$\mathbb{X}$-Sample Contrastive Loss: Improving Contrastive Learning with Sample Similarity Graphs

Vlad Sobal [1 2] Mark Ibrahim [2] Randall Balestriero [3] Vivien Cabannes [2] Diane Bouchacourt [2] Pietro Astolfi [2]
Kyunghyun Cho [1 4 5] Yann LeCun [1 2]

Abstract

Learning good representations involves capturing the diverse ways in which data samples relate. Contrastive loss—an objective matching related samples—underlies methods from self-supervised to multimodal learning. Contrastive losses, however, can be viewed more broadly as modifying a similarity graph to indicate how samples should relate. This view reveals a shortcoming: the contrastive similarity graph is binary, as only one sample is the positive sample. Crucially, similarities *across* samples are ignored. We revise the standard contrastive loss to explicitly encode how a sample relates to others, and introduce a new objective, called $\mathbb{X}$-Sample Contrastive, to train vision models based on similarities in class or text caption descriptions. Our study spans three scales: ImageNet-1k with 1 million, CC3M with 3 million, and CC12M with 12 million samples. The representations learned via our objective outperform both contrastive self-supervised and vision-language models trained on the same data across a range of tasks. When training on CC12M, we outperform CLIP by 0.6% on both ImageNet and ImageNet Real. Our objective appears to work particularly well in lower-data regimes, with gains over CLIP of 16.8% on ImageNet and 18.1% on ImageNet Real when training with CC3M. Finally, our objective seems to encourage the model to learn representations that separate objects from their attributes and backgrounds, with gains of 3.3-5.6% over CLIP on ImageNet9.

1. Introduction

Contrastive loss underlies methods from self-supervised learning (SSL) to multimodal learning (Radford et al., 2021; Chen et al., 2020; Oord et al., 2018). In SSL, contrastive learning encourages the model to associate a sample with another view of the sample created using hand-crafted data augmentation—this related view is the positive sample. Other samples are then pushed away as negative, unrelated samples in the models' representation space. Contrastive losses also play a crucial role in multimodal models such as CLIP (Radford et al., 2021), where the model associates an image with its text caption in representation space. Here contrastive learning designates the caption and image representations as positives while all other text-image pairs are designated as unrelated negatives.

More broadly, contrastive losses can be seen as modifying a similarity graph to indicate how samples should relate in the model's representation space (Cabannes et al., 2023). This view reveals a shortcoming in contrastive learning: the similarity graph is binary, as only one sample is the related positive sample. Crucially, similarities across samples, containing precious signals about how aspects of one sample may relate to another, are ignored. For example, as shown in fig. 1, contrastive learning treats each text-image pair independently, without explicitly encoding similarities in the images depicting dogs and the others sharing a grassy background. Standard contrastive objectives do not explicitly account for similarities across samples, thereby limiting the quality of the learned representations. Here, we explore here how to capture such similarities by modifying the standard contrastive objective.

To account for similarities across samples, we first remove the binary negative vs. positive designations in standard contrastive loss. We introduce instead a similarity graph with continuous scalars capturing the extent to which two samples are related. Consider the example in fig. 1, where the two dog images have a high similarity while the dog and cat images have a more moderate similarity. We experiment with this new objective, called $\mathbb{X}$-Sample Contrastive ($\mathbb{X}$-CLR), by training vision models using a graph of similarities inferred from class or text caption descriptions found in common datasets. Our study spans three training dataset scales from 1 million samples with high-quality labels from ImageNet (Deng et al., 2009) to 3 and 12 million noisy image-text caption pairs from CC3M and CC12M (Sharma et al., 2018).

We find that compared to contrastive baseline methods

[1]New York University [2]Meta FAIR [3]Brown University [4]Genentech [5]CIFAR. Correspondence to: Vlad Sobal <us441@nyu.edu>.

Published at ICML 2024 Workshop on Foundation Models in the Wild. Copyright 2024 by the author(s).

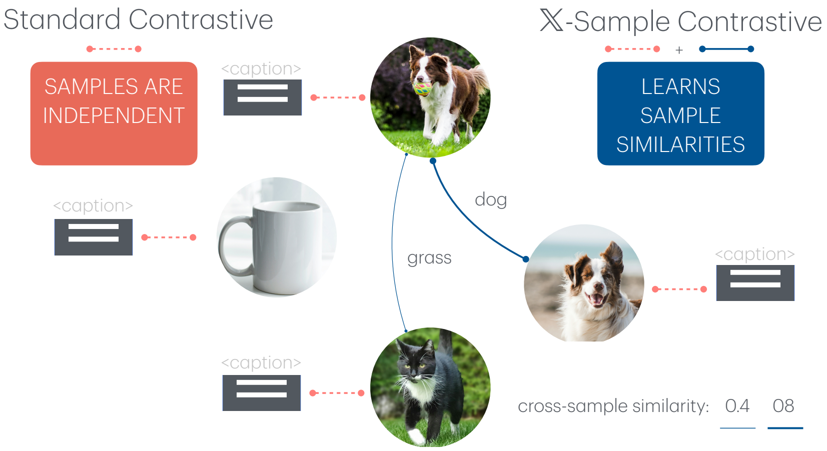

Standard Contrastive
$\mathbb{X}$-Sample Contrastive
+
SAMPLES ARE INDEPENDENT
LEARNS SAMPLE SIMILARITIES
<caption>
<caption>
<caption>
<caption>
dog
grass
cross-sample similarity: 0.4 08

Figure 1: **Capturing similarities across samples in vision-language data with X-Sample Contrastive Loss.** Standard contrastive losses do not model the relationship across the samples (left), while our method $\mathbb{X}$-CLR (right) takes into account the soft inter-sample relationships. $\mathbb{X}$-CLR pushes the representations of two different photos of dogs together. The relationship between the photos of a dog and a cat on grassy backgrounds is also captured, albeit with a smaller similarity.

trained on the same data, representation trained using $\mathbb{X}$-CLR outperform contrastive training on a range of tasks from standard classification to tasks involving the decomposition of objects from their attributes and backgrounds. When training on CC12M, we outperform CLIP by 0.6% on both ImageNet and ImageNet Real (Beyer et al., 2020). Furthermore, $\mathbb{X}$-CLR seems to encourage the model to learn representations that separate objects from their attributes and backgrounds, with gains of 3.4-4.9% over CLIP on ImageNet9 (Xiao et al., 2020). We also find for fine-grained disambiguation of object attributes, the quality of labels used to infer the similarity graph is much more important than the data quantity. Compared to noisier web caption data, we find $\mathbb{X}$-CLR trained on 1 million higher quality class labels outperforms representations learned via standard contrastive CLIP trained 12× more data. Finally, we find $\mathbb{X}$-CLR appears to work particularly well in lower-data regimes, with gains over CLIP of 16.8% on ImageNet and 18.1% on ImageNet Real when training with CC3M. In short, we find representations learned using $\mathbb{X}$-CLR generalize better, decompose objects from their attributes and backgrounds, and are more data-efficient. Overall, our contributions are:

1. We present a graph similarity perspective of contrastive losses, revealing standard losses encode a sparse similarity matrix that treats other, related, samples as negatives.
2. Consequently, we propose a new $\mathbb{X}$-CLR loss that explicitly accounts for similarities across samples
3. We experiment with this objective across three levels of data scale from 1-12 million samples.
4. We find representations learned via $\mathbb{X}$-CLR
   (a) Generalize better on standard classification tasks with consistent gains over contrastive baselines trained on the same data. For example, when training on CC12M we outperform CLIP by 0.6% on both ImageNet and ImageNet Real.
   (b) Disambiguate aspects of images such as attributes and backgrounds more reliably, with gains of 3.3-5.6% over CLIP on background robustness benchmarks for ImageNet.
   (c) Finally, we find $\mathbb{X}$-CLR learns more efficiently when data is scarce, with gains of 16.8% on ImageNet and 18.1% on ImageNet Real when pretraining on the smaller 3 million sample CC3M dataset.

We hope the proposed solution takes a small step towards developing richer learning objectives for understanding sample relations in foundation models to encode richer, more generalizable representations.

2. Understanding contrastive losses via similarity graphs

2.1. X-Sample Graphs

Throughout this study, a similarity graph denotes a graph in which the nodes represent data samples, and edges similarity – relationships. A graph is expressed through its symmetric adjacency matrix $\boldsymbol{G} \in \mathbb{R}^{N \times N}$, the semantic relation between inputs i and j being encoded in the real entry $\boldsymbol{G}_{i,j}$. In fig. 2, we show graphs of different learning paradigms. SSL does not rely on labels, but on positive pairs/tuples/views generated at each epoch. Let us denote by V the number

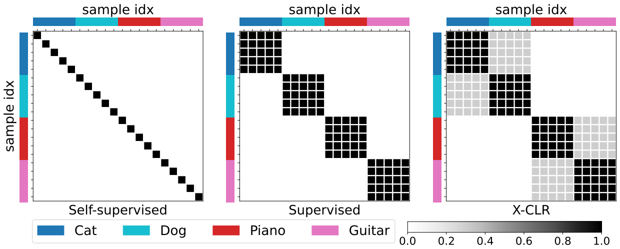


Figure 2: **Sample similarity adjacency matrices of existing methods vs. our $\mathbb{X}$-Sample Contrastive similarity loss (right).** We show pairwise similarities of 20 samples belonging to 4 classes. Similarity of 1 means the samples are identical, 0 – they are completely unrelated. In case of self-supervised learning, none of the inter-sample relationships are modelled (left). Supervised learning relies on the labels to group samples of the same class together (center). $\mathbb{X}$-CLR models inter-class relationships by associating cats with dogs and pianos with guitars.

of positive views generated, commonly $V = 2$ for positive pairs, and denote by E the training epochs. In that case, the original N input samples are transformed into $N \times V \times E$ "augmented" samples

$$\boldsymbol{X}^{(A)} \triangleq [\underbrace{\mathcal{T}(\boldsymbol{x}_1),\dots,\mathcal{T}(\boldsymbol{x}_1)}_{\text{repeated } V \times E \text{ times}},\dots,\mathcal{T}(\boldsymbol{x}_N),\dots,\mathcal{T}(\boldsymbol{x}_N)]^\top,$$

where each $\mathcal{T}$ has its own randomness. The corresponding graph is given by:

$$\boldsymbol{G}^{(\mathrm{ssl})}_{i,j} = \mathbf{1}_{\{\lfloor i/VE \rfloor = \lfloor j/VE \rfloor\}}, \qquad (1)$$

where the associated similarity graph captures if two samples were generated as augmentations of the same original input. Such graphs $\boldsymbol{G}$, as defined by eq. (1), are the ones used as targets in common SSL methods, as formalized below denoting $\boldsymbol{Z} \triangleq f_\theta(\boldsymbol{X}) \in \mathbb{R}^{N \times K}$.

Theorem 1 ((Cabannes et al., 2023)). *SimCLR (Chen et al., 2020) loss can be expressed in terms of the graph $\boldsymbol{G}$ (1)*

$$\mathcal{L}_{\mathrm{SimCLR}}(\boldsymbol{Z}; \boldsymbol{G}) = -\sum_{i,j \in [N]} \boldsymbol{G}_{i,j} \log \left(\frac{\exp(\tilde{\boldsymbol{z}}_i^\top \tilde{\boldsymbol{z}}_j)}{\sum_{k \in [N]} \exp(\tilde{\boldsymbol{z}}_i^\top \tilde{\boldsymbol{z}}_k)} \right)$$

where $\tilde{\boldsymbol{z}} \triangleq \boldsymbol{z} / \|\boldsymbol{z}\|$ and $\tilde{\boldsymbol{Z}}$ the column normalized $\boldsymbol{Z}$ so that each column has unit norm.

In our study, we will focus on contrastive learning, i.e., SimCLR family of losses. We will demonstrate how to move away from the ad-hoc graph $\boldsymbol{G}$ from eq. (1).

2.2. Revisiting contrastive losses with similarity graphs: $\mathbb{X}$-CLR

We introduce the soft cross-sample similarity to the widely used InfoNCE objective (Oord et al., 2018). We note that the proposed framework isn't necessarily only limited to InfoNCE-based methods and can potentially be integrated into non-contrastive objectives. In SimCLR (Chen et al., 2020), given a batch of N images, each image is augmented twice, so each sample has a true positive. The $2N$ images are then encoded to get representation vectors z. Then:

$$p_{i,j} = \frac{\exp(\mathrm{sim}(z_i, z_j)/\tau)}{\sum_{i=1}^{2N} \mathbb{1}_{[k \neq i]} \exp(\mathrm{sim}(z_i, z_k)/\tau)}$$

$$\mathcal{L}_{\mathrm{SimCLR}} = \frac{1}{2N} \sum_{i=1}^{2N} H(\mathbb{1}_{i'}, p_i)$$

where H is the cross-entropy, and $\mathbb{1}_{i'}$ is the one-hot distribution where all the probability mass is assigned to the index of the positive sample corresponding to i, and sim is the cosine similarity. Intuitively, we are training the model to classify positive examples in a batch, so the similarity p should be high only for the true positive. We introduce the soft objective by replacing the hard positive distribution $\mathbb{1}_{i'}$ with a distribution s_i. Or, in terms of graphs, we replace the graph from the eq. (1) with a soft graph where connection strengths can be any number in $[0, 1]$, and, similarly, the distribution s_i and does not have to be one-hot. Considering the example of fig. 1, we want the a photo of a dog to have a representation similar to that of another photo of a dog, somewhat similar to the representation of a cat photo, and different from the representation of a photo of a mug. Given that distribution s, we can plug it in directly:

$$\mathcal{L}_{\mathbb{X}\text{-CLR}} = \frac{1}{2N} \sum_{i=1}^{2N} H(s_i, p_i)$$

There are many possible ways to obtain this distribution s. We could use the meta-data associated with the dataset; in our case, we utilize a trained text encoder f_{text}, and encode the text provided with each image to obtain a representation, which is then used to calculate similarity between samples i and j using the cosine similarity. Those pairwise similarities describe the soft graph: $\boldsymbol{G}^{(\mathrm{soft})}_{i,j} = \mathrm{sim}(f_{\text{text}}(c_i), f_{\text{text}}(c_j))$, were c_i is the caption associated with the i-th sample. The last step before plugging the similarities into the loss function is converting them to a valid distribution using softmax:

$$s_{i,j} = \frac{\exp(\boldsymbol{G}^{(\mathrm{soft})}_{i,j}/\tau_s)}{\sum_{k=1}^{2N} \exp(\boldsymbol{G}^{(\mathrm{soft})}_{i,k}/\tau_s)}$$

Note that τ_s is a separate hyperparameter from τ in the softmax to calculate the learned similarities. Higher values of τ_s put more weight on the 'soft' positives, while lower values in the limit recover the original SimCLR objective.

3. Experiments

3.1. Experimental setup

We test $\mathbb{X}$-CLR on three datasets of varying scale: ImageNet (Deng et al., 2009) (1M), and conceptual captions 3M and 12M (Sharma et al., 2018). We compare to SimCLR (Chen et al., 2020), to CLIP (Radford et al., 2021) when captions are available, and to SupCon (Khosla et al., 2020) on ImageNet. We use the Sentence Transformer (Reimers & Gurevych, 2019) as the text encoder to construct similarities. For ImageNet experiments, we generate captions by using the template "a photo of a _" to generate captions out of class names. In our experiments with the conceptual captions dataset (Sharma et al., 2018), we use the captions as is. For more details, see appendix A.8, and for more experiments, see appendix A.3.

In all our experiments, to isolate the effect of our learning objective, we fix the backbone architecture to be a ResNet-50 (He et al., 2015) model as this is the most widely studied and optimized model for standard contrastive self-supervised learning (Chen et al., 2020). We use the same architecture for CLIP's vision encoder and take advantage of already optimized publicly available checkpoints provided by OpenCLIP (Ilharco et al., 2021) for CC12M. Since no comparable public checkpoint is available for CC3M, we train our own model, see appendix A.7.

3.2. $\mathbb{X}$-CLR with Well-Labeled Samples

We first experiment with $\mathbb{X}$-Sample Contrastive using well-labeled samples to understand the effect of incorporating similarities across samples in the training objective. To do so, we use class labels from ImageNet. We compare $\mathbb{X}$-Sample Contrastive ($\mathbb{X}$-CLR) to SimCLR as well as Supervised Contrastive (SupCon), a model whose objective is to explicitly match samples based on their class labels. We evaluate all models across a suite of benchmarks to gauge how well representations generalize in terms of classification performance.

We find in table 1 that the representations learned via $\mathbb{X}$-CLR improve on standard classification performance, with gains of 12.4% relative to SimCLR and 1.2% relative to Supervised Contrastive on ImageNet. We find similar gains when evaluated on revised labels from ImageNet Real of 14.1% and 1.9%, respectively. Finally, we find by capturing similarities across samples, representations learned via $\mathbb{X}$-CLR are more capable of disambiguating objects from backgrounds and attributes with gains on ImageNet-9 (for details see appendix A.6) (Xiao et al., 2020) and ObjectNet (Barbu et al., 2019).

Table 1: **$\mathbb{X}$-CLR with ImageNet training.**

			Background Decomposition			MIT States	
Method	ImageNet	ImageNet Real	Same Class	Mixed	ObjectNet	Objects	Attributes
SimCLR	63.2	67.5	45.5	38.3	12.5	40.7	28.9
SupCon	74.4	79.7	63.3	58.8	24.1	45.3	**31.1**
$\mathbb{X}$-CLR	**75.6**	**81.6**	**66.5**	**62.3**	**27.7**	**45.8**	30.9

Table 2: **$\mathbb{X}$-CLR with CC3M training.**

			Background Decomposition		
Method	ImageNet	ImageNet Real	Same Class	Mixed	ObjectNet
SimCLR	57.0	64.0	24.4	18.9	10.8
CLIP	41.0	47.6	12.5	10.6	7.8
$\mathbb{X}$-CLR	**58.2**	**65.6**	**26.7**	**20.3**	**11.5**

3.3. $\mathbb{X}$-CLR with Noisy Multimodal Samples

Contrastive loss also plays a pivotal role in multimodal vision-language models such as CLIP. The contrastive training objective matches noisy caption-image pairs. Here we experiment with $\mathbb{X}$-Sample Contrastive by using the noisy captions to learn similarities across samples. We compare both SimCLR as a standard contrastive model and CLIP trained on the same caption-image data across two levels of scale: 3 and 12 million samples from CC3M and CC12M.

We find incorporating $\mathbb{X}$-Contrastive leads to representations with higher classification accuracy and disambiguation of objects from their attributes and backgrounds. With CC12M training shown in table 3, $\mathbb{X}$-Contrastive outperforms SimCLR by 0.5% and CLIP by 0.6% with CC12M with similar gains for ImageNet Real. We also find $\mathbb{X}$-CLR training can better disambiguate object foreground from backgrounds, with gains of 0.6-1.5% over SimCLR and 3.3-5.6% over CLIP.

We find learning similarites across samples with $\mathbb{X}$-CLR leads to more considerable gains when less data is available. $\mathbb{X}$-CLR outperforms SimCLR by 1.2% and CLIP by 16.8% on ImageNet, with similar gains on ImageNet Real as shown in table 2. We find $\mathbb{X}$-CLR training can more considerably disambiguate object foregrounds from backgrounds compared to CLIP when less training data is available, with gains of 10.3-13.3% over CLIP.

Table 3: **$\mathbb{X}$-CLR with CC12M training.**

			Background Decomposition		
Method	ImageNet	ImageNet Real	Same Class	Mixed	ObjectNet
SimCLR	58.9	66	24.6	19.8	12.7
CLIP	58.8	66.1	20.5	17.1	11.9
$\mathbb{X}$-CLR	**59.4**	**66.7**	**26.1**	**20.4**	**13.4**

4. Discussion

We propose a new graph perspective on the commonly used contrastive learning methods and develop a better learning objective, $\mathbb{X}$-CLR, by using a soft similarity graph. We experiment with different ways of constructing the graph, and find that we can build a soft graph that improves over the existing binary graph contrastive methods. However, we believe that there are better ways of constructing the graph than what we found, particularly for the conceptual captions dataset where the captions are quite noisy. We also believe that ideas from $\mathbb{X}$-CLR can possibly be integrated into non-contrastive objectives such as BYOL (Grill et al., 2020) or VICReg (Bardes et al., 2021) to enrich representations with similarities across samples.

References

Andonian, A., Chen, S., and Hamid, R. Robust cross-modal representation learning with progressive self-distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16430–16441, 2022.

Barbu, A., Mayo, D., Alverio, J., Luo, W., Wang, C., Gutfreund, D., Tenenbaum, J., and Katz, B. Objectnet: A large-scale bias-controlled dataset for pushing the limits of object recognition models. *Advances in neural information processing systems*, 32, 2019.

Bardes, A., Ponce, J., and LeCun, Y. Vicreg: Variance-invariance-covariance regularization for self-supervised learning. *arXiv preprint arXiv:2105.04906*, 2021.

Beyer, L., Hénaff, O. J., Kolesnikov, A., Zhai, X., and Oord, A. v. d. Are we done with imagenet? *arXiv preprint arXiv:2006.07159*, 2020.

Cabannes, V., Bottou, L., Lecun, Y., and Balestriero, R. Active self-supervised learning: A few low-cost relationships are all you need. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 16274–16283, 2023.

Caron, M., Misra, I., Mairal, J., Goyal, P., Bojanowski, P., and Joulin, A. Unsupervised learning of visual features by contrasting cluster assignments. *Advances in neural information processing systems*, 33:9912–9924, 2020.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pp. 1597–1607. PMLR, 2020.

Chopra, S., Hadsell, R., and LeCun, Y. 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, pp. 539–546 vol. 1, 2005. doi: 10.1109/CVPR.2005.202.

Cubuk, E. D., Zoph, B., Mane, D., Vasudevan, V., and Le, Q. V. Autoaugment: Learning augmentation policies from data. *arXiv preprint arXiv:1805.09501*, 2018.

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.

Denize, J., Rabarisoa, J., Orcesi, A., Hérault, R., and Canu, S. Similarity contrastive estimation for self-supervised soft contrastive learning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 2706–2716, 2023.

Dwibedi, D., Aytar, Y., Tompson, J., Sermanet, P., and Zisserman, A. With a little help from my friends: Nearest-neighbor contrastive learning of visual representations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9588–9597, 2021.

Fellbaum, C. *WordNet: An Electronic Lexical Database*. Bradford Books, 1998. URL https://mitpress.mit.edu/9780262561167/.

Fini, E., Astolfi, P., Alahari, K., Alameda-Pineda, X., Mairal, J., Nabi, M., and Ricci, E. Semi-supervised learning made simple with self-supervised clustering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 3187–3197, 2023a.

Fini, E., Astolfi, P., Romero-Soriano, A., Verbeek, J., and Drozdzal, M. Improved baselines for vision-language pre-training. *Transactions on Machine Learning Research*, 2023b. ISSN 2835-8856. URL https://openreview.net/forum?id=a7nvXxNmdV. Featured Certification.

Gao, Y., Liu, J., Xu, Z., Wu, T., Zhang, E., Li, K., Yang, J., Liu, W., and Sun, X. Softclip: Softer cross-modal alignment makes clip stronger. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 1860–1868, 2024.

Grill, J.-B., Strub, F., Altché, F., Tallec, C., Richemond, P., Buchatskaya, E., Doersch, C., Avila Pires, B., Guo, Z., Gheshlaghi Azar, M., et al. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in neural information processing systems*, 33:21271–21284, 2020.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. corr abs/1512.03385 (2015), 2015.

He, K., Fan, H., Wu, Y., Xie, S., and Girshick, R. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9729–9738, 2020.

Hinton, G., Vinyals, O., and Dean, J. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Hoffmann, D. T., Behrmann, N., Gall, J., Brox, T., and Noroozi, M. Ranking info noise contrastive estimation: Boosting contrastive learning via ranked positives. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 897–905, 2022.

Huang, H., Nie, Z., Wang, Z., and Shang, Z. Cross-modal and uni-modal soft-label alignment for image-text retrieval. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 18298–18306, 2024.

Ilharco, G., Wortsman, M., Wightman, R., Gordon, C., Carlini, N., Taori, R., Dave, A., Shankar, V., Namkoong, H., Miller, J., Hajishirzi, H., Farhadi, A., and Schmidt, L. Openclip, July 2021. URL https://doi.org/10.5281/zenodo.5143773. If you use this software, please cite it as below.

Isola, P., Lim, J. J., and Adelson, E. H. Discovering states and transformations in image collections. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1383–1391, 2015.

Khosla, P., Teterwak, P., Wang, C., Sarna, A., Tian, Y., Isola, P., Maschinot, A., Liu, C., and Krishnan, D. Supervised contrastive learning. *Advances in neural information processing systems*, 33:18661–18673, 2020.

Oord, A. v. d., Li, Y., and Vinyals, O. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PMLR, 2021.

Reimers, N. and Gurevych, I. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*, 2019.

Ryali, C. K., Schwab, D. J., and Morcos, A. S. Characterizing and improving the robustness of self-supervised learning through background augmentations. *arXiv preprint arXiv:2103.12719*, 2021.

Schroff, F., Kalenichenko, D., and Philbin, J. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 815–823, 2015.

Sharma, P., Ding, N., Goodman, S., and Soricut, R. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2556–2565, 2018.

Tian, Y., Fan, L., Isola, P., Chang, H., and Krishnan, D. Stablerep: Synthetic images from text-to-image models make strong visual representation learners. *Advances in Neural Information Processing Systems*, 36, 2024.

Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

Xiao, K., Engstrom, L., Ilyas, A., and Madry, A. Noise or signal: The role of image backgrounds in object recognition. *arXiv preprint arXiv:2006.09994*, 2020.

You, Y., Gitman, I., and Ginsburg, B. Large batch training of convolutional networks. *arXiv preprint arXiv:1708.03888*, 2017.

Zheng, M., You, S., Wang, F., Qian, C., Zhang, C., Wang, X., and Xu, C. Ressl: Relational self-supervised learning with weak augmentation. *Advances in Neural Information Processing Systems*, 34:2543–2555, 2021.

A. Appendix

A.1. Related Work

Contrastive learning Various contrastive objectives have been proposed over the years (Chopra et al., 2005; Schroff et al., 2015). More recently, the InfoNCE objective (Oord et al., 2018) has been the most popular choice for self-supervised methods, e.g. SimCLR (Chen et al., 2020) and MoCo (He et al., 2020). InfoNCE objective has also been successfully used to learn vision-language models using CLIP (Radford et al., 2021). The basis of those objectives is to make positive pairs have similar representations, while the negatives, which typically are just all other elements in a batch, should have a different representation. In its original form, InfoNCE is binary, meaning it only works with positive and negative pairs, and does not support degrees of similarity. The positive pairs are usually two augmentations of the same sample, which makes well-tuned augmentations crucial for good performance (Ryali et al., 2021). Dwibedi et al. (2021) estimate positives using nearest neighbors in the latent space instead and therefore can use weaker augmentations, while (Caron et al., 2020) use cluster assignment. A few methods have proposed modifications wherein multiple positive pairs are supported, e.g., Khosla et al. (2020) groups positive by class labels, Hoffmann et al. (2022) propose using WordNet (Fellbaum, 1998) hierarchy to define ranked positive samples, and Tian et al. (2024) uses a generative model to obtain multiple positives for the same concept.

Soft targets Using soft targets provides more learning signal to the model, possibly making it learn better and faster. This has been explored with distillation by Hinton et al. (2015). Soft targets have also been used with InfoNCE in the context of distillation by Zheng et al. (2021) and (Denize et al., 2023), where the target cross-sample similarity comes from the teacher model. Similarly, Fini et al. (2023a) computes soft targets via latent clustering and applies it to semi-supervised learning. Andonian et al. (2022) proposes to use soft targets for CLIP (Radford et al., 2021) training, and calculates the targets via self-distillation. Further soft CLIP objectives are explored by Fini et al. (2023b), who apply label smoothing to obtain soft targets, and Gao et al. (2024), who estimate soft targets by comparing fine-grained image information. Finally, Huang et al. (2024) train CLIP with non-zero cross-sample similarities computed based on pre-trained uni-modal models for text and vision. In this study, we build on the work of (Cabannes et al., 2023) who propose a unifying framework to view SSL and supervised learning objectives as learning with different underlying similarity graphs. We take inspiration from the soft targets literature and propose using a soft graph.

A.2. Limitations

The main limitation of the present work is that constructing the cross-sample similarity graph requires extra data, as well as some extra memory to store it. When the extra data is not available, the only options remaining are to build the graph using the augmentations, self-distillation, or other pre-trained models. The resulting method is also highly dependent on the quality of the graph, as we have seen with conceptual captions datasets.

A.3. Additional results

A.3.1. $\mathbb{X}$-Sample Contrastive introduces only minimal computational overhead

Both for ImageNet and conceptual captions datasets, we don't run the text encoder for each sample we see, and instead precompute the similarity values. For more details, see appendix A.8. Avoiding running the text encoder during model training avoids the extra overhead at the price of some pre-processing. Pre-processing takes less than 2 hours for CC12M when using one GPU, about 30 minutes for CC3M, and less than 5 minutes for ImageNet. To further analyze how much overhead there is, we compare the average time it takes to process one batch for SimCLR and $\mathbb{X}$-CLR. The results are shown in table 5. Overall, we didn't notice any significant difference in the amount of time it takes to train models with the $\mathbb{X}$-CLR objective compared to the regular contrastive objective. To train on ImageNet, we used 8 Nvidia V100s, and each run took about 30 hours. With the same setup, CC3M runs took about 50 hours, and CC12M runs took roughly 9 days.

A.3.2. $\mathbb{X}$-Sample Contrastive can be used to finetune pretrained backbones

We validate whether $\mathbb{X}$-CLR can be used as a finetuning objective for pretrained backbones, given the growing abundance of publicly available backbones. Here, we evaluate a pretrained SimCLR model by finetuning for 10 epochs on ImageNet with $\mathbb{X}$-CLR instead of the original SimCLR contrastive objective. We see in table 4 finetuning with $\mathbb{X}$-CLR improves classification performance on ImageNet by 3.3% and on ImageNet Real by 6.9%. Furthermore, we see by relating samples

Table 4: **𝕏-CLR can be used to finetune pretrained models.**

			Background Decomposition		
	ImageNet	ImageNet Real	Same Class	Mixed	ObjectNet
SimCLR	63.2	67.5	45.5	38.3	12.5
+ 𝕏-CLR finetuning	**66.5**	**74.4**	**53.9**	**50**	**17.4**

Table 5: **Analyzing the computation overhead of the 𝕏-Sample Contrastive objective during training.** 𝕏-CLR introduces nearly no computational overhead compared to SimCLR.

Method	Seconds per batch ImageNet	Seconds per batch CC
SimCLR	0.866 ± 0.008	0.874 ± 0.034
𝕏-CLR	0.866 ± 0.010	0.877 ± 0.032

during the finetuning stage, 𝕏-CLR can disambiguate object foregrounds from backgrounds with grains of 8.4-11.7% on ImageNet-9 as well as improvements on natural object transformations from ObjectNet with a gain of 4.9% after finetuning.

A.3.3. The impact of label quality for fine-grained attribute disambiguation

We show in table 6 how label quality can impact downstream performance on finer-grained attribute disambiguation. We find larger labels from noisy captions degrades performance for fine-grained object attributes in MIT States (Isola et al., 2015) for both Contrastive and CLIP. We find 𝕏-CLR with high quality labels from ImageNet, can outperform models trained on much larger noisier data. Compared to CLIP trained on 12× larger data, 𝕏-CLR achieves 30.9% vs. 23.3% for CLIP on attribute classification and 45.8% vs. 36.9% for CLIP on object classification under different states. To see more details regarding the MIT States evaluation, see appendix A.8.

A.3.4. Analyzing representations learned via 𝕏-Sample Contrastive

Can we improve contrastive learning under data scarcity? To answer this question, we train all three models SimCLR, SupCon, and 𝕏-CLR by varying the number of samples seen for each class in ImageNet. We find 𝕏-CLR, by incorporating information about how classes relate, is able to learn representations that match the performance of SupCon trained with ground truth class labels and outperform SimCLR even when few training samples are available per class as shown in fig. 4a.

KNN clustering To confirm the representations learned via 𝕏-CLR also work well for downstream tasks with non-linear decision boundaries, we perform evaluation using the common K-nearest neighbor (KNN) protocol. The results shown in fig. 4b demonstrate 𝕏-CLR outperforms both SimCLR and SupCon baselines across a range of choices for K. We also show KNN results for models trained on conceptual captions in appendix A.5.

Visualizing the learned graph from 𝕏-Sample Contrastive representations Here we examine whether the learned representations from 𝕏-Sample Contrastive capture semantically meaningful similarities. To do so, we select four groups of three ImageNet classes: felines, dogs, types of balls, and musical instruments. For each pair of classes, we then compare

Table 6: **Label quality matters for fine-grained attribute disambiguation.**

Pretraining	Data Size	Quality	MIT States Attributes	MIT States Objects
CLIP CC3M	3M	Noisy	27.0	40.1
CLIP CC12M	12M	Noisy	23.3	36.9
𝕏-CLR CC3M	3M	Noisy	29.5	40.7
𝕏-CLR CC12M	12M	Noisy	30.1	42.1
𝕏-CLR ImageNet	1M	High	**30.9**	**45.8**

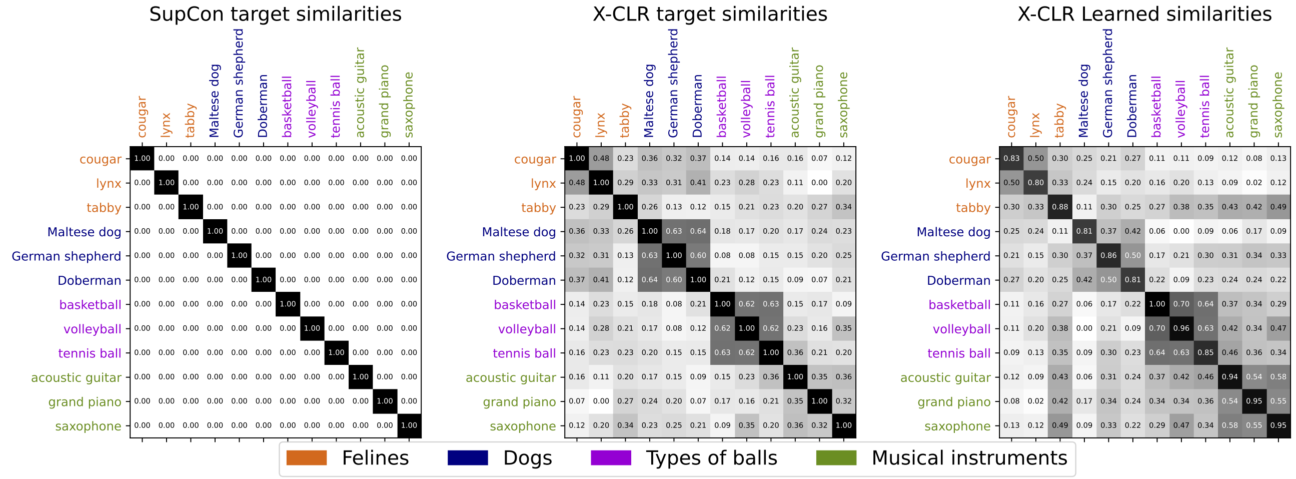
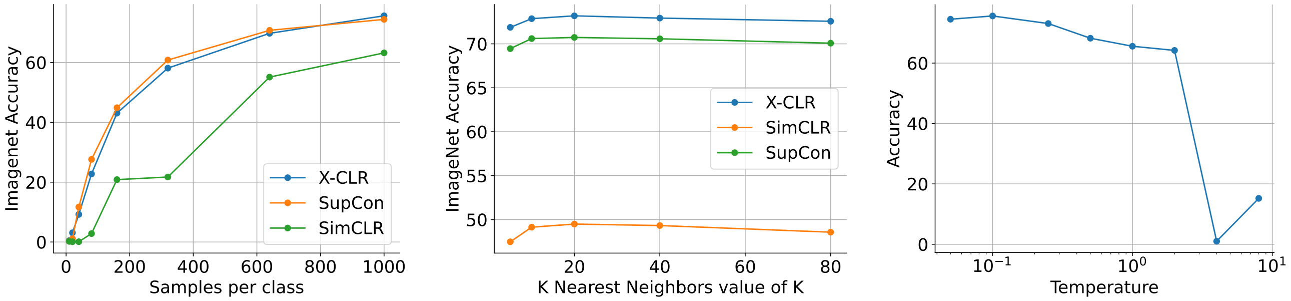


Figure 3: **Visualizing pairwise similarities** SupCon (Khosla et al., 2020) objective does not encourage non-zero similarity between samples of different classes (left), while 𝕏-CLR target similarities take into account semantic closeness within categories such as dogs or types of balls (center). On the right, we see that the trained model successfully learns the soft similarity. For more graphs, see fig. 5.



(a)

(b)

(c)

Figure 4: **(a) 𝕏-Sample Contrastive Loss is data efficient with ImageNet pretraining.** We outperform SimCLR in low data regimes and match Supervised Contrastive trained on ground truth labels at varying levels of data scarcity. **(b) KNN performance ImageNet.** 𝕏-CLR outperforms other methods with KNN probing for a range of values of K. **(c) Sensitivity of 𝕏-Sample Contrastive to temperature.** We test the performance of our method when trained with different values of temperature τ_s on ImageNet data.

Table 7: The effect of the similarity source on the model performance.

Similarity source	ImageNet	ImageNet Real	Same Class	Mixed	ObjectNet
Augmentation graph only (SimCLR)	63.2	67.5	45.5	38.3	12.5
Sentence Transformer ($\mathbb{X}$-CLR)	75.6	81.6	66.5	62.3	27.7
CLIP text encoder	74.4	80.6	67.5	64.2	24.5
LLama2 text encoder	40.9	45.8	38.3	36.0	4.3
Random per class pair, 1 for same class	74.5	80.8	71.0	68.0	26.6
Random per sample pair	0.1	0.1	0.0	0.0	0.0
True class graph (SupCon)	74.4	79.7	63.3	58.8	24.1
Distance in WordNet hierarchy	68.3	74.9	55.7	52.1	21.2

the representation similarities using cosine similarity. A higher average pairwise similarity indicates the model's latent representations encode the classes similarly. In fig. 3 we show the graph of similarities learned after training with $\mathbb{X}$-CLR on ImageNet. We find that the image encoder successfully captures the similarity within the class groups.

The effect of softmax temperature, and inferred similarity graph We also examine the effect of hyperparameter choices. We show the sensitivity of $\mathbb{X}$-CLR to temperature τ_s in fig. 4c on ImageNet. In the limit, when temperature goes to 0, we recover Supervised Contrastive method for ImageNet, or SimCLR in case of conceptual captions. With low temperature, the similarity is 1 only if the captions are exactly the same. As the temperature increases, more weight is put on the soft positives compared to the true positives (i.e. augmentations of the same sample). With high temperature, our method is unstable as too much emphasis is put on the soft positive examples compared to the true positives. We find that the value of 0.1 strikes the optimal balance and provides an improvement over pure Supervised Contrastive objective, while still emphasizing true positives enough. For more details regarding how τ_s changes the objective, see fig. 7b.

We also experiment with different ways of inferring the graph, including using different text encoders, using WordNet (Fellbaum, 1998) hierarchy distance, and the purely random graph. We find that overall, calculating the similarities using the sentence transformer worked the best (Reimers & Gurevych, 2019). A more detailed comparison of different graph sources can be found in appendix A.4.

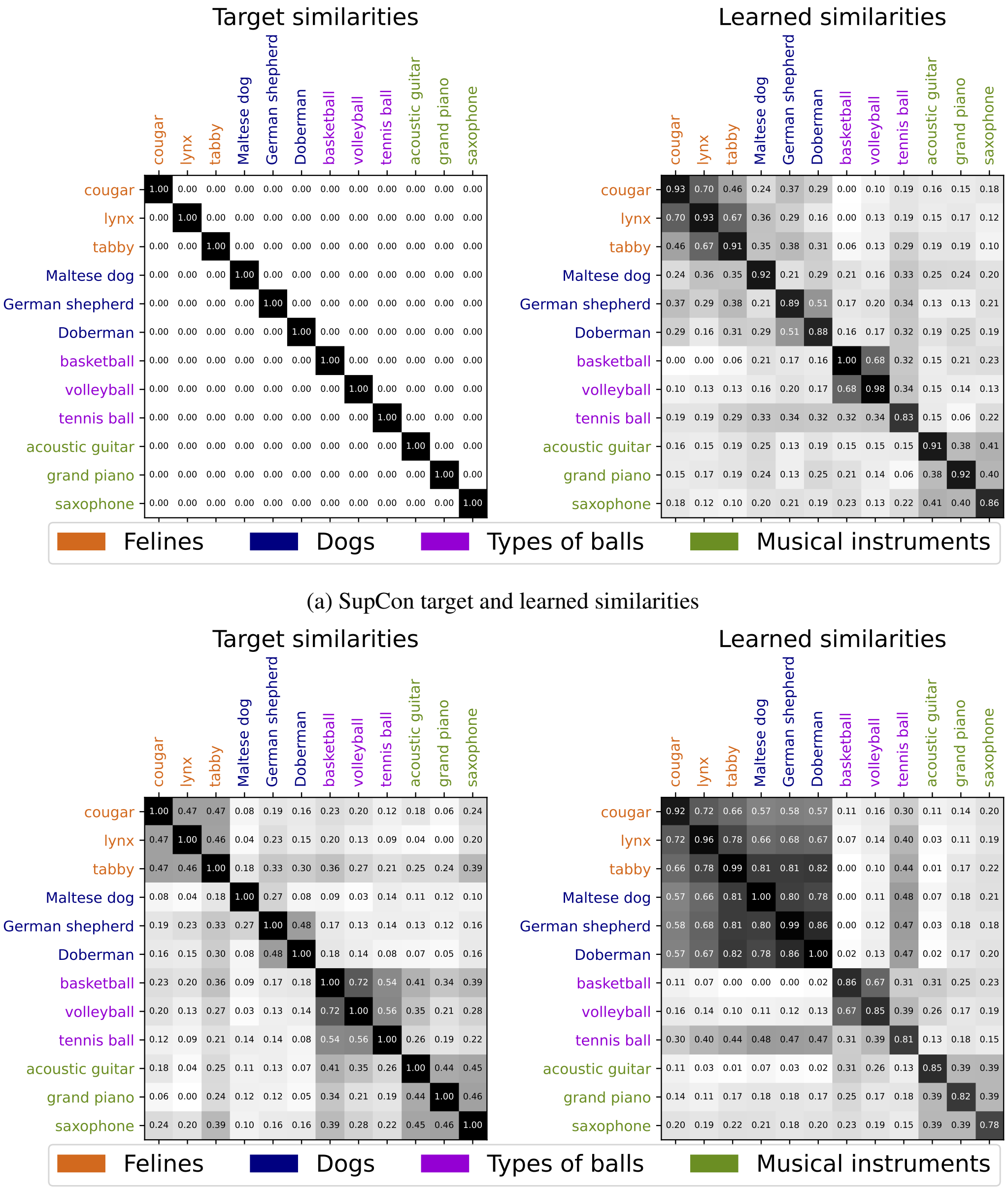
A.4. More learned similarities comparisons

We compare inferring the similarity graph using different text encoders:

- Graph with connections only between samples of the same class (SupCon);
- Graph with connections only between augmentations of the same image (SimCLR);
- Graph where soft similarity is inferred by comparing representations of the sample captions. The representations are computed using the sentence transformer (Reimers & Gurevych, 2019), CLIP text encoder (Radford et al., 2021), LLama2 encoder (Touvron et al., 2023);
- Graph where the connection strength is defined by the distance in WordNet (Fellbaum, 1998) hierarchy;
- Random graph where the cross-sample connections' strengths are fully random;

The results are shown in table 7. We find that overall, the Sentence Transformer graph performs the best, although the CLIP text encoder achieves good performance as well. Interestingly, we find that using WordNet hierarchy distance did not work well. We visualize learned and target similarities for SupCon graph and for the graph built using CLIP text encoder in fig. 5.

Visualising similarities In fig. 3, to visualize learned similarities, for each class we pick 100 examples from the dataset, encode them. Then, to calculate the average learned similarity between two classes, we take the 100 examples for each of the two classes, and calculate the Cartesian product, yielding 10,000 similarities. We take the mean over those 10,000 similarities to represent the average learn similarity for a class pair.



(a) SupCon target and learned similarities

(b) CLIP target and learned similarities

Figure 5: Target and learned similarities for different graphs.

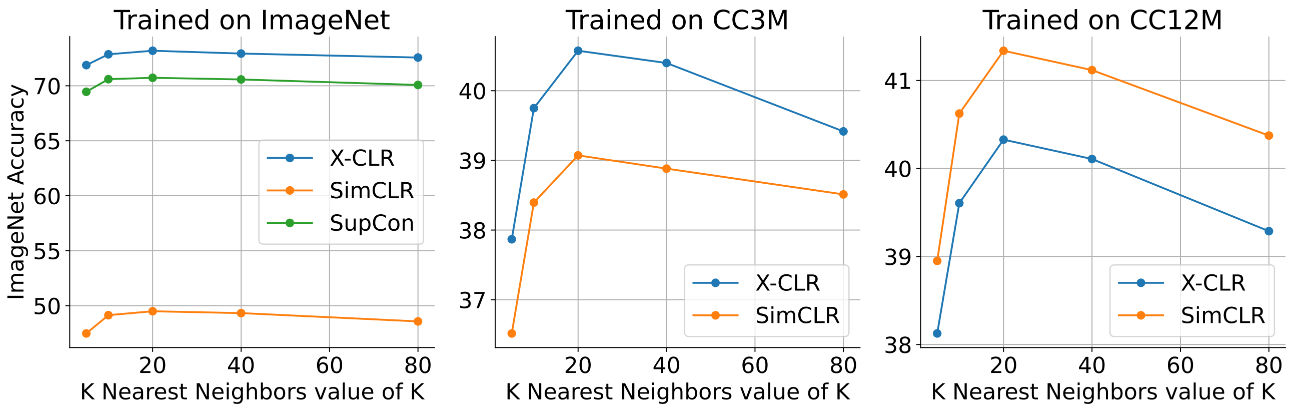


Figure 6: Results of models trained on ImageNet, CC3M, CC12M on ImageNet validation when using KNN classifier.

Table 8: **CLIP on CC3M** We train our own models on CC3M and find that training longer improves the performance. Nevertheless, CLIP struggles with small datasets.

			Background Decomposition		
Method	ImageNet	ImageNet Real	Same Class	Mixed	ObjectNet
CLIP 100 epochs	41.0	47.6	12.5	10.6	7.8
CLIP 32 epochs	36.8	42.0	11.5	9.8	6.0

A.5. KNN evaluation

Apart from testing the models trained on ImageNet using KNN, we also evaluate the models trained on CC3M and CC12M. The results are shown in fig. 6. We see that $\mathbb{X}$-CLR performs better on CC3M, and comparatively with SimCLR when trained on CC12M.

A.6. ImageNet-9 details

ImageNet-9 (Xiao et al., 2020) proposes multiple benchmarks to test model robustness to the background perturbation. In our work, we use "Mixed-Same" and "Mixed-Rand" tasks from ImageNet-9, and refer to them together as "Background Decomposition".

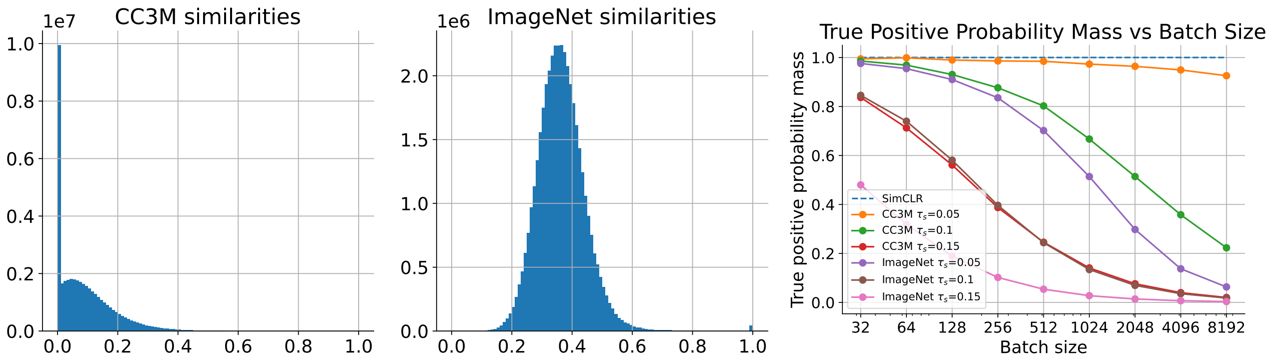
A.7. CLIP details

In CC3M experiments, we train the model from scratch, as OpenCLIP didn't have a checkpoint trained on that dataset. We trained both for 32 and 100 epochs, and found that the model trained for 100 epochs performs better. Since 32 epochs is the default CLIP number of epochs, we also report results for 32 epochs. The results are shown in table 8.

A.8. More training details

For experiments on ImageNet, we follow SupCon and use AutoAugment (Cubuk et al., 2018). All experiments on the ImageNet dataset were run for 100 epochs with 1024 batch size. The learning rate was set to 0.075 for ImageNet models. For experiments on CC3M and CC12M, we used the standard SimCLR augmentations, and a learning rate of 0.1. The rest of the settings were kept the same. We train SimCLR, SupCon and $\mathbb{X}$-CLR using the LARS optimizer (You et al., 2017). In all cases, we use the same ResNet-50, with a two layer projector on top. The output dimension of the projector is 128.

Fetching similarities For ImageNet, since the number of classes is known, we pre-compute the similarity matrix of dimension 1000×1000, and retrieve elements from it depending on the associated class labels for a given sample pair to obtain the similarity value. For conceptual captions, we run the text encoder on the full dataset and save the encodings to disk. Then, when loading an image from disk, we also load the associated encoding of the corresponding caption. The



(a)

(b)

Figure 7: (a) Histograms of the similarities calculated using Sentence Transformer on ImageNet and CC3M. While for ImageNet the average similarity is around 0.35, it is much lower on CC3M, signifying that the graph contains less information for CC3M. (b) Effect of the temperature and batch size on the weight assigned to the true positvie.

similarity matrix for a given batch is then obtained by calculating the Cartesian product of those encodings.

MIT States In order to evaluate on this dataset using linear probing, we split the dataset randomly into two even parts, one used for training the linear layer, the other for evaluation. We train separately to classify objects and attributes.

A.9. Understanding similarities

To understand the graphs we built using for different datasets, we investigate the average cross-sample similarity in the dataset. The result is shown in fig. 7a. We find that CC3M similarities are in general lower, possibly because of lower quality annotations. We also investigate how much weight is assigned to the true positive examples. For SimCLR, it's always 1. For our method, the amount of similarity assigned to other samples in the batch depends on the temperature τ_s, and the batch size. The exact relationship is shown in fig. 7b.