CONTRASTIVE LEARNING WITH HARD NEGATIVE SAMPLES

Anonymous authors
Paper under double-blind review

ABSTRACT

We consider the question: how can you sample good negative examples for contrastive learning? We argue that, as with metric learning, learning contrastive representations benefits from hard negative samples (i.e., points that are difficult to distinguish from an anchor point). The key challenge toward using hard negatives is that contrastive methods must remain unsupervised, making it infeasible to adopt existing negative sampling strategies that use *true* similarity information. In response, we develop a new class of unsupervised methods for selecting hard negative samples where the user can control the amount of hardness. A limiting case of this sampling results in a representation that tightly clusters each class, and pushes different classes as far apart as possible. The proposed method improves downstream performance across multiple modalities, requires only few additional lines of code to implement, and introduces no computational overhead.

1 INTRODUCTION

Owing to their empirical success, contrastive learning methods (Chopra et al., 2005; Hadsell et al., 2006) have become one of the most popular self-supervised approaches for learning representations (Oord et al., 2018; Tian et al., 2019; Chen et al., 2020a). For instance, on certain computer vision tasks unsupervised contrastive learning methods have even outperformed supervised pre-training (Misra & Maaten, 2020; He et al., 2020).

Contrastive learning relies on two key ingredients: notions of similar (positive) (x, x^+) and dissimilar (negative) (x, x^-) pairs of data points. The training objective, typically *noise-contrastive estimation* (Gutmann & Hyvärinen, 2010), guides the learned representation f to map positive pairs to nearby locations, and negative pairs farther apart; other objectives have also been considered (Chen et al., 2020a). The success of the associated methods depends on the design of informative of the positive and negative pairs, which cannot exploit *true* similarity information since there is no supervision.

Much research effort has addressed sampling strategies for positive pairs (Blum & Mitchell, 1998; Xu et al., 2013; Bachman et al., 2019; Tian et al., 2020). For image data, positive sampling strategies often apply transformations that preserve semantic content, e.g., jittering, random cropping, separating color channels, etc. (Chen et al., 2020a;c; Tian et al., 2019). Such transformations have also been effective in learning control policies from raw pixel data (Srinivas et al., 2020). Positive sampling techniques have also been proposed for sentence, audio, and video data (Logeswaran & Lee, 2018; Oord et al., 2018; Purushwalkam & Gupta, 2020; Sermanet et al., 2018).

Surprisingly, the choice of negative pairs has drawn much lesser attention in contrastive learning. Often, given an "anchor" point x, a "negative" x^- is simply sampled uniformly from the training data, independent of how informative it may be for the learned representation. In supervised and metric learning settings, "hard" (negative) examples can help guide a learning method to correct its mistakes more quickly (Schroff et al., 2015; Song et al., 2016). For representation learning, informative negative examples are intuitively those pairs that are mapped nearby but should be far apart. This idea is successfully applied in metric learning, where true pairs of dissimilar points are available, as opposed to unsupervised contrastive learning.

With this motivation, we address the challenge of selecting informative negatives for contrastive representation learning. In particular, we propose a solution that builds a tunable sampling distribution that prefers negative pairs whose representations are currently very similar. This solution faces two

challenges: (1) we do not have access to any true similarity or dissimilarity information; (2) we need an efficient sampling strategy for this tunable distribution. We overcome (1) by building on ideas from positive-unlabeled learning (Elkan & Noto, 2008; Du Plessis et al., 2014), and (2) by designing an efficient, easy to implement importance sampling technique that incurs no computational overhead.

Our theoretical analysis shows that, as a function of the tuning parameter, the optimal representations for our new method place similar inputs in tight clusters, whilst spacing the clusters as far apart as possible. Empirically, our hard negative sampling strategy improves the downstream task performance for image, graph and text data, supporting that indeed, our negative examples are more informative.

Contributions. In summary, we make the following contributions:

1. We formulate to our knowledge the first sampling distribution for hard negative pairs for contrastive representation learning, and an efficient sampling strategy that takes into account the lack of true dissimilarity information;
2. We theoretically analyze the properties of the objective and optimal representations, showing that they capture desired goals for representations;
3. We empirically observe that the proposed sampling method improves the downstream task performance on image, graph and text data.

Before moving onto the problem formulation and our results, we summarize related work below.

1.1 Related Work

Contrastive Representation Learning. Various frameworks for contrastive learning of visual representations have been proposed, including SimCLR (Chen et al., 2020a;b), which uses augmented views of other items in a minibatch as negative samples, and MoCo (He et al., 2020; Chen et al., 2020c), which uses a momentum updated memory bank of old negative representations to enable the use of very large batches of negative samples. Most contrastive methods are unsupervised, however there exist some that use label information (Sylvain et al., 2020; Khosla et al., 2020). Many works study the role of positive pairs, and, e.g., propose to apply large perturbations for images Chen et al. (2020a;c), or argue to minimize the mutual information within positive pairs, apart from relevant information for the ultimate prediction task (Tian et al., 2020). Beyond visual data, contrastive methods have been developed for sentence embeddings (Logeswaran & Lee, 2018), sequential data (Oord et al., 2018; Hénaff et al., 2020), graph (Sun et al., 2020; Hassani & Khasahmadi, 2020; Li et al., 2019) and node representation learning (Velickovic et al., 2019), and learning representations from raw images for off-policy control (Srinivas et al., 2020). The role of negative pairs hase been much less studied. Chuang et al. (2020) propose a method for "debiasing", i.e., correcting for the fact that not all negative pairs may be true negatives. It does so by taking the viewpoint of Positive-Unlabeled learning, and exploits a decomposition of the true negative distribution. Kalantidis et al. (2020) consider applying Mixup to generate hard negatives, and Jin et al. (2018) exploit the specific temporal structure of video to generate negatives for object detection.

Negative Mining in Deep Metric Learning. As opposed to the contrastive representation learning literature, selection strategies for negative samples have been thoroughly studied in (deep) metric learning (Schroff et al., 2015; Song et al., 2016; Harwood et al., 2017; Wu et al., 2017; Ge, 2018; Suh et al., 2019). Most of these works observe that it is helpful to use negative samples that are difficult for the current embedding to discriminate. Schroff et al. (2015) qualify this, observing that some examples are simply too hard, and propose selecting "semi-hard" negative samples. The well known importance of negative samples in metric learning, where (partial) true dissimilarity information is available, raises the question of negative samples in contrastive learning, the subject of this paper.

2 Contrastive Learning Setup

We begin with the setup and the idea of contrastive representation learning. We wish to learn an embedding $f : \mathcal{X} \to \mathbb{S}^{d-1}/t$ that maps an observation x to a point on a hypersphere $\mathbb{S}^{d-1}/t$ in $\mathbb{R}^d$ of radius $1/t$, where t is the "temperature" scaling hyperparameter.

Following the setup of Arora et al. (2019), we assume an underlying set of discrete latent classes $\mathcal{C}$ that represent semantic content, so that similar pairs (x, x^+) have the same latent class. Denoting the distribution over latent classes by $\rho(c)$ for $c \in \mathcal{C}$, we define the joint distribution $p_{x,c}(x, c) = p(x|c)\rho(c)$ whose marginal $p(x)$ we refer to simply as p, and assume $\text{supp}(p) = \mathcal{X}$. For simplicity,

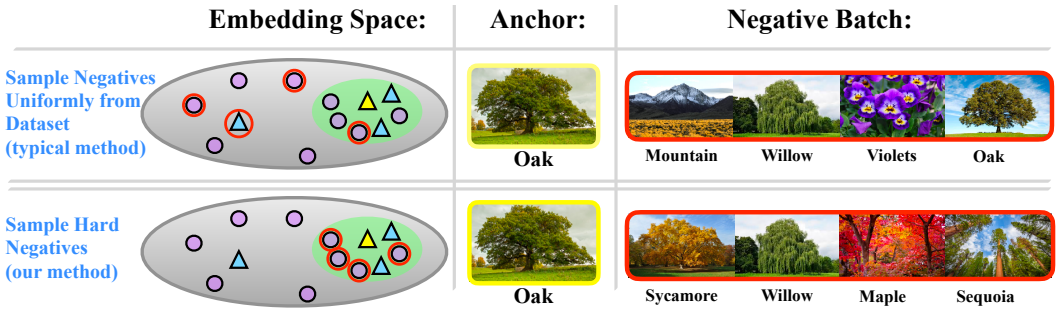

Figure 1: Schematic illustration of negative sampling methods for the example of classifying species of tree. Top row: uniformly samples negative examples (red rings); mostly focuses on very different data points from the anchor (yellow triangle), and may even sample examples from the same class (triangles, vs. circles). Bottom row: Hard negative sampling prefers examples that are (incorrectly) close to the anchor.

we assume $\rho(c) = \tau^+$ is uniform, and let $\tau^- = 1 - \tau^+$ be the probability of another class. Since the class-prior τ^+ is unknown in practice, it must either be treated as a hyperparameter, or estimated (Christoffel et al., 2016; Jain et al., 2016).

Let $h : \mathcal{X} \to \mathcal{C}$ be the true underlying hypothesis that assigns class labels to inputs. We write $x \sim x'$ to denote the label equivalence relation $h(x) = h(x')$. We denote by $p_x^+(x') = p(x'|h(x') = h(x))$, the distribution over points with same label as x, and by $p_x^-(x') = p(x'|h(x') \neq h(x))$, the distribution over points with labels different from x. We drop the subscript x when the context is clear. Following the usual convention, we overload '$\sim$' and also write $x \sim p$ to denote a point sampled from p.

For each data point $x \sim p$, the noise-contrastive estimation (NCE) objective (Gutmann & Hyvärinen, 2010) for learning the representation f uses a *positive* example x^+ with the same label as x, and *negative* examples $\{x_i^-\}_{i=1}^N$ with (supposedly) different labels, $h(x_i^-) \neq h(x)$, sampled from q:

$$\mathbb{E}_{\substack{x\sim p,\ \ x^+\sim p_x^+ \\ \{x_i^-\}_{i=1}^N\sim q}} \left[-\log \frac{e^{f(x)^T f(x^+)}}{e^{f(x)^T f(x^+)} + \frac{Q}{N}\sum_{i=1}^N e^{f(x)^T f(x_i^-)}} \right]. \tag{1}$$

The weighting parameter Q is introduced for the purpose of analysis. When N is finite we take $Q = N$, yielding the usual form of the contrastive objective. The negative sample distribution q is frequently chosen to be the marginal distribution p, or, in practice, an empirical approximation of it (Tian et al., 2019; Chen et al., 2020a;c; He et al., 2020; Chen et al., 2020c; Oord et al., 2018; Hénaff et al., 2020). In this paper we ask: is there a better way to choose q?

3 Hard Negative Sampling

In this section we describe our approach for hard negative sampling. We begin by asking *what makes a good negative sample?* To answer this question we adopt the following two guiding principles:

Principle 1. *q should only sample "negatives" x_i^- whose labels differ from that of the anchor x.*

Principle 2. *The most useful negative samples are ones that the embedding currently believes to be similar to the anchor.*

In short, negative samples that have different label from the anchor, but that are embedded nearby are likely to be most useful and provide significant gradient information during training. In metric learning there is access to true negative pairs, automatically fulfilling the first principle.

In contrastive learning there is no supervision, so upholding Principle 1 is impossible to do exactly. In this paper we propose a method that upholds Principle 1 approximately, and simultaneously combines this idea with the key additional conceptual ingredient of "hardness" (encapsulated in Principle 2). The level of "hardness" in our method can be smoothly adjusted, allowing the user to select the hardness that best trades-off between an improved learning signal from hard negatives, and the harm due to the correction of false negatives being only approximate. This important since the hardest points are those closest to the anchor, and are expected to have a high propensity to have the same

label. Therefore the damage from the approximation not removing all false negatives becomes larger for harder samples, creating the trade-off. As a special case our our method, when the hardness level is tuned fully down, we obtain the method proposed in (Chuang et al., 2020) that only upholds Principle 1 (approximately) but not Principle 2. Finally, beyond Principles 1 and 2, we wish to design an efficient sampling method that does not add additional computational overhead during training.

3.1 Proposed Hard Sampling Method

Our first goal is to design a distribution q on $\mathcal{X}$ that is allowed to depend on the embedding f and the anchor x. From q we sample a batch of negatives $\{x_i^-\}_{i=1}^N$ according to the principles noted above. We propose sampling negatives from the distribution q_β^- defined as

$$q_\beta^-(x^-) := q_\beta(x^- | h(x) \neq h(x^-)), \quad \text{where} \quad q_\beta(x^-) \propto e^{\beta f(x)^\top f(x^-)} \cdot p(x^-),$$

for $\beta \geq 0$. Note that q_β^- and q_β both depend on x, but we suppress the dependance from the notation. The exponential term in q_β is an unnormalized von Mises–Fisher distribution with mean direction $f(x)$ and "concentration parameter" β (Mardia & Jupp, 2000). There are two key components to q_β^-, corresponding to each principle: 1) conditioning on the event $\{h(x) \neq h(x^-)\}$ which guarantees that (x, x^-) correspond to different latent classes (Principle 1); 2) the concentration parameter β term controls the degree by which q_β up-weights points x^- that have large inner product (similarity) to the anchor x (Principle 2). Since f lies on the surface of a hypersphere of radius $1/t$, we have $\|f(x) - f(x')\|^2 = 2/t^2 - 2f(x)^\top f(x')$ so preferring points with large inner product is equivalent to preferring points with small squared Euclidean distance.

Although we have designed q_β^- to have all of the desired components, it is not clear how to sample efficiently from it. To work towards a practical method, note that we can rewrite this distribution by adopting a PU-learning viewpoint (Elkan & Noto, 2008; Du Plessis et al., 2014; Chuang et al., 2020). That is, by conditioning on the event $\{h(x) = h(x^-)\}$ we can split $q_\beta(x^-)$ as

$$q_\beta(x^-) = \tau^- q_\beta^-(x^-) + \tau^+ q_\beta^+(x^-), \tag{2}$$

where $q_\beta^+(x^-) = q_\beta(x^- | h(x) = h(x^-)) \propto e^{\beta f(x)^\top f(x^-)} \cdot p^+(x^-)$. Rearranging equation 2 yields a formula $q_\beta^-(x^-) = \big(q_\beta(x^-) - \tau^+ q_\beta^+(x^-)\big)/\tau^-$ for the negative sampling distribution q_β^- in terms of two distributions that are tractable since we have samples from p and can approximate samples from p^+ using a set of semantics-preserving transformations, as is typical in contrastive learning methods.

It is possible to generate samples from q_β and (approximately from) q_β^+ using rejection sampling. However, rejection sampling involves an algorithmic complication since the procedure for sampling batches must be modified. To avoid this, we instead take an importance sampling approach. To obtain this, first note that fixing the number Q and taking the limit $N \to \infty$ in the objective (1) yields,

$$\mathcal{L}(f, q) = \mathbb{E}_{\substack{x \sim p \\ x^+ \sim p_x^+}} \left[-\log \frac{e^{f(x)^T f(x^+)}}{e^{f(x)^T f(x^+)} + Q \mathbb{E}_{x^- \sim q}[e^{f(x)^T f(x^-)}]} \right]. \tag{3}$$

The original objective (1) can be viewed as a finite negative sample approximation to $\mathcal{L}(f, q)$ (note implicitly $\mathcal{L}(f, q)$ depends on Q) . Inserting $q = q_\beta^-$ and using the rearrangement of equation (2) we obtain the following hardness-biased objective:

$$\mathbb{E}_{\substack{x \sim p \\ x^+ \sim p_x^+}} \left[-\log \frac{e^{f(x)^T f(x^+)}}{e^{f(x)^T f(x^+)} + \frac{Q}{\tau^-}\left(\mathbb{E}_{x^- \sim q_\beta}[e^{f(x)^T f(x^-)}] - \tau^+ \mathbb{E}_{v \sim q_\beta^+}[e^{f(x)^T f(v)}]\right)} \right]. \tag{4}$$

This objective suggests that we need only to approximate *expectations* $\mathbb{E}_{x^- \sim q_\beta}[e^{f(x)^T f(x^-)}]$ and $\mathbb{E}_{v \sim q_\beta^+}[e^{f(x)^T f(v)}]$ over q_β and q_β^+ (rather than explicily sampling). This can be achieved using classical Monte-Carlo importance sampling techniques using samples from p and p^+ as follows:

$$\mathbb{E}_{x^- \sim q_\beta}[e^{f(x)^T f(x^-)}] = \mathbb{E}_{x^- \sim p}[e^{f(x)^T f(x^-)} q_\beta / p] = \mathbb{E}_{x^- \sim p}[e^{(\beta+1) f(x)^T f(x^-)} / Z_\beta],$$
$$\mathbb{E}_{v \sim q_\beta^+}[e^{f(x)^T f(v)}] = \mathbb{E}_{v \sim p^+}[e^{f(x)^T f(v)} q_\beta^+ / p^+] = \mathbb{E}_{v \sim p^+}[e^{(\beta+1) f(x)^T f(v)} / Z_\beta^+],$$

where Z_β, Z_β^+ are the partition functions of q_β and q_β^+ respectively. The right hand terms readily admit empirical approximations by replacing p and p^+ with $\hat{p}(x) = \frac{1}{N}\sum_{i=1}^N \delta_{x_i^-}(x)$ and $\hat{p}^+(x) = \frac{1}{M}\sum_{i=1}^M \delta_{x_i^+}(x)$ respectively (δ_w denotes the Dirac delta function centered at w). The only unknowns left are the partition functions, $Z_\beta = \mathbb{E}_{x^-\sim p}[e^{\beta f(x)^\top f(x^-)}]$ and $Z_\beta^+ = \mathbb{E}_{x^+\sim p^+}[e^{\beta f(x)^\top f(x^+)}]$ which themselves are expectations over p and p^+ and therefore admit empirical estimates,

$$\widehat{Z}_\beta = \frac{1}{N}\sum_{i=1}^N e^{\beta f(x)^\top f(x_i^-)}, \qquad \widehat{Z}_\beta^+ = \frac{1}{M}\sum_{i=1}^M e^{\beta f(x)^\top f(x_i^+)}.$$

It is important to emphasize the simplicity of the implementation of our proposed approach. Since we propose to reweight the objective instead of modifying the sampling procedure, only two extra lines of code are needed to implement our approach, with no additional computational overhead. PyTorch-style pseudocode for the objective is given in Figure 13 in Appendix D.

4 Analysis of Hard Negative Sampling

4.1 Hard Sampling Interpolates Between Marginal and Worst-Case Negatives

Intuitively, the concentration parameter β in our proposed negative sample distribution q_β^- controls the level of "hardness" of the negative samples. As discussed earlier, the debiasing method of Chuang et al. (2020) can be recovered as a special case: taking $\beta = 0$ to obtain the distribution q_0^-. This case amounts to correcting for the fact that some samples in a negative batch sampled from p will have the same label as the anchor. But what interpretation does large β admit? Specifically, what does the distribution q_β^- converge to in the limit $\beta \to \infty$, if anything? We show that in the limit q_β^- approximates an inner solution to the following zero-sum two player game.

$$\inf_f \sup_{q\in\Pi} \left\{ \mathcal{L}(f,q) = \mathbb{E}_{\substack{x\sim p\\ x^+\sim p_x^+}} \left[-\log \frac{e^{f(x)^T f(x^+)}}{e^{f(x)^T f(x^+)} + Q\mathbb{E}_{x^-\sim q}[e^{f(x)^T f(x^-)}]} \right] \right\}. \tag{5}$$

where $\Pi = \{q = q(\cdot; x, f) : \operatorname{supp}\big(q(\cdot; x, f)\big) \subseteq \{x' \in \mathcal{X} : x' \nsim x\}, \forall x \in \mathcal{X}\}$ is the set of distributions with support that is disjoint from points with the same class as x (without loss of generality we assume $\{x' \in \mathcal{X} : x' \nsim x\}$ is non-empty). Since $q = q(\cdot; x, f)$ depends on x and f it can be thought of as a family of distributions. The formal statement is as follows.

Proposition 3. *Let $\mathcal{L}^*(f) = \sup_{q\in\Pi} \mathcal{L}(f,q)$. Then for any $t > 0$ and $f : \mathcal{X} \to \mathbb{S}^{d-1}/t$ we observe the convergence $\mathcal{L}(f, q_\beta^-) \longrightarrow \mathcal{L}^*(f)$ as $\beta \to \infty$.*

Proof. See Appendix A.1. □

To develop a better intuitive understanding of the worst case negative distribution objective $\mathcal{L}^*(f) = \sup_{q\in\Pi} \mathcal{L}(f,q)$, we note that the supremum can be characterized analytically. Indeed,

$$\begin{aligned}\sup_{q\in\Pi} \mathcal{L}(f,q) &= -\mathbb{E}_{\substack{x\sim p\\ x^+\sim p_x^+}} f(x)^T f(x^+) + \sup_{q\in\Pi} \mathbb{E}_{\substack{x\sim p\\ x^+\sim p_x^+}} \log\left\{ e^{f(x)^T f(x^+)} + Q\mathbb{E}_{x^-\sim q}[e^{f(x)^T f(x^-)}] \right\} \\ &= -\mathbb{E}_{\substack{x\sim p\\ x^+\sim p_x^+}} f(x)^T f(x^+) + \mathbb{E}_{\substack{x\sim p\\ x^+\sim p_x^+}} \log\left\{ e^{f(x)^T f(x^+)} + Q \cdot \sup_{q\in\Pi} \mathbb{E}_{x^-\sim q}[e^{f(x)^T f(x^-)}] \right\}.\end{aligned}$$

The supremum over q can be pushed inside the expectation since q is a family of distribution indexed by x, reducing the problem to maximizing $\mathbb{E}_{x^-\sim q}[e^{f(x)^T f(x^-)}]$, which is solved by any q^* whose support is a subset of $\arg\sup_{x^-: x^- \nsim x} e^{f(x)^T f(x^-)}$ if the supremum is attained. However, computing such points involves maximizing a neural network. Instead of taking this challenging route, using q_β^- defines a lower bound by placing higher probability on x^- for which $f(x)^T f(x^-)$ is large. This lower bound becomes tight as $\beta \to \infty$ (Proposition 3).

4.2 OPTIMAL EMBEDDINGS ON THE HYPERSPHERE FOR WORST-CASE NEGATIVE SAMPLES

What desirable properties does an optimal contrastive embedding (global minimizer of $\mathcal{L}$) possess that make the representation generalizable? To study this question, we first analyze the distribution of an optimal embedding f^* on the hypersphere when negatives are sampled from the adversarial worst-case distribution. We consider a different limiting viewpoint of objective (1) as the number of negative samples $N \to \infty$. Following the formulation of Wang & Isola (2020) we take $Q = N$ in (1), and subtract $\log N$. This changes neither the set of minimizers, nor the geometry of the loss surface. Taking the number of negative samples $N \to \infty$ yields the limiting objective,

$$\mathcal{L}_\infty(f, q) = \mathbb{E}_{\substack{x \sim p \\ x^+ \sim p_x^+}} \left[-\log \frac{e^{f(x)^T f(x^+)}}{\mathbb{E}_{x^- \sim q}\left[e^{f(x)^T f(x^-)}\right]} \right]. \tag{6}$$

Theorem 4. *Suppose the downstream task is classification (i.e. $\mathcal{C}$ is finite), and let $\mathcal{L}_\infty^*(f) = \sup_{q \in \Pi} \mathcal{L}_\infty(f, q)$. The infimum $\inf_{f:\text{ measurable}} \mathcal{L}_\infty^*(f)$ is attained, and any f^* achieving the global minimum is such that $f^*(x) = f^*(x^+)$ almost surely. Furthermore, letting $\mathbf{v}_c = f^*(x)$ for any x such that $h(x) = c$ (so $\mathbf{v}_c$ is well defined up to a set of x of measure zero), f^* is characterized as being any solution to the following ball-packing problem,*

$$\max_{\{\mathbf{v}_c \in \mathbb{S}^{d-1}/t\}_{c \in \mathcal{C}}} \sum_{c \in \mathcal{C}} \rho(c) \cdot \min_{c' \neq c} \|\mathbf{v}_c - \mathbf{v}_{c'}\|^2. \tag{7}$$

Proof. See Appendix A.2. □

Interpretation: The first component of the result is that $f^*(x) = f^*(x^+)$ almost surely for an optimal f^*. That is, an optimal embedding f^* must be invariant across pairs of similar inputs x, x^+. The second component is characterizing solutions via the classical Tammes Ball-Packing Problem from geometry Tammes (1930) (Eq. 7) that has only been solved exactly for uniform ρ, for specific of $|\mathcal{C}|$ and typically for $\mathbb{S}^2$ Schütte & Van der Waerden (1951); Musin & Tarasov (2015); Tammes (1930). When the distribution ρ over classes is uniform this problem is solved by a set of $|\mathcal{C}|$ points on the hypersphere such that the average squared-ℓ_2 distance from a point to the nearest other point is as large as possible. In other words, suppose we wish to place $|\mathcal{C}|$ number of balls[1] on $\mathbb{S}^{d-1}$ so that they do not intersect. Then solutions to Tammes' Problem (7) expresses (twice) the largest possible average squared radius that the balls can have. So, we have a ball-packing problem where instead of trying to pack as many balls as possible of a fixed size, we aim to pack a fixed number of balls (one for each class) to have as big radii as possible. Non-uniform ρ adds importance weights to each fixed ball. In summary, solutions of the problem $\min_f \mathcal{L}_\infty^*(f)$ are a maximum margin clustering.

This understanding of global minimizers of $\mathcal{L}_\infty^*(f) = \sup_{q \in \Pi} \mathcal{L}_\infty(f, q)$ can further developed into a better understanding of generalization on downstream tasks. The next result shows that representations that achieve small excess risk on the objective $\mathcal{L}_\infty^*$ still separate clusters well in the sense that a simple 1-nearest neighbor classifier achieves low classification error.

Theorem 5. *Suppose ρ is uniform on $\mathcal{C}$ and f is such that $\mathcal{L}_\infty^*(f) - \inf_{\bar{f}\text{ measurable}} \mathcal{L}_\infty^*(\bar{f}) \leq \varepsilon$ with $\varepsilon \leq 1$. Let $\{\mathbf{v}_c^* \in \mathbb{S}^{d-1}/t\}_{c \in \mathcal{C}}$ be a solution to Problem 7, and define $\xi = \min_{c, c^- : c \neq c^-} \left\|\mathbf{v}_c^* - \mathbf{v}_{c-}^*\right\| > 0$. Then there exists a set of vectors $\{\mathbf{v}_c \in \mathbb{S}^{d-1}/t\}_{c \in \mathcal{C}}$ such that the following 1-nearest neighbor classifier,*

$$\hat{h}(x) = \hat{c}, \quad \textit{where} \quad \hat{c} = \arg\min_{\bar{c} \in \mathcal{C}} \left\|f(x) - \mathbf{v}_{\bar{c}}\right\| \quad \textit{(ties broken arbitrarily)}$$

achieves misclassification risk,

$$\mathbb{P}(\hat{h}(x) \neq c) \leq \frac{8\varepsilon}{(\xi^2 - 2|\mathcal{C}|\,(1 + 1/t)\varepsilon^{1/2})^2}$$

Proof. See Appendix A.3. □

[1] For a manifold $\mathcal{M} \subseteq \mathbb{R}^d$, we say $C \subset \mathcal{M}$ is a ball if it is connected, and there exists a Euclidean ball $\mathcal{B} = \{x \in \mathbb{R}^d : \|x\|_2 \leq R\}$ for which $C = \mathcal{M} \cap \mathcal{B}$.

In particular, $\mathbb{P}(\hat{h}(x) \neq c) = \mathcal{O}(\varepsilon)$ as $\varepsilon \to 0$, and in the limit $\varepsilon \to 0$ we recover Theorem 4 as a special case. The result can easily be generalized to arbitrary ρ by replacing $|\mathcal{C}|$ in the bound by $1/\min_c \rho(c)$. The result also implies that it is possible to build simple classifiers for tasks that involve only a subset of classes from $\mathcal{C}$, or classes that are a union of classes from $\mathcal{C}$. The constant $\xi = \min_{c,c^-:c\neq c^-} \left\|\mathbf{v}_c^* - \mathbf{v}_{c^-}^*\right\| > 0$ is a purely geometrical property of spheres, and describes the minimum separation distance between a set of points that solves the Tammes' ball-packing problem.

5 Empirical Results

Next, we evaluate our hard negative sampling method empirically, and apply it as a modification to state-of-the-art contrastive methods on image, graph, and text data. For all experiments β is treated as a hyper-parameter (see ablations in Figure 2 for more understanding of how to pick β). Values for M and τ^+ must also be determined. We fix $M = 1$ for all experiments, since taking $M > 1$ would increase the number of inputs for the forward-backward pass. Lemma 11 in the appendix gives a theoretical justification for the choice of $M = 1$. Choosing the class-prior τ^+ can be done in two ways: estimating it from data (Christoffel et al., 2016; Jain et al., 2016), or treating it as a hyper-parameter. The first option requires the possession of labeled data *before* contrastive training.

5.1 Image Representations

We begin by testing the hard sampling method on vision tasks using the STL10, CIFAR100 and CIFAR10 data. We use SimCLR (Chen et al., 2020a) as the baseline method. The results in Figure 2 show consistent improvement over SimCLR ($q = p$) and the particular case of our method with $\beta = 0$ proposed in (Chuang et al., 2020) (called debiasing) on STL10 and CIFAR100. For $N = 510$ negative examples per data point we observe absolute improvements of 3% and 7.3% over SimCLR on CIFAR100 and STL10 respectively, and absolute improvements over the best debiased baseline of 1.9% and 3.2%. On CIFAR10 there is a slight improvement for smaller N, which disappears at larger N. See Appendix D.1 for full setup details.

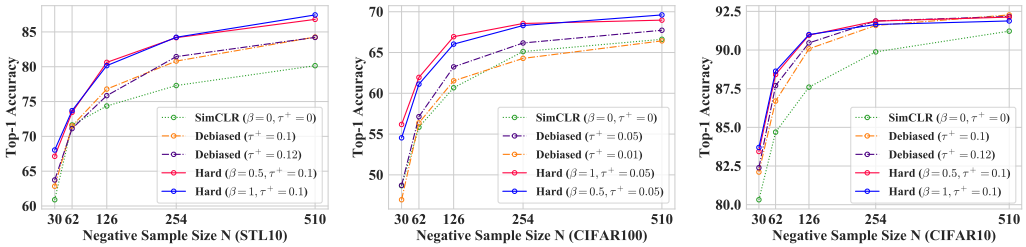

Figure 2: **Classification accuracy on downstream tasks.** Embeddings trained using hard, debiased, and standard ($\beta = 0, \tau^+ = 0$) versions of SimCLR, and evaluated using linear readout accuracy.

5.2 Graph Representations

Second, we consider hard negative sampling in the context of learning graph representations. We use the state-of-the-art InfoGraph method introduced by Sun et al. (2020) as the baseline, which is suitable for downstream graph-level classification. The objective is of a slightly different form from the NCE loss. Because of this we use a generalization of the formulation presented in Section 3 (See Appendix B for details). In doing so, we illustrate that it is easy to adapt our hard sampling method to other contrastive objectives beyond the NCE loss.

Table 3 shows the results of fine-tuning an SVM (Boser et al., 1992; Cortes & Vapnik, 1995) on the fixed, learned embedding for a range of different values of β. Hard sampling does as well as InfoGraph in all cases, and better in 6 out of 8 cases. For ENZYMES and REDDIT, hard negative samples improve the accuracy by 3.2% and 2.4%, respectively, for DD and PTC by $1 - 2\%$, and for IMDB-B and MUTAG by at least 0.5%. Usually, multiple different choices of $\beta > 0$ were competitive with the InfoGraph baseline: 17 out of the 24 values of $\beta > 0$ tried (across all 8 datasets) achieve accuracy as high or better than InfoGraph ($\beta = 0$).

5.3 Sentence Representations

Third, we test hard negative sampling on learning representations of sentences using the *quick-thoughts* (QT) vectors framework introduced by Logeswaran & Lee (2018), which uses adjacent sentences (before/after) as positive samples. Embeddings are trained using the unlabeled BookCorpus

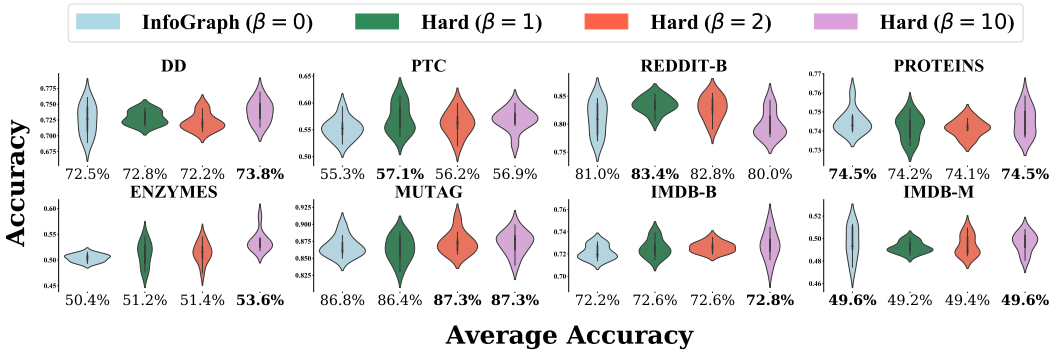

Figure 3: **Classification accuracy on downstream tasks.** We compare graph representations on four classification tasks. Accuracies are obtained by fine-tuning an SVM readout function, and are the average of 10 runs, each using 10-fold cross validation. Results in **bold** indicate best performer.

dataset (Kiros et al., 2015), and evaluated following the protocol of Logeswaran & Lee (2018) on six downstream tasks. The results are reported in Table 1. Hard sampling outperforms or equals the QT baseline in 5 out of 6 cases, the debiased baseline (Chuang et al., 2020) in 4 out of 6, and both in 3 out of 6 cases. Setting $\tau^+ > 0$ led to numerical issues in optimization for hard sampling.

Objective	MR	CR	SUBJ	MPQA	TREC	MSRP (Acc)	MSRP (F1)
QT ($\beta = 0, \tau^+ = 0$)	76.8	81.3	86.6	93.4	**89.8**	73.6	81.8
Debiased ($\tau^+ = 0.01$)	76.2	82.9	86.9	**93.7**	89.1	**74.7**	**82.7**
Hard ($\beta = 1, \tau^+ = 0$)	77.1	82.5	**87.0**	92.9	89.2	73.9	82.2
Hard ($\beta = 2, \tau^+ = 0$)	**77.4**	**83.6**	86.8	93.4	88.7	73.5	82.0

Table 1: **Classification accuracy on downstream tasks.** Sentence representations are learned using quick-thoughts (QT) vectors on the BookCorpus dataset and evaluated on six classification tasks. Evaluation of binary classification tasks (MR, CR, SUBJ, MPQA) uses 10-fold cross validation.

6 A Closer Look at Hard Sampling

6.1 Are Harder Samples Necessarily Better?

By setting β to large values, one can focus on only the hardest samples in a training batch. But is this desirable? Figure 4 (left, middle) shows that for vision problems, taking larger β does not necessarily lead to better representations, corroborating related observations by Wu et al. (2020). In contrast, when one uses true positive pairs during training (green curve, uses label information for positive but not negative pairs), the downstream performance monotonically increases with β until convergence (Figure 4 , middle). Interestingly, this is achieved without using label information for the negative pairs. This observation suggests an explanation for why bigger β hurts performance in practice. Debiasing (conditioning on the event $\{h(x) \neq h(x^-)\}$) using the true p^+ corrects for sampling x^- with the same label as x. However, since in practice we approximate p^+ using a set of data transformations, we can only partially correct. This is harmful for large β since this regime strongly prefers x^- for which $f(x^-)$ is close to $f(x)$, many of whom will have the same label as x if not corrected for. We note also that by annealing β (gradually decreasing β to 0 throughout training; see Appendix D.1 for details) it is possible to be more robust to the choice of initial β, with only minor negative impact on downstream accuracy compared to the best fixed value of β.

6.2 Does Avoiding False Negatives Improve Hard Sampling?

Our proposed hard negative sampling method conditions on the event $\{h(x) \neq h(x^-)\}$ in order to avoid false negatives (termed "debiasing" (Chuang et al., 2020)). But does this help? To test this, we train four embeddings: hard sampling with and without debiasing, and uniform sampling ($\beta = 0$) with and without debiasing. The results in Figure 4 (right) show that hard sampling with debiasing

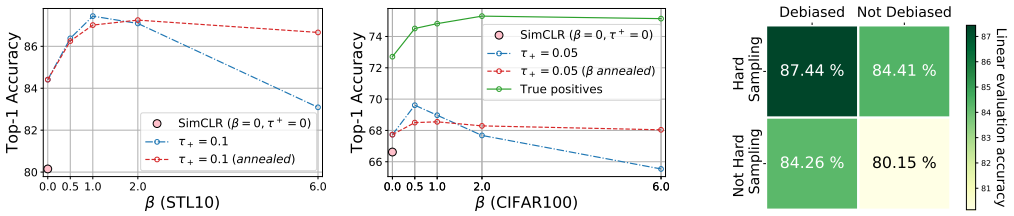

Figure 4: Left: the effect of varying concentration parameter β on linear readout accuracy. Middle: linear readout accuracy as concentration parameter β varyies, in the case of contrastive learning (fully unsupervised), using true positive samples (uses label information), and an annealing method that improves robustness to the choice of β (see Appendix D.1 for details). Right: STL10 linear readout accuracy for hard sampling with and without debiasing, and non-hard sampling ($\beta = 0$) with and without debiasing. Best results come from using both simultaneously.

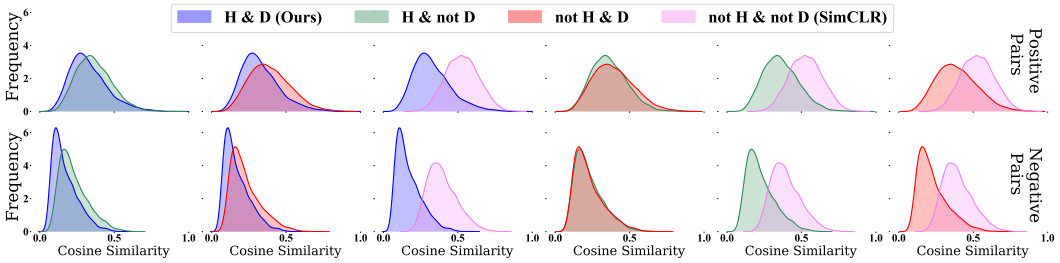

Figure 5: Histograms of cosine similarity of pairs of points with the same label (top) and different labels (bottom) for embeddings trained on STL10 with four different objectives. H=Hard Sampling, D=Debiasing. Histograms overlaid pairwise to allow for convenient comparison.

obtains the highest linear readout accuracy on STL10, only using hard sampling or only debiasing yields (in this case) similar accuracy. All improve over the SimCLR baseline.

Figure 5 compares the histograms of cosine similarities of positive and negative pairs for the four learned representations. The representation trained with hard negatives and debiasing assigns much lower similarity score to a pair of negative samples than other methods. On the other hand, the SimCLR baseline assigns higher cosine similarity scores to pairs of positive samples. However, to discriminate positive and negative pairs, a key property is the amount of *overlap* of positive and negative histograms. Our hard sampling method achieves less overlap than SimCLR, by better trading off higher dissimilarity of negative pairs with less similarity of positive pairs. Similar tradeoffs are observed for the debiased objective, and hard sampling without debiasing.

6.3 How do Hard Negatives Affect Optimization?

Figure 11 (in Appendix C due to space constraints) shows the performance on STL10 and CIFAR100 of SimCLR versus using hard negatives throughout training. We use weighted k-nearest neighbors with $k = 200$ as the classifier and evaluate each model once every five epochs. Hard sampling with $\beta = 1$ leads to much faster training: on STL10 hard sampling takes only 60 epochs to reach the same performance as SimCLR does in 400 epochs. On CIFAR100 hard sampling takes only 125 epochs to reach the same performance as SimCLR does in 400 epochs. We speculate that the speedup is, in part, due to hard negatives providing non-negligible gradient information during training.

7 Conclusion

We argue for the value of hard negatives in unsupervised contrastive representation learning, and introduce a simple hard negative sampling method. Our work connects two major lines of work: contrastive learning, and negative mining in metric learning. Doing so requires overcoming an apparent roadblock: negative mining in metric learning uses pairwise similarity information as a core component, while contrastive learning is unsupervised. Our method enjoys several nice aspects: having desirable theoretical properties, a very simple implementation that requires modifying only a couple of lines of code, not changing anything about the data sampling pipeline, introducing absolutely no extra computational overhead, and handling false negatives in a principled way.

REFERENCES

Sanjeev Arora, Hrishikesh Khandeparkar, Mikhail Khodak, Orestis Plevrakis, and Nikunj Saunshi. A theoretical analysis of contrastive unsupervised representation learning. In *Int. Conference on Machine Learning (ICML)*, pp. 5628–5637, 2019.

Philip Bachman, R Devon Hjelm, and William Buchwalter. Learning representations by maximizing mutual information across views. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 15535–15545, 2019.

Avrim Blum and Tom Mitchell. Combining labeled and unlabeled data with co-training. In *Proceedings of the eleventh annual conference on Computational learning theory*, pp. 92–100, 1998.

Bernhard E Boser, Isabelle M Guyon, and Vladimir N Vapnik. A training algorithm for optimal margin classifiers. In *Proceedings of the fifth annual workshop on Computational learning theory*, pp. 144–152, 1992.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *Int. Conference on Machine Learning (ICML)*, pp. 10709–10719, 2020a.

Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey Hinton. Big self-supervised models are strong semi-supervised learners. *arXiv:2006.10029*, 2020b.

Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv:2003.04297*, 2020c.

Sumit Chopra, Raia Hadsell, and Yann LeCun. Learning a similarity metric discriminatively, with application to face verification. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pp. 539–546, 2005.

Marthinus Christoffel, Gang Niu, and Masashi Sugiyama. Class-prior estimation for learning from positive and unlabeled data. In *Asian Conference on Machine Learning*, pp. 221–236, 2016.

Ching-Yao Chuang, Joshua Robinson, Lin Yen-Chen, Antonio Torralba, and Stefanie Jegelka. Debiased Contrastive Learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Corinna Cortes and Vladimir Vapnik. Support-Vector Networks. *Machine learning*, 20(3):273–297, 1995.

Bill Dolan, Chris Quirk, and Chris Brockett. Unsupervised construction of large paraphrase corpora: Exploiting massively parallel news sources. In *Proceedings of the 20th international conference on Computational Linguistics*, pp. 350. Association for Computational Linguistics, 2004.

Marthinus C Du Plessis, Gang Niu, and Masashi Sugiyama. Analysis of learning from positive and unlabeled data. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 703–711, 2014.

Charles Elkan and Keith Noto. Learning classifiers from only positive and unlabeled data. In *ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 213–220, 2008.

Weifeng Ge. Deep metric learning with hierarchical triplet loss. In *Europ. Conference on Computer Vision (ECCV)*, pp. 269–285, 2018.

Michael Gutmann and Aapo Hyvärinen. Noise-Contrastive Estimation: A new estimation principle for unnormalized statistical models. In *Proc. Int. Conference on Artificial Intelligence and Statistics (AISTATS)*, pp. 297–304, 2010.

Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1735–1742, 2006.

Ben Harwood, Vijay Kumar BG, Gustavo Carneiro, Ian Reid, and Tom Drummond. Smart mining for deep metric learning. In *Int. Conference on Computer Vision (ICCV)*, pp. 2821–2829, 2017.

Kaveh Hassani and Amir Hosein Khasahmadi. Contrastive multi-view representation learning on graphs. In *Int. Conference on Machine Learning (ICML)*, pp. 3451–3461, 2020.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9729–9738, 2020.

Olivier J Hénaff, Aravind Srinivas, Jeffrey De Fauw, Ali Razavi, Carl Doersch, SM Eslami, and Aaron van den Oord. Data-efficient image recognition with contrastive predictive coding. In *Int. Conference on Machine Learning (ICML)*, pp. 6661–6671, 2020.

Minqing Hu and Bing Liu. Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 168–177, 2004.

Shantanu Jain, Martha White, and Predrag Radivojac. Estimating the class prior and posterior from noisy positives and unlabeled data. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 2693–2701, 2016.

SouYoung Jin, Aruni RoyChowdhury, Huaizu Jiang, Ashish Singh, Aditya Prasad, Deep Chakraborty, and Erik Learned-Miller. Unsupervised hard example mining from videos for improved object detection. In *Europ. Conference on Computer Vision (ECCV)*, pp. 307–324, 2018.

Yannis Kalantidis, Mert Bulent Sariyildiz, Noe Pion, Philippe Weinzaepfel, and Diane Larlus. Hard negative mixing for contrastive learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. *arXiv:2004.11362*, 2020.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Int. Conf. on Learning Representations (ICLR)*, 2015.

Ryan Kiros, Yukun Zhu, Russ R Salakhutdinov, Richard Zemel, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. Skip-Thought Vectors. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 3294–3302, 2015.

Yujia Li, Chenjie Gu, Thomas Dullien, Oriol Vinyals, and Pushmeet Kohli. Graph matching networks for learning the similarity of graph structured objects. In *Int. Conference on Machine Learning (ICML)*, pp. 3835–3845, 2019.

Lajanugen Logeswaran and Honglak Lee. An efficient framework for learning sentence representations. In *Int. Conf. on Learning Representations (ICLR)*, 2018.

K. V. Mardia and P. Jupp. *Directional Statistics*. John Wiley and Sons Ltd., second edition, 2000.

Ishan Misra and Laurens van der Maaten. Self-supervised learning of pretext-invariant representations. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6707–6717, 2020.

Christopher Morris, Nils M Kriege, Franka Bause, Kristian Kersting, Petra Mutzel, and Marion Neumann. Tudataset: A collection of benchmark datasets for learning with graphs. In *Graph Representation Learning and Beyond, ICML Workshop*, 2020.

Oleg R Musin and Alexey S Tarasov. The tammes problem for n= 14. *Experimental Mathematics*, 24 (4):460–468, 2015.

Sebastian Nowozin, Botond Cseke, and Ryota Tomioka. f-GAN: Training generative neural samplers using variational divergence minimization. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 271–279, 2016.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv:1807.03748*, 2018.

Bo Pang and Lillian Lee. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42nd annual meeting on Association for Computational Linguistics*, pp. 271. Association for Computational Linguistics, 2004.

Bo Pang and Lillian Lee. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the 43rd annual meeting on association for computational linguistics*, pp. 115–124. Association for Computational Linguistics, 2005.

Senthil Purushwalkam and Abhinav Gupta. Demystifying contrastive self-supervised learning: Invariances, augmentations and dataset biases. *arXiv:2007.13916*, 2020.

Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 815–823, 2015.

K Schütte and BL Van der Waerden. Auf welcher kugel haben 5, 6, 7, 8 oder 9 punkte mit mindestabstand eins platz? *Mathematische Annalen*, 123(1):96–124, 1951.

Pierre Sermanet, Corey Lynch, Yevgen Chebotar, Jasmine Hsu, Eric Jang, Stefan Schaal, Sergey Levine, and Google Brain. Time-contrastive networks: Self-supervised learning from video. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1134–1141, 2018.

Hyun Oh Song, Yu Xiang, Stefanie Jegelka, and Silvio Savarese. Deep metric learning via lifted structured feature embedding. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4004–4012, 2016.

Aravind Srinivas, Michael Laskin, and Pieter Abbeel. CURL: Contrastive unsupervised representations for reinforcement learning. In *Int. Conference on Machine Learning (ICML)*, pp. 10360–10371, 2020.

Yumin Suh, Bohyung Han, Wonsik Kim, and Kyoung Mu Lee. Stochastic class-based hard example mining for deep metric learning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7251–7259, 2019.

Fan-Yun Sun, Jordan Hoffmann, Vikas Verma, and Jian Tang. InfoGraph: Unsupervised and semi-supervised graph-level representation learning via mutual information maximization. In *Int. Conf. on Learning Representations (ICLR)*, 2020.

Tristan Sylvain, Linda Petrini, and Devon Hjelm. Locality and compositionality in zero-shot learning. In *Int. Conf. on Learning Representations (ICLR)*, 2020.

Pieter Merkus Lambertus Tammes. On the origin of number and arrangement of the places of exit on the surface of pollen-grains. *Recueil des travaux botaniques néerlandais*, 27(1):1–84, 1930.

Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive Multiview Coding. In *Europ. Conference on Computer Vision (ECCV)*, pp. 770–786, 2019.

Yonglong Tian, Chen Sun, Ben Poole, Dilip Krishnan, Cordelia Schmid, and Phillip Isola. What makes for good views for contrastive learning? *arXiv:2005.10243*, 2020.

Petar Velickovic, William Fedus, William L Hamilton, Pietro Liò, Yoshua Bengio, and R Devon Hjelm. Deep Graph Infomax. In *Int. Conf. on Learning Representations (ICLR)*, 2019.

Ellen M Voorhees and Donna Harman. Overview of trec 2002. 2002.

Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *Int. Conference on Machine Learning (ICML)*, pp. 9574–9584, 2020.

Janyce Wiebe, Theresa Wilson, and Claire Cardie. Annotating expressions of opinions and emotions in language. *Language resources and evaluation*, 39(2-3):165–210, 2005.

Chao-Yuan Wu, R Manmatha, Alexander J Smola, and Philipp Krahenbuhl. Sampling matters in deep embedding learning. In *Int. Conference on Computer Vision (ICCV)*, pp. 2840–2848, 2017.

Mike Wu, Chengxu Zhuang, Milan Mosse, Daniel Yamins, and Noah Goodman. On mutual information in contrastive learning for visual representations. *arXiv:2005.13149*, 2020.

Chang Xu, Dacheng Tao, and Chao Xu. A survey on multi-view learning. *arXiv:1304.5634*, 2013.

Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? In *Int. Conf. on Learning Representations (ICLR)*, 2019.