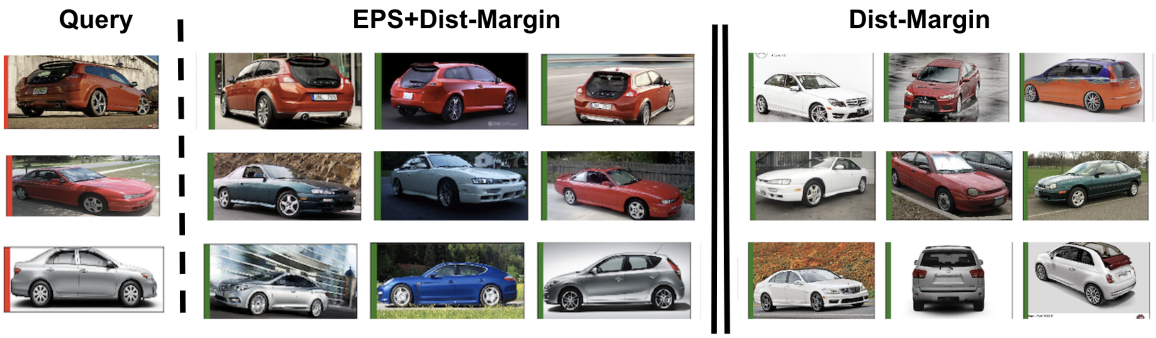

Figure 3: Retrieval results for randomly chosen query images in Cars196 dataset. Using EPS creates more homogeneous neighbourhood relationships with respect to the car viewpoint.

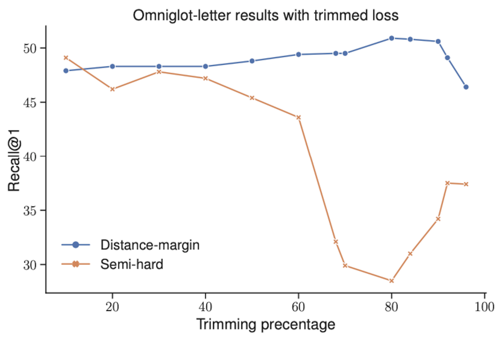

Figure 4: Recall@1 performance with Trimmed loss across varying trimming percentage. Except for small improvement in the Distance-margin case, there is no significant improvement when applying the Trimmed loss.

	Language		Letters	
	Semi-hard	Semi-hard+EPS	Semi-hard	Semi-hard+EPS
NMI	**93.6**	67.3	78.4	**87.1**
R@1	**99.9**	94.5	70.3	**77.5**
R@2	**100**	96.8	80.4	**86.3**
R@4	**100**	98.1	87.9	**92.4**
R@8	**100**	99.2	93.3	**96.0**

Table 4: Results of semi-hard with/without EPS on the Omniglot training dataset. Without EPS the network feet almost perfectly to the training set. However, using EPS results in batter performances on the letters fine-grained task.

positive samples will not contain any visually similar image to the anchor. In case of the Omniglot experiment, the effect of this hyperparameter is clear; It determines the probability that the set of positive samples will include a sample from the same letter as the anchor letter. As can be seen in Figure 5(b), the performance of the model increases as the probability of having another sample with the same letter as the anchor increases.

5.2.5 Trimmed Loss comparison

The situation where a class consists of multiple modes can also be seen as a noisy data scenario with respect to the embedding loss, where positive tuples consisting of examples from different modes are considered as 'bad' labelling. One approach to address noisy labels is by back-propagating the loss only on the k-elements in the batch with the lowest current loss [37]. Although this approach resembles [24], the difference is that in [24] they apply the trimming only on the positive tuples. We test the effect of using Trimmed Loss on random sampled triplets with different level of trimming percentage. As can be seen in Figure 4, there is only a minor improvement when applying the loss on top of the distance-margin loss on the Omniglot-letters dataset. This emphasizes the importance of constraining the trimming to the positive sampling only.

5.2.6 Embedding behavior on training sets

The class-collapsing phenomena also occur in the training process of the image retrieval datasets. Figure 6 visualise the t-SNE embedding [45] of Cars196 training classes. As can be seen, when training without EPS each class fits well to a bivariate normal-distribution with small variance and different means. Training with EPS result in more diverse distributions and in some of the classes fits batter to a mixture of multiple different distributions.

This can also be measured qualitatively on the Omniglod detest; although training without the EPS results in batter overfitting to training samples, the results on the letters fine-grained task are significantly inferior comparing to training with the EPS (Table 4). It is also important to note the

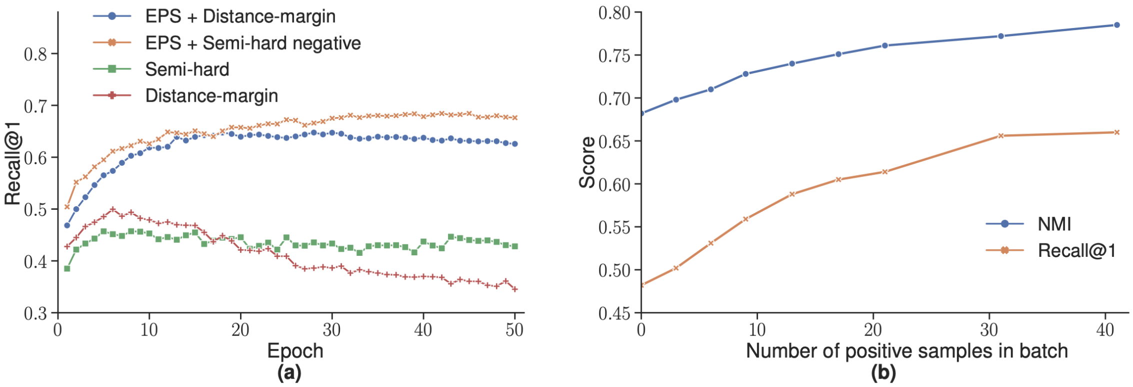

Figure 5: Results on Omniglot-letters. **(a)** Recall@1 performance of each model per epoch. **(b)** performance of *EPS + distance-margin* model on the Omniglot dataset, as a function of the number of positive samples in batch (where zero is equivalent to only using only distance sampling). Increasing the number the number of positive samples enhances the model performance.

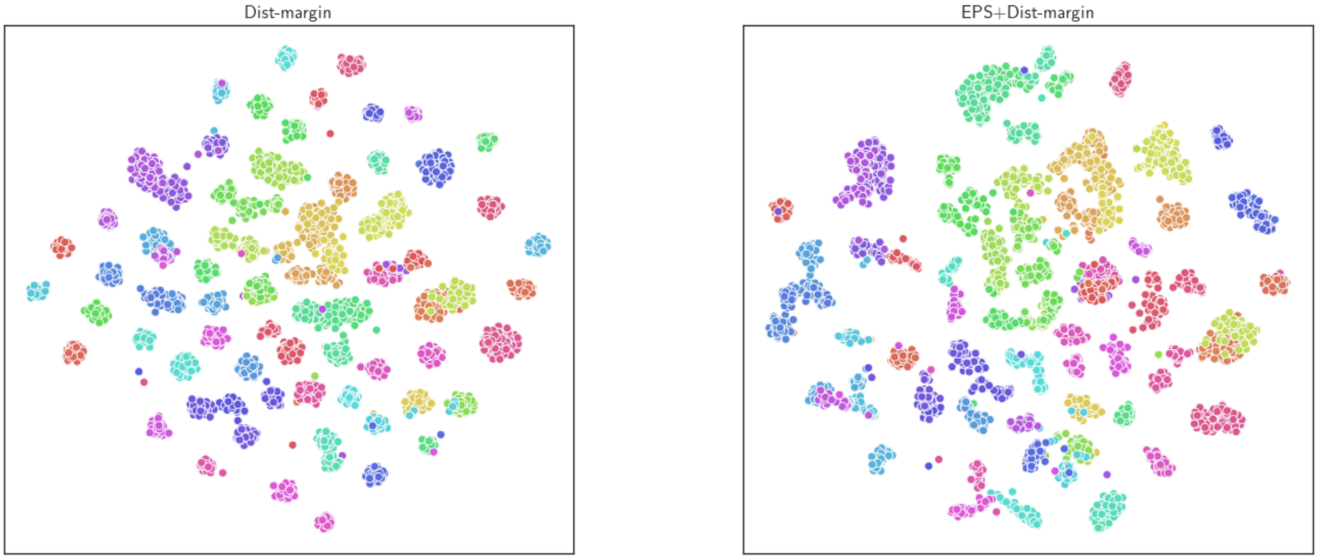

Figure 6: t-SNE visualization of Cars196 training classes (each class has a different color). Training with EPS results in more diverse classes appearance.

low NMI score when using EPS with the number of clusters equal to the number of languages, and the increment of this score when increasing the number of clusters to the number of letters. This indicates that training with EPS results in more homogeneous small clusters, which are more blended in the embedding space comparing to training without EPS.

6. Conclusion

In this work we we investigate the class collapsing phenomena with respect to popular embedding losses such as the Triplet loss and the Margin loss. While in clean environments there is a diverse and rich family of optimal solutions, when noise is present, the optimal solution collapses to a degenerate embedding. We propose a simple solution to this issue based on 'easy' positive sampling, and prove that indeed adding this sampling results in non-degenerate embeddings. We also compare and evaluate our method on standard image retrieval datasets and demonstrate a consistent performance boost on all of them. While our analysis and results have been limited to metric learning frameworks, we believe that this type of noisy analysis might be useful in other settings, and can better reflect the training dynamic of neural-networks on real-world datasets.

References

[1] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic. Netvlad: Cnn architecture for weakly supervised place recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5297–5307, 2016. 2

[2] B. Brattoli, K. Roth, and B. Ommer. Mic: Mining interclass characteristics for improved metric learning. In *Proceedings of the Intl. Conf. on Computer Vision (ICCV)*, 2019. 5

[3] G. Chechik, V. Sharma, U. Shalit, and S. Bengio. Large scale online learning of image similarity through ranking. *Journal of Machine Learning Research*, 11:1109–1135, 2010. 1, 2, 3, 6

[4] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton. A simple framework for contrastive learning of visual representations. 2020. 2

[5] S. Chopra, R. Hadsell, and Y. LeCun. Learning a similarity metric discriminatively, with application to face verification. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 539–546. IEEE, 2005. 1

[6] E. D. Cubuk, B. Zoph, D. Mane, V. Vasudevan, and Q. V. Le. Autoaugment: Learning augmentation strategies from data. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 113–123, 2019. 2

[7] G. Cybenko. Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals, and Systems (MCSS)*, 2(4):303–314, Dec. 1989. 2

[8] A. Der Kiureghian and O. Ditlevsen. Aleatoric or epistemic? does it matter? *Structural Safety*, 31(2):105–112, 2009. 2

[9] A. Frome, G. S. Corrado, J. Shlens, S. Bengio, J. Dean, M. Ranzato, and T. Mikolov. Devise: A deep visual-semantic embedding model. In *Advances in neural information processing systems*, pages 2121–2129, 2013. 1

[10] A. Gordo, J. Almazán, J. Revaud, and D. Larlus. Deep image retrieval: Learning global representations for image search. In *European conference on computer vision*, pages 241–257. Springer, 2016. 1

[11] R. Hadsell, S. Chopra, and Y. LeCun. Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 2 (CVPR06)*. IEEE. 3

[12] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020. 2

[13] K. He, X. Zhang, S. Ren, and J. Sun. Identity mappings in deep residual networks. In *Computer Vision – ECCV 2016*, pages 630–645. Springer International Publishing, 2016. 5

[14] E. Hoffer and N. Ailon. Deep metric learning using triplet network. In *International Workshop on Similarity-Based Pattern Recognition*, pages 84–92. Springer, 2015. 1

[15] H. Jégou, M. Douze, and C. Schmid. Product quantization for nearest neighbor search. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(1):117–128, jan 2011. 4

[16] L. Jiang, Z. Zhou, T. Leung, L.-J. Li, and L. Fei-Fei. Mentornet: Learning data-driven curriculum for very deep neural networks on corrupted labels. *arXiv preprint arXiv:1712.05055*, 2017. 2

[17] A. Khetan, Z. C. Lipton, and A. Anandkumar. Learning from noisy singly-labeled data. *arXiv preprint arXiv:1712.04577*, 2017. 2

[18] J. Krause, M. Stark, J. Deng, and L. Fei-Fei. 3d object representations for fine-grained categorization. In *4th International IEEE Workshop on 3D Representation and Recognition (3dRR-13)*, Sydney, Australia, 2013. 5

[19] B. M. Lake, R. Salakhutdinov, and J. B. Tenenbaum. Human-level concept learning through probabilistic program induction. *Science*, 350(6266):1332–1338, dec 2015. 5

[20] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. 5

[21] I. Loshchilov and F. Hutter. Online batch selection for faster training of neural networks. *arXiv preprint arXiv:1511.06343*, 2015. 2

[22] D. G. Lowe. Similarity metric learning for a variable-kernel classifier. *Neural Computation*, 7(1):72–85, jan 1995. 1

[23] E. Malach and S. Shalev-Shwartz. Decoupling" when to update" from" how to update". In *Advances in Neural Information Processing Systems*, pages 960–970, 2017. 2

[24] T. Malisiewicz and A. A. Efros. Recognition by association via learning per-exemplar distances. In *CVPR*, June 2008. 7

[25] C. D. Manning, P. Raghavan, and H. Schutze. *Introduction to Information Retrieval*. Cambridge University Press, 2008. 4

[26] Y. Movshovitz-Attias, A. Toshev, T. K. Leung, S. Ioffe, and S. Singh. No fuss distance metric learning using proxies. In *2017 IEEE International Conference on Computer Vision (ICCV)*. IEEE, oct 2017. 5, 6

[27] N. Natarajan, I. S. Dhillon, P. K. Ravikumar, and A. Tewari. Learning with noisy labels. In *Advances in neural information processing systems*, pages 1196–1204, 2013. 2

[28] H. Oh Song, Y. Xiang, S. Jegelka, and S. Savarese. Deep metric learning via lifted structured feature embedding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4004–4012, 2016. 2

[29] O. M. Parkhi, A. Vedaldi, and A. Zisserman. Deep face recognition. 2015. 2

[30] Q. Qian, L. Shang, B. Sun, J. Hu, H. Li, and R. Jin. Softtriple loss: Deep metric learning without triplet sampling. 1

[31] B. Recht, R. Roelofs, L. Schmidt, and V. Shankar. Do cifar-10 classifiers generalize to cifar-10? *arXiv preprint arXiv:1806.00451*, 2018. 2

[32] B. Recht, R. Roelofs, L. Schmidt, and V. Shankar. Do imagenet classifiers generalize to imagenet? *arXiv preprint arXiv:1902.10811*, 2019. 2

[33] S. Reed, H. Lee, D. Anguelov, C. Szegedy, D. Erhan, and A. Rabinovich. Training deep neural networks on noisy labels with bootstrapping. *arXiv preprint arXiv:1412.6596*, 2014. 2

[34] K. Roth, T. Milbich, S. Sinha, P. Gupta, B. Ommer, and J. P. Cohen. Revisiting training strategies and generalization performance in deep metric learning. 5

[35] F. Schroff, D. Kalenichenko, and J. Philbin. FaceNet: A unified embedding for face recognition and clustering. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, jun 2015. 2, 4, 6

[36] C. Scott, G. Blanchard, and G. Handy. Classification with asymmetric label noise: Consistency and maximal denoising. In *Conference On Learning Theory*, pages 489–511, 2013. 2

[37] Y. Shen and S. Sanghavi. Learning with bad training data via iterative trimmed loss minimization. In K. Chaudhuri and R. Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 5739–5748, Long Beach, California, USA, 09–15 Jun 2019. PMLR. 2, 7

[38] A. Shrivastava, A. Gupta, and R. Girshick. Training region-based object detectors with online hard example mining. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 761–769, 2016. 2

[39] E. Simo-Serra, E. Trulls, L. Ferraz, I. Kokkinos, P. Fua, and F. Moreno-Noguer. Discriminative learning of deep convolutional feature point descriptors. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 118–126, 2015. 2

[40] K. K. Singh and Y. J. Lee. Hide-and-seek: Forcing a network to be meticulous for weakly-supervised object and action localization. In *2017 IEEE international conference on computer vision (ICCV)*, pages 3544–3553. IEEE, 2017. 2

[41] K. Sohn. Improved deep metric learning with multi-class n-pair loss objective. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 1857–1865. Curran Associates, Inc., 2016. 1

[42] C. Sun, A. Shrivastava, S. Singh, and A. Gupta. Revisiting unreasonable effectiveness of data in deep learning era. In *Proceedings of the IEEE international conference on computer vision*, pages 843–852, 2017. 2

[43] Y. Sun, Y. Chen, X. Wang, and X. Tang. Deep learning face representation by joint identification-verification. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 1988–1996. Curran Associates, Inc., 2014. 1

[44] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1701–1708, 2014. 1

[45] L. van der Maaten and G. Hinton. Visualizing high-dimensional data using t-sne. 2008. 7

[46] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The Caltech-UCSD Birds-200-2011 Dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011. 5

[47] J. Wang, Y. Song, T. Leung, C. Rosenberg, J. Wang, J. Philbin, B. Chen, and Y. Wu. Learning fine-grained image similarity with deep ranking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1386–1393, 2014. 1

[48] X. Wang and A. Gupta. Unsupervised learning of visual representations using videos. In *ICCV*, 2015. 2

[49] X. Wang, X. Han, W. Huang, D. Dong, and M. R. Scott. Multi-similarity loss with general pair weighting for deep metric learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5022–5030, 2019. 4, 6

[50] X. Wang, A. Shrivastava, and A. Gupta. A-fast-rcnn: Hard positive generation via adversary for object detection. In *CVPR*, 2017. 2

[51] C.-Y. Wu, R. Manmatha, A. J. Smola, and P. Krahenbuhl. Sampling matters in deep embedding learning. In *2017 IEEE International Conference on Computer Vision (ICCV)*. IEEE, oct 2017. 1, 2, 3, 4, 5, 6

[52] Z. Wu, Y. Xiong, S. X. Yu, and D. Lin. Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3733–3742, 2018. 2

[53] H. Xuan, A. Stylianou, and R. Pless. Improved embeddings with easy positive triplet mining. In *IEEE Winter Conference on Applications of Computer Vision, WACV 2020, Snowmass Village, CO, USA, March 1-5, 2020*, pages 2463–2471. IEEE, 2020. 1, 2

[54] C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals. Understanding deep learning requires rethinking generalization. *arXiv preprint arXiv:1611.03530*, 2016. 2