A Theoretical Analysis of Contrastive Unsupervised Representation Learning

Sanjeev Arora [1 2] **Hrishikesh Khandeparkar** [1] **Mikhail Khodak** [3] **Orestis Plevrakis** [1] **Nikunj Saunshi** [1]

Abstract

Recent empirical works have successfully used unlabeled data to learn feature representations that are broadly useful in downstream classification tasks. Several of these methods are reminiscent of the well-known word2vec embedding algorithm: leveraging availability of pairs of semantically "similar" data points and "negative samples," the learner forces the inner product of representations of similar pairs with each other to be higher on average than with negative samples. The current paper uses the term *contrastive learning* for such algorithms and presents a theoretical framework for analyzing them by introducing *latent classes* and hypothesizing that semantically similar points are sampled from the same latent class. This framework allows us to show provable guarantees on the performance of the learned representations on the average classification task that is comprised of a subset of the same set of latent classes. Our generalization bound also shows that learned representations can reduce (labeled) sample complexity on downstream tasks. We conduct controlled experiments in both the text and image domains to support the theory.

1. Introduction

This paper concerns *unsupervised representation learning*: using unlabeled data to learn a representation function f such that replacing data point x by feature vector $f(x)$ in new classification tasks reduces the requirement for labeled data. This is distinct from *semi-supervised learning*, where learning can leverage unlabeled as well as labeled data. (Section 7 surveys other prior ideas and models).

[1]Princeton University, Princeton, New Jersey, USA. [2]Institute for Advanced Study, Princeton, New Jersey, USA. [3]Carnegie Mellon University, Pittsburgh, Pennsylvania, USA. Correspondence to: Orestis Plevrakis <orestisp@cs.princeton.edu>, Nikunj Saunshi <nsaunshi@cs.princeton.edu>.

Proceedings of the 36^{th} International Conference on Machine Learning, Long Beach, California, PMLR 97, 2019.

For images, a *proof of existence* for broadly useful representations is the output of the penultimate layer (the one before the softmax) of a powerful deep net trained on ImageNet. In natural language processing (NLP), low-dimensional representations of text – called *text embeddings* – have been computed with unlabeled data (Peters et al., 2018; Devlin et al., 2018). Often the embedding function is trained by using the embedding of a piece of text to predict the surrounding text (Kiros et al., 2015; Logeswaran & Lee, 2018; Pagliardini et al., 2018). Similar methods that leverage similarity in nearby frames in a video clip have had some success for images as well (Wang & Gupta, 2015).

Many of these algorithms are related: they assume access to pairs or tuples (in the form of co-occurrences) of text/images that are more *semantically similar* than randomly sampled text/images, and their objective forces representations to respect this similarity on average. For instance, in order to learn a representation function f for sentences, a simplified version of what Logeswaran & Lee (2018) minimize is the following loss function

$$\mathop{\mathbb{E}}_{x,x^+,x^-}\left[-\log\left(\frac{e^{f(x)^T f(x^+)}}{e^{f(x)^T f(x^+)}+e^{f(x)^T f(x^-)}}\right)\right]$$

where (x, x^+) are a similar pair and x^- is presumably dissimilar to x (often chosen to be a random point) and typically referred to as a *negative sample*. Though reminiscent of past ideas – e.g. kernel learning, metric learning, co-training (Cortes et al., 2010; Bellet et al., 2013; Blum & Mitchell, 1998) – these algorithms lack a theoretical framework *quantifying* when and why they work. While it seems intuitive that minimizing such loss functions should lead to representations that capture 'similarity,' formally it is unclear why the learned representations should do well on downstream *linear classification tasks* – their somewhat mysterious success is often treated as an obvious consequence. To analyze this success, a framework must connect 'similarity' in unlabeled data with the semantic information that is implicitly present in downstream tasks.

We propose the term *Contrastive Learning* for such methods and provide a new conceptual framework with minimal assumptions[1]. Our main contributions are the following:

[1]The alternative would be to make assumptions about generative models of data. This is difficult for images and text.

1. We formalize the notion of semantic similarity by introducing *latent classes*. Similar pairs are assumed to be drawn from the same latent class. A downstream task is comprised of a subset of these latent classes.
2. Under this formalization, we prove that a representation function f learned from a function class $\mathcal{F}$ by contrastive learning has low *average* linear classification loss if $\mathcal{F}$ contains a function with low unsupervised loss. Additionally, we show a generalization bound for contrastive learning that depends on the Rademacher complexity of $\mathcal{F}$. After highlighting inherent limitations of negative sampling, we show sufficient properties of $\mathcal{F}$ which allow us to overcome these limitations.
3. Using insights from the above framework, we provide a novel extension of the algorithm that can leverage larger blocks of similar points than pairs, has better theoretical guarantees, and performs better in practice.

Ideally, one would like to show that contrastive learning always gives representations that *compete* with those learned from the same function class with plentiful labeled data. Our formal framework allows a rigorous study of such questions: we show a simple counterexample that prevents such a blanket statement without further assumptions. However, if the representations are well-concentrated and the mean classifier (Definition 2.1) has good performance, we can show a weaker version of the ideal result (Corollary 5.1.1). Sections 2 and 3 give an overview of the framework and the results, and subsequent sections deal with the analysis. Related work is discussed in Section 7 and Section 8 describes experimental verification and support for our framework.

2. Framework for Contrastive Learning

We first set up notation and describe the framework for unlabeled data and classification tasks that will be essential for our analysis. Let $\mathcal{X}$ denote the set of all possible data points. Contrastive learning assumes access to *similar* data in the form of pairs (x, x^+) that come from a distribution $\mathcal{D}_{sim}$ as well as k i.i.d. *negative samples* $x_1^-, x_2^-, \ldots, x_k^-$ from a distribution $\mathcal{D}_{neg}$ that are presumably unrelated to x. Learning is done over $\mathcal{F}$, a class of *representation functions* $f : \mathcal{X} \to \mathbb{R}^d$, such that $\|f(\cdot)\| \leq R$ for some $R > 0$.

Latent Classes

To formalize the notion of semantically similar pairs (x, x^+), we introduce the concept of *latent classes*.

Let $\mathcal{C}$ denote the set of all latent classes. Associated with each class $c \in \mathcal{C}$ is a probability distribution $\mathcal{D}_c$ over $\mathcal{X}$.

Roughly, $\mathcal{D}_c(x)$ captures how relevant x is to class c. For example, $\mathcal{X}$ could be natural images and c the class "dog" whose associated $\mathcal{D}_c$ assigns high probability to images containing dogs and low/zero probabilities to other images. Classes can overlap arbitrarily.[2] Finally, we assume a distribution ρ over the classes that characterizes how these classes naturally occur in the unlabeled data. Note that we make no assumption about the functional form of $\mathcal{D}_c$ or ρ.

Semantic Similarity

To formalize similarity, we assume similar data points x, x^+ are i.i.d. draws from the same class distribution $\mathcal{D}_c$ for some class c picked randomly according to measure ρ. Negative samples are drawn from the marginal of $\mathcal{D}_{sim}$:

$$\mathcal{D}_{sim}(x, x^+) = \mathop{\mathbb{E}}_{c\sim\rho} \mathcal{D}_c(x)\mathcal{D}_c(x^+) \tag{1}$$

$$\mathcal{D}_{neg}(x^-) = \mathop{\mathbb{E}}_{c\sim\rho} \mathcal{D}_c(x^-) \tag{2}$$

Since classes are allowed to overlap and/or be fine-grained, this is a plausible formalization of "similarity." As the identity of the class in not revealed, we call it unlabeled data. Currently empirical works heuristically identify such similar pairs from co-occurring image or text data.

Supervised Tasks

We now characterize the tasks that a representation function f will be tested on. A $(k+1)$-way[3] supervised task $\mathcal{T}$ consists of distinct classes $\{c_1, \ldots, c_{k+1}\} \subseteq \mathcal{C}$. The labeled dataset for the task $\mathcal{T}$ consists of m i.i.d. draws from the following process:

A label $c \in \{c_1, ..., c_{k+1}\}$ is picked according to a distribution $\mathcal{D}_{\mathcal{T}}$. Then, a sample x is drawn from $\mathcal{D}_c$. Together they form a labeled pair (x, c) with distribution

$$\mathcal{D}_{\mathcal{T}}(x, c) = \mathcal{D}_c(x)\mathcal{D}_{\mathcal{T}}(c) \tag{3}$$

A key subtlety in this formulation is that the classes in downstream tasks and their associated data distributions $\mathcal{D}_c$ are the same as in the unlabeled data. This provides a path to formalizing how capturing similarity in unlabeled data can lead to quantitative guarantees on downstream tasks. $\mathcal{D}_{\mathcal{T}}$ is assumed to be uniform[4] for theorems in the main paper.

Evaluation Metric for Representations

The quality of the representation function f is evaluated by its performance on a multi-class classification task $\mathcal{T}$ using *linear classification*. For this subsection, we fix a task $\mathcal{T} = \{c_1, ..., c_{k+1}\}$. A multi-class classifier for $\mathcal{T}$ is a function $g : \mathcal{X} \to \mathbb{R}^{k+1}$ whose output coordinates are indexed by the classes c in task $\mathcal{T}$.

The loss incurred by g on point $(x, y) \in \mathcal{X} \times \mathcal{T}$ is defined

[2] An image of a dog by a tree can appear in both $\mathcal{D}_{dog}$ & $\mathcal{D}_{tree}$.
[3] We use k as the number of negative samples later.
[4] We state and prove the general case in the Appendix.

as $\ell(\{g(x)_y - g(x)_{y'}\}_{y' \neq y})$, which is a function of a k-dimensional vector of differences in the coordinates. The two losses we will consider in this work are the standard hinge loss $\ell(\boldsymbol{v}) = \max\{0, 1+\max_i\{-\boldsymbol{v}_i\}\}$ and the logistic loss $\ell(\boldsymbol{v}) = \log_2(1 + \sum_i \exp(-\boldsymbol{v}_i))$ for $\boldsymbol{v} \in \mathbb{R}^k$. Then the supervised loss of the classifier g is

$$L_{sup}(\mathcal{T}, g) \coloneqq \mathop{\mathbb{E}}_{(x,c)\sim\mathcal{D}_{\mathcal{T}}} \left[\ell\big(\{g(x)_c - g(x)_{c'}\}_{c' \neq c}\big)\right]$$

To use a representation function f with a linear classifier, a matrix $W \in \mathbb{R}^{(k+1)\times d}$ is trained and $g(x) = Wf(x)$ is used to evaluate classification loss on tasks. Since the best W can be found by fixing f and training a linear classifier, we abuse notation and define the *supervised loss* of f on $\mathcal{T}$ to be the loss when the best W is chosen for f:

$$L_{sup}(\mathcal{T}, f) = \inf_{W \in \mathbb{R}^{(k+1)\times d}} L_{sup}(\mathcal{T}, Wf) \tag{4}$$

Crucial to our results and experiments will be a specific W where the rows are the means of the representations of each class which we define below.

Definition 2.1 (Mean Classifier)**.** *For a function f and task $\mathcal{T} = (c_1, \ldots, c_{k+1})$, the mean classifier is W^μ whose c^{th} row is the mean μ_c of representations of inputs with label c: $\mu_c \coloneqq \mathop{\mathbb{E}}_{x\sim\mathcal{D}_c}[f(x)]$. We use $L^\mu_{sup}(\mathcal{T}, f) \coloneqq L_{sup}(\mathcal{T}, W^\mu f)$ as shorthand for its loss.*

Since contrastive learning has access to data with latent class distribution ρ, it is natural to have better guarantees for tasks involving classes that have higher probability in ρ.

Definition 2.2 (Average Supervised Loss)**.** *Average loss for a function f on $(k+1)$-way tasks is defined as*

$$L_{sup}(f) \coloneqq \mathop{\mathbb{E}}_{\{c_i\}_{i=1}^{k+1}\sim\rho^{k+1}} \left[L_{sup}(\{c_i\}_{i=1}^{k+1}, f) \mid c_i \neq c_j\right]$$

The average supervised loss of its mean classifier *is*

$$L^\mu_{sup}(f) \coloneqq \mathop{\mathbb{E}}_{\{c_i\}_{i=1}^{k+1}\sim\rho^{k+1}} \left[L^\mu_{sup}(\{c_i\}_{i=1}^{k+1}, f) \mid c_i \neq c_j\right]$$

Contrastive Learning Algorithm

We describe the training objective for contrastive learning: the choice of loss function is dictated by the ℓ used in the supervised evaluation, and k denotes number of negative samples used for training. Let $(x, x^+) \sim \mathcal{D}_{sim}$, $(x_1^-, .., x_k^-) \sim \mathcal{D}_{neg}^k$ as defined in Equations (1) and (2).

Definition 2.3 (Unsupervised Loss)**.** *The population loss is*

$$L_{un}(f) \coloneqq \mathbb{E}\left[\ell\left(\left\{f(x)^T\left(f(x^+) - f(x_i^-)\right)\right\}_{i=1}^k\right)\right] \tag{5}$$

and its empirical counterpart with M samples $(x_j, x_j^+, x_{j1}^-, ..., x_{jk}^-)_{j=1}^M$ from $\mathcal{D}_{sim} \times \mathcal{D}_{neg}^k$ is

$$\widehat{L}_{un}(f) = \frac{1}{M}\sum_{j=1}^{M} \ell\left(\left\{f(x_j)^T\left(f(x_j^+) - f(x_{ji}^-)\right)\right\}_{i=1}^k\right) \tag{6}$$

Note that, by the assumptions of the framework described above, we can now express the unsupervised loss as

$$L_{un}(f) = \mathop{\mathbb{E}}_{\substack{c^+, c_i^- \\ \sim\rho^{k+1}}} \mathop{\mathbb{E}}_{\substack{x, x^+\sim\mathcal{D}_{c^+}^2 \\ x_i^-\sim\mathcal{D}_{c_i^-}}} \left[\ell\left(\left\{f(x)^T\left(f(x^+) - f(x_i^-)\right)\right\}\right)\right]$$

The algorithm to learn a representation function from $\mathcal{F}$ is to find a function $\widehat{f} \in \arg\min_{f\in\mathcal{F}} \widehat{L}_{un}(f)$ that minimizes the empirical unsupervised loss. This function $\widehat{f}$ can be subsequently used for supervised linear classification tasks. In the following section we proceed to give an overview of our results that stem from this framework.

3. Overview of Analysis and Results

What can one *provably* say about the performance of $\widehat{f}$? As a first step we show that L_{un} is like a "surrogate" for L_{sup} by showing that $L_{sup}(f) \leq \alpha L_{un}(f), \forall f \in \mathcal{F}$, suggesting that minimizing L_{un} makes sense. This lets us show a bound on the supervised performance $L_{sup}(\widehat{f})$ of the representation learned by the algorithm. For instance, when training with one negative sample, the performance on average binary classification has the following guarantee:

Theorem 4.1 *(Informal binary version).*

$$L_{sup}(\widehat{f}) \leq \alpha L_{un}(f) + \eta\, Gen_M + \delta \qquad \forall f \in \mathcal{F}$$

where α, η, δ are constants depending on the distribution ρ and $Gen_M \to 0$ as $M \to \infty$. When ρ is uniform and $|\mathcal{C}| \to \infty$, we have that $\alpha, \eta \to 1,\ \delta \to 0$.

At first glance, this bound seems to offer a somewhat complete picture: *When the number of classes is large, if the unsupervised loss can be made small by $\mathcal{F}$, then the supervised loss of $\widehat{f}$, learned using finite samples, is small.*

While encouraging, this result still leaves open the question: Can $L_{un}(f)$ indeed be made small on reasonable datasets using function classes $\mathcal{F}$ of interest, even though the similar pair and negative sample can come from the same latent class? We shed light on this by upper-bounding $L_{un}(f)$ by two components: (a) the loss $L^{\neq}_{un}(f)$ for the case where the positive and negative samples are from different classes; (b) a notion of deviation $s(f)$, within each class.

Theorem 4.5 *(Informal binary version).*

$$L_{sup}(\widehat{f}) \leq L^{\neq}_{un}(f) + \beta s(f) + \eta\, Gen_M \qquad \forall f \in \mathcal{F}$$

for constants β, η that depend on the distribution ρ. Again, when ρ is uniform and $|\mathcal{C}| \to \infty$ we have $\beta \to 0, \eta \to 1$.

This bound lets us infer the following: *if the class $\mathcal{F}$ is rich enough to contain a function f for which $L^{\neq}_{un}(f) + \beta s(f)$ is*

low, then $\widehat{f}$ has high supervised performance. Both $L_{un}^{\neq}(f)$ and $s(f)$ can potentially be made small for rich enough $\mathcal{F}$.

Ideally, however, one would want to show that $\widehat{f}$ can compete on classification tasks with every $f \in \mathcal{F}$

$$\textit{(Ideal Result):} \quad L_{sup}(\widehat{f}) \leq \alpha L_{sup}(f) + \eta \, Gen_M \quad (7)$$

Unfortunately, we show in Section 5.1 that the algorithm can pick something far from the optimal f. However, we extend Theorem 4.5 to a bound similar to (7) (where the classification is done using the mean classifier) under assumptions about the intraclass concentration of f and about its mean classifier having high margin.

Sections 6.1 and 6.2 extend our results to the more complicated setting where the algorithm uses k negative samples (5) and note an interesting behavior: increasing the number of negative samples beyond a threshold can hurt the performance. In Section 6.3 we show a novel extension of the algorithm that utilizes larger blocks of similar points. Finally, we describe experiments in Section 8 and Appendix D to validate components of our framework and corroborate our suspicion that the mean classifier of representations learned using labeled data has good classification performance.

4. Guaranteed Average Binary Classification

To provide the main insights, we prove the algorithm's guarantee when we use only 1 negative sample ($k = 1$). For this section, let $L_{sup}(f)$ and $L_{sup}^{\mu}(f)$ be as in Definition 2.2 for binary tasks. We will refer to the two classes in the supervised task as well as the unsupervised loss as c^+, c^-. Let $\mathcal{S} = \{x_j, x_j^+, x_j^-\}_{j=1}^M$ be our training set sampled from the distribution $\mathcal{D}_{sim} \times \mathcal{D}_{neg}$ and $\widehat{f} \in \arg\min_{f \in \mathcal{F}} \widehat{L}_{un}(f)$.

4.1. Upper Bound using Unsupervised Loss

Let $f_{|\mathcal{S}} = \left(f_t(x_j), f_t(x_j^+), f_t(x_j^-)\right)_{j \in [M], t \in [d]} \in \mathbb{R}^{3dM}$ be the restriction on $\mathcal{S}$ for any $f \in \mathcal{F}$. Then, the statistical complexity measure relevant to the estimation of the representations is the following Rademacher average

$$\mathcal{R}_{\mathcal{S}}(\mathcal{F}) = \mathop{\mathbb{E}}_{\sigma \sim \{\pm 1\}^{3dM}} \left[\sup_{f \in \mathcal{F}} \langle \sigma, f_{|\mathcal{S}} \rangle \right]$$

Let $\tau = \mathop{\mathbb{E}}_{c, c' \sim \rho^2} \mathbf{1}\{c = c'\}$ be the probability that two classes sampled independently from ρ are the same.

Theorem 4.1. *With probability at least $1 - \delta$, for all $f \in \mathcal{F}$*

$$L_{sup}^{\mu}(\widehat{f}) \leq \frac{1}{(1-\tau)}(L_{un}(f) - \tau) + \frac{1}{(1-\tau)} Gen_M$$

where

$$Gen_M = O\left(R \frac{\mathcal{R}_{\mathcal{S}}(\mathcal{F})}{M} + R^2 \sqrt{\frac{\log \frac{1}{\delta}}{M}} \right)$$

Remark. *The complexity measure $\mathcal{R}_S(\mathcal{F})$ is tightly related to the labeled sample complexity of the classification tasks. For the function class $\mathcal{G} = \{w^T f(\cdot) | f \in \mathcal{F}, \, \|w\| \leq 1\}$ that one would use to solve a binary task from scratch using labeled data, it can be shown that $\mathcal{R}_{\mathcal{S}}(\mathcal{F}) \leq d\mathcal{R}_{\mathcal{S}}(\mathcal{G})$, where $\mathcal{R}_{\mathcal{S}}(\mathcal{G})$ is the usual Rademacher complexity of $\mathcal{G}$ on $\mathcal{S}$ (Definition 3.1 from (Mohri et al., 2018)).*

We state two key lemmas needed to prove the theorem.

Lemma 4.2. *With probability at least $1 - \delta$ over the training set $\mathcal{S}$, for all $f \in \mathcal{F}$*

$$L_{un}(\widehat{f}) \leq L_{un}(f) + Gen_M$$

We prove Lemma 4.2 in Appendix A.3.

Lemma 4.3. *For all $f \in \mathcal{F}$*

$$L_{sup}^{\mu}(f) \leq \frac{1}{(1-\tau)} \left(L_{un}(f) - \tau\right)$$

Proof. The key idea in the proof is the use of Jensen's inequality. Unlike the unsupervised loss which uses a random point from a class as a classifier, using the mean of the class as the classifier should only make the loss lower. Let $\mu_c = \mathop{\mathbb{E}}_{x \sim \mathcal{D}_c} f(x)$ be the mean of the class c.

$$\begin{aligned}
L_{un}(f) &= \mathop{\mathbb{E}}_{\substack{(x,x^+) \sim \mathcal{D}_{sim} \\ x^- \sim \mathcal{D}_{neg}}} \left[\ell(f(x)^T(f(x^+) - f(x^-)))\right] \\
&=^{(a)} \mathop{\mathbb{E}}_{\substack{c^+, c^- \sim \rho^2 \\ x \sim \mathcal{D}_{c^+}}} \mathop{\mathbb{E}}_{\substack{x^+ \sim \mathcal{D}_{c^+} \\ x^- \sim \mathcal{D}_{c^-}}} \left[\ell(f(x)^T(f(x^+) - f(x^-)))\right] \\
&\geq^{(b)} \mathop{\mathbb{E}}_{c^+, c^- \sim \rho^2} \mathop{\mathbb{E}}_{x \sim \mathcal{D}_{c^+}} \left[\ell(f(x)^T(\mu_{c^+} - \mu_{c^-}))\right] \\
&=^{(c)} (1-\tau) \mathop{\mathbb{E}}_{c^+, c^- \sim \rho^2} [L_{sup}^{\mu}(\{c^+, c^-\}, f) | c^+ \neq c^-] + \tau \\
&=^{(d)} (1-\tau) L_{sup}^{\mu}(f) + \tau
\end{aligned}$$

where (a) follows from the definitions in (1) and (2), (b) follows from the convexity of ℓ and Jensen's inequality by taking the expectation over x^+, x^- inside the function, (c) follows by splitting the expectation into the cases $c^+ = c^-$ and $c^+ \neq c^-$, from symmetry in c^+ and c^- in sampling and since classes in tasks are uniformly distributed (general distributions are handled in Appendix B.1). Rearranging terms completes the proof. □

Proof of Theorem 4.1. The result follows directly by applying Lemma 4.3 for $\widehat{f}$ and finishing up with Lemma 4.2. □

One could argue that if $\mathcal{F}$ is rich enough such that L_{un} can be made small, then Theorem 4.1 suffices. However, in the next section we explain that unless $\tau \ll 1$, this may not always be possible and we show one way to alleviate this.

4.2. Price of Negative Sampling: Class Collision

Note first that the unsupervised loss can be decomposed as

$$L_{un}(f) = \tau L^{=}_{un}(f) + (1-\tau) L^{\neq}_{un}(f) \tag{8}$$

where $L^{\neq}_{un}(f)$ is the loss suffered when the similar pair and the negative sample come from different classes.

$$L^{\neq}_{un}(f) = \mathop{\mathbb{E}}_{\substack{c^+,c^-\sim\rho^2 \\ x,x^+\sim\mathcal{D}^2_{c^+} \\ x^-\sim\mathcal{D}_{c^-}}} \left[\ell(f(x)^T(f(x^+) - f(x^-)))|c^+ \neq c^-\right]$$

and $L^{=}_{un}(f)$ is when they come from the *same class*. Let ν be a distribution over $\mathcal{C}$ with $\nu(c) \propto \rho^2(c)$, then

$$\begin{aligned} L^{=}_{un}(f) &= \mathop{\mathbb{E}}_{\substack{c\sim\nu \\ x,x^+,x^-\sim\mathcal{D}^3_c}} \left[\ell(f(x)^T(f(x^+) - f(x^-)))\right] \\ &\geq \mathop{\mathbb{E}}_{c\sim\nu, x\sim\mathcal{D}_c} \left[\ell(f(x)^T(\mu_c - \mu_c))\right] = 1 \end{aligned}$$

by Jensen's inequality again, which implies $L_{un}(f) \geq \tau$. In general, without any further assumptions on f, $L_{un}(f)$ can be far from τ, rendering the bound in Theorem 4.1 useless. However, as we will show, the magnitude of $L^{=}_{un}(f)$ can be controlled by the intraclass deviation of f. Let $\Sigma(f,c)$ the covariance matrix of $f(x)$ when $x \sim \mathcal{D}_c$. We define a notion of intraclass deviation as follows:

$$s(f) \coloneqq \mathop{\mathbb{E}}_{c\sim\nu} \left[\sqrt{\|\Sigma(f,c)\|_2} \mathop{\mathbb{E}}_{x\sim\mathcal{D}_c} \|f(x)\|\right] \tag{9}$$

Lemma 4.4. *For all $f \in \mathcal{F}$,*

$$L^{=}_{un}(f) - 1 \leq c' s(f)$$

where c' is a positive constant.

We prove Lemma 4.4 in Appendix A.1. Theorem 4.1 combined with Equation (8) and Lemma 4.4 gives the following result.

Theorem 4.5. *With probability at least $1-\delta$, $\forall f \in \mathcal{F}$*

$$L_{sup}(\widehat{f}) \leq L^{\mu}_{sup}(\widehat{f}) \leq L^{\neq}_{un}(f) + \beta\, s(f) + \eta\, Gen_M$$

where $\beta = c' \frac{\tau}{1-\tau}$, $\eta = \frac{1}{1-\tau}$ and c' is a constant.

The above bound highlights two sufficient properties of the function class for unsupervised learning to work: when the function class $\mathcal{F}$ is rich enough to contain *some* f with low $\beta s(f)$ as well as low $L^{\neq}_{un}(f)$ then $\widehat{f}$, the empirical minimizer of the unsupervised loss – learned using sufficiently large number of samples – will have good performance on supervised tasks (low $L_{sup}(\widehat{f})$).

5. Towards Competitive Guarantees

We provide intuition and counter-examples for why contrastive learning does not always pick the best supervised representation $f \in \mathcal{F}$ and show how our bound captures these. Under additional assumptions, we show a competitive bound where classification is done using the mean classifier.

5.1. Limitations of contrastive learning

The bound provided in Theorem 4.5 might not appear as the most natural guarantee for the algorithm. Ideally one would like to show a bound like the following: for all $f \in \mathcal{F}$,

$$\textit{(Ideal 1):} \quad L_{sup}(\widehat{f}) \leq \alpha L_{sup}(f) + \eta\, Gen_M \tag{10}$$

for constants α, η and generalization error Gen_M. This guarantees that $\widehat{f}$ is competitive against the *best* f on the average binary classification task. However, the bound we prove has the following form: for all $f \in \mathcal{F}$,

$$L^{\mu}_{sup}(\widehat{f}) \leq \alpha L^{\neq}_{un}(f) + \beta s(f) + \eta\, Gen_M$$

To show that this discrepancy is not an artifact of our analysis but rather stems from limitations of the algorithm, we present two examples in Figure 1. Our bound appropriately captures these two issues individually owing to the large values of $L^{\neq}_{un}(f)$ or $s(f)$ in each case, for the optimal f.

In Figure 1a, we see that there is a direction on which f_1 can be projected to perfectly separate the classes. Since the algorithm takes inner products between the representations, it inevitably considers the spurious components along the orthogonal directions. This issue manifests in our bound as the term $L^{\neq}_{un}(f_1)$ being high even when $s(f_1) = 0$. Hence, contrastive learning will not always work when the only guarantee we have is that $\mathcal{F}$ can make L_{sup} small.

This should not be too surprising, since we show a relatively strong guarantee – a bound on L^{μ}_{sup} for the *mean classifier* of $\widehat{f}$. This suggests a natural stronger assumption that $\mathcal{F}$ can make L^{μ}_{sup} small (which is observed experimentally in Appendix D for function classes of interest) and raises the question of showing a bound that looks like the following: for all $f \in \mathcal{F}$,

$$\textit{(Ideal 2):} \quad L^{\mu}_{sup}(\widehat{f}) \leq \alpha L^{\mu}_{sup}(f) + \eta Gen_M \tag{11}$$

without accounting for any intraclass deviation – recall that $s(f)$ captures a notion of this deviation in our bound. However this is not true: high intraclass deviation may not imply high $L^{\mu}_{sup}(f)$, but can make $L^{=}_{un}(f)$ (and thus $L_{un}(f)$) high, resulting in the failure of the algorithm. Consequently, the term $s(f)$ also increases while $L^{\neq}_{un}$ does not necessarily have to. This issue, apparent in Figure 1b, shows that a guarantee like (11) cannot be shown without further assumptions.

(a) Mean is bad (b) High intraclass variance

Figure 1. In both examples we have uniform distribution over classes $\mathcal{C} = \{c_1, c_2\}$, blue and red points are in c_1 and c_2 respectively and $\mathcal{D}_{c_i}$ is uniform over the points of c_i. In the first figure we have one point per class, while in the second we have two points per class. Let $\mathcal{F} = \{f_0, f_1\}$ where f_0 maps all points to $(0, 0)$ and f_1 is defined in the figure. In both cases, using the hinge loss, $L_{sup}(f_1) = 0$, $L_{sup}(f_0) = 1$ and in the second case $L^{\mu}_{sup}(f_1) = 0$. However, in both examples the algorithm will pick f_0 since $L_{un}(f_0) = 1$ but $L_{un}(f_1) = \Omega(r^2)$.

5.2. Competitive Bound via Intraclass Concentration

We saw that $L^{\mu}_{sup}(f)$ being small does not imply low $L^{\mu}_{sup}(\widehat{f})$, if f is not concentrated within the classes. In this section we show that when there is an f that has intraclass concentration in a strong sense (sub-Gaussianity) and can separate classes with high margin (on average) with the mean classifier, then $L^{\mu}_{sup}(\widehat{f})$ will be low.

Let $\ell_\gamma(x) = (1 - \frac{x}{\gamma})_+$ be the hinge loss with margin γ and $L^{\mu}_{\gamma,sup}(f)$ be $L^{\mu}_{sup}(f)$ with the loss function ℓ_γ.

Lemma 5.1. *For $f \in \mathcal{F}$, if the random variable $f(X)$, where $X \sim D_c$, is σ^2-sub-Gaussian in every direction for every class c and has maximum norm $R = max_{x \in \mathcal{X}} \|f(x)\|$, then for all $\epsilon > 0$,*

$$L^{\neq}_{un}(f) \leq \gamma L^{\mu}_{\gamma,sup}(f) + \epsilon$$

where $\gamma = 1 + c' R\sigma\sqrt{\log \frac{R}{\epsilon}}$ and c' is some constant.

The proof of Lemma 5.1 is provided in the Appendix A.2. Using Lemma 5.1 and Theorem 4.5, we get the following:

Corollary 5.1.1. *For all $\epsilon > 0$, with probability at least $1 - \delta$, for all $f \in \mathcal{F}$,*

$$L^{\mu}_{sup}(\widehat{f}) \leq \gamma(f) L^{\mu}_{\gamma(f),sup}(f) + \beta s(f) + \eta Gen_M + \epsilon$$

where $\gamma(f)$ is as defined in Lemma 5.1, $\beta = c' \frac{\tau}{1-\tau}$, $\eta = \frac{\tau}{1-\tau}$ and c' is a constant.

6. Utilizing Multiple Negative Samples and Block Similarity

In this section we explore two extensions to our analysis. First, in Section 6.1, inspired by empirical works like Logeswaran & Lee (2018) that often use more than one negative sample for every similar pair, we show provable guarantees for this case by careful handling of class collision. Additionally, in Section 6.2 we show simple examples where increasing negative samples beyond a certain threshold can hurt contrastive learning. Second, in Section 6.3, we explore a modified algorithm that leverages access to *blocks* of similar data, rather than just pairs and show that it has stronger guarantees as well as performs better in practice.

6.1. Guarantees for k Negative Samples

Here the algorithm utilizes k negative samples $x^-_1, ..., x^-_k$ drawn i.i.d. from $\mathcal{D}_{neg}$ for every positive sample pair x, x^+ drawn from $\mathcal{D}_{sim}$ and minimizes (6). As in Section 4, we prove a bound for $\widehat{f}$ of the following form:

Theorem 6.1. *(Informal version) For all $f \in \mathcal{F}$*

$$\mathcal{L}_{sup}(\widehat{f}) \leq \mathcal{L}^{\mu}_{sup}(\widehat{f}) \leq \alpha L^{\neq}_{un}(f) + \beta s(f) + \eta\, Gen_M$$

where $L^{\neq}_{un}(f)$ and Gen_M are extensions of the corresponding terms from Section 4 and $s(f)$ remains unchanged. The formal statement of the theorem and its proof appears in Appendix B.1. The key differences from Theorem 4.5 are β and the distribution of tasks in $\mathcal{L}_{sup}$ that we describe below. The *coefficient* β of $s(f)$ increases with k, e.g. when ρ is uniform and $k \ll |\mathcal{C}|$, $\beta \approx \frac{k}{|\mathcal{C}|}$.

The *average supervised loss* that we bound is

$$\mathcal{L}_{sup}(\widehat{f}) := \mathop{\mathbb{E}}_{\mathcal{T} \sim \mathcal{D}} \left[L_{sup}(\mathcal{T}, \widehat{f}) \right]$$

where $\mathcal{D}$ is a distribution over tasks, defined as follows: sample $k + 1$ classes $c^+, c^-_1, \ldots, c^-_k \sim \rho^{k+1}$, conditioned on the event that c^+ does not also appear as a negative sample. Then, set $\mathcal{T}$ to be the set of distinct classes in $\{c^+, c^-_1, \ldots, c^-_k\}$. $\mathcal{L}^{\mu}_{sup}(\widehat{f})$ is defined by using $L^{\mu}_{sup}(\mathcal{T}, \widehat{f})$.

Remark. *Bounding $\mathcal{L}_{sup}(\widehat{f})$ directly gives a bound for average $(k+1)$-wise classification loss $L_{sup}(\widehat{f})$ from Definition 2.2, since $L_{sup}(\widehat{f}) \leq \mathcal{L}_{sup}(\widehat{f})/p$, where p is the probability that the $k + 1$ sampled classes are distinct. For $k \ll |\mathcal{C}|$ and $\rho \approx$ uniform, these metrics are almost equal.*

We also extend our competitive bound from Section 5.2 for the above $\widehat{f}$ in Appendix B.2.

6.2. Effect of Excessive Negative Sampling

The standard belief is that increasing the number of negative samples always helps, at the cost of increased computational costs. In fact for Noise Contrastive Estimation (NCE) (Gutmann & Hyvärinen, 2010), which is invoked to explain the success of negative sampling, increasing negative samples has shown to provably improve the asymptotic variance of

the learned parameters. However, we find that such a phenomenon does not always hold for contrastive learning – larger k can hurt performance for the same inherent reasons highlighted in Section 5.1, as we illustrate next.

When ρ is close to uniform and the number of negative samples is $k = \Omega(|\mathcal{C}|)$, frequent class collisions can prevent the unsupervised algorithm from learning the representation $f \in \mathcal{F}$ that is optimal for the supervised problem. In this case, owing to the contribution of $s(f)$ being high, a large number of negative samples could hurt. This problem, in fact, can arise even when the number of negative samples is much smaller than the number of classes. For instance, if the best representation function $f \in \mathcal{F}$ groups classes into t "clusters",[5] such that f cannot contrast well between classes from the same cluster, then $L_{un}^{\neq}$ will contribute to the unsupervised loss being high even when $k = \Omega(t)$. We illustrate, by examples, how these issues can lead to picking suboptimal $\widehat{f}$ in Appendix C. Experimental results in Figures D.1a and D.1b also suggest that larger negative samples hurt performance beyond a threshold, confirming our suspicions.

6.3. Blocks of Similar Points

Often a dataset consists of *blocks* of similar data instead of just pairs: a block consists of $x_0, x_1, \dots x_b$ that are i.i.d. draws from a class distribution D_c for a class $c \sim \rho$. In text, for instance, paragraphs can be thought of as a *block* of sentences sampled from the same latent class. How can an algorithm leverage this additional structure?

We propose an algorithm that uses two blocks: one for positive samples $x, x_1^+, \dots, x_b^+$ that are i.i.d. samples from $c^+ \sim \rho$ and another one of negative samples $x_1^-, \dots x_b^-$ that are i.i.d. samples from $c^- \sim \rho$. Our proposed algorithm then minimizes the following loss:

$$L_{un}^{block}(f) := \mathbb{E}\left[\ell\left(f(x)^T\left(\frac{\sum_i f(x_i^+)}{b} - \frac{\sum_i f(x_i^-)}{b}\right)\right)\right] \tag{12}$$

This is reminiscent of the average of embeddings used in *word2vec*, where blocks correspond to windows of consecutive words. To understand why this loss function make sense, recall that the connection between L_{sup}^{μ} and L_{un} was made in Lemma 4.3 by applying Jensen's inequality. Thus, the algorithm that uses the average of the positive and negative samples in blocks as a proxy for the classifier instead of just one point each should have a strictly better bound owing to the Jensen's inequality getting tighter. We formalize this intuition below. Let τ be as defined in Section 4.

[5]This can happen when $\mathcal{F}$ is not rich enough.

Proposition 6.2. $\forall f \in \mathcal{F}$

$$L_{sup}(f) \leq \frac{1}{1-\tau}\left(L_{un}^{block}(f) - \tau\right) \leq \frac{1}{1-\tau}\left(L_{un}(f) - \tau\right)$$

This bound tells us that L_{un}^{block} is a better surrogate for L_{sup}, making it a more attractive choice than L_{un} when larger blocks are available[6]. The algorithm can be extended, analogously to Equation (5), to handle more than one negative block. Experimentally we find that minimizing L_{un}^{block} instead of L_{un} can lead to better performance and our results are summarized in Section 8.1. We defer the proof of Proposition 6.2 to Appendix A.4.

7. Related Work

The contrastive learning framework is inspired by several empirical works, some of which were mentioned in the introduction. The use of co-occurring words as semantically similar points and negative sampling for learning word embeddings was introduced in Mikolov et al. (2013). Subsequently, similar ideas have been used by Logeswaran & Lee (2018) and Pagliardini et al. (2018) for sentences representations and by Wang & Gupta (2015) and Jean et al. (2018) for images. Notably the sentence representations learned by the *quick thoughts (QT)* method in Logeswaran & Lee (2018) that we analyze has state-of-the-art results on many text classification tasks. Previous attempts to explain negative sampling (Dyer, 2014) use the idea of Noise Contrastive Estimation (NCE) (Gutmann & Hyvärinen, 2010) by relying on the assumption that data distribution belongs to some known parametric family and enables them to consider a broader class of distributions for negative sampling. Ranking-based NCE is similar to our loss function and is analyzed in Ma & Collins (2018) using similar assumptions to NCE. The mean classifier that appears in our guarantees is of significance in meta-learning and is a core component of ProtoNets (Snell et al., 2017). The N-pair loss in Sohn (2016) is the same as our contrastive loss with multiple negative samples.

Our data model for similarity is reminiscent of the one in *co-training* (Blum & Mitchell, 1998). They assume access to pairs of "views" with the same label that are conditionally independent given the label. Our unlabeled data model can be seen as a special case of theirs, where the two views have the same conditional distributions. However, they additionally assume access to some labeled data (semi-supervised), while we learn representations using only unlabeled data, which can be subsequently used for classification when labeled data is presented. *Two-stage kernel learning* (Cortes et al., 2010; Kumar et al., 2012) is similar in this sense: in the first stage, a positive linear combination of some base

[6]Rigorous comparison of the generalization errors is left for future work.

kernels is learned and is then used for classification in the second stage; they assume access to labels in both stages. *Similarity/metric learning* (Bellet et al., 2012; 2013) learns a linear feature map that gives low distance to similar points and high to dissimilar. While they identify dissimilar pairs using labels, due to lack of labels we resort to negative sampling and pay the price of class collision. While these works analyze linear function classes, we can handle arbitrarily powerful representations. Learning of representations that are broadly useful on a distribution of tasks is done in *multitask learning*, specifically in the *learning-to-learn model* (Maurer et al., 2016) but using labeled data.

Recently Hazan & Ma (2016) proposed "assumption-free" methods for representation learning via MDL/compression arguments, but do not obtain any guarantees comparable to ours on downstream classification tasks. As noted by Arora & Risteski (2017), this compression approach has to preserve *all* input information (e.g. preserve every pixel of the image) which seems suboptimal.

8. Experimental Results

We report experiments in text and vision domains supporting our theory. Since contrastive learning has already shown to obtain state-of-the-art results on text classification by *quick thoughts* (QT) in Logeswaran & Lee (2018), most of our experiments are conducted to corroborate our theoretical analysis. The controlled experiments are described in Appendix D. We also show that our extension to similarity blocks in Section 6.3 can improve QT on a real-world task.

Datasets: Two datasets were used in the controlled experiments. (1) The CIFAR-100 dataset (Krizhevsky, 2009) consisting of 32x32 images categorized into 100 classes with a 50000/10000 train/test split. (2) Lacking an appropriate NLP dataset with large number of classes, we create the Wiki-3029 dataset, consisting of 3029 Wikipedia articles as the classes and 200 sentences from each article as samples. The train/dev/test split is 70%/10%/20%. To test our method on a more standard task, we also use the unsupervised part of the IMDb review corpus (Maas et al., 2011), which consists of 560K sentences from 50K movie reviews. Representations trained using this corpus are evaluated on the supervised IMDb binary classification task, consisting of training and testing set with 25K reviews each. For CIFAR-100 and Wiki-3029, we simulate the data generation process described in Section 2. Details are provided in Appendix D.

8.1. Effect of Block Size

As suggested in Section 6.3, a natural extension to the model would be access to blocks of similar points. We refer to our method of minimizing the loss in (12) as *CURL* for *Contrastive Unsupervised Representation Learning* and perform experiments on CIFAR-100, Wiki-3029, and IMDb. In Table 1 we see that for CIFAR-100 and Wiki-3029, increasing block size yields an improvement in classification accuracy. For IMDb, as is evident in Table 1, using larger blocks provides a clear benefit and the method does better than QT, which has state-of-the-art performance on many tasks. A thorough evaluation of CURL and its variants on other unlabeled datasets is left for future work.

Table 1. Effect of larger block size on representations. For CIFAR-100 and WIKI-3029 we measure the average binary classification accuracy. IMDB representations are tested on IMDB supervised task. CURL is our large block size contrastive method, QT is the algorithm from (Logeswaran & Lee, 2018). For larger block sizes, QT uses all pairs within a block as similar pairs. We use the same GRU architecture for both CURL and QT for a fair comparison.

DATASET	METHOD	$b = 2$	$b = 5$	$b = 10$
CIFAR-100	CURL	88.1	89.6	89.7
WIKI-3029	CURL	96.6	97.5	97.7
IMDB	CURL	89.2	89.6	89.7
	QT	86.5	87.7	86.7

9. Conclusion

Contrastive learning methods have been empirically successful at learning useful feature representations. We provide a new conceptual framework for thinking about this form of learning, which also allows us to formally treat issues such as guarantees on the quality of the learned representations. The framework gives fresh insights into what guarantees are possible and impossible, and shapes the search for new assumptions to add to the framework that allow tighter guarantees. The framework currently ignores issues of efficient minimization of various loss functions, and instead studies the interrelationships of their minimizers as well as sample complexity requirements for training to generalize, while clarifying what generalization means in this setting. Our approach should be viewed as a first cut; possible extensions include allowing tree structure – more generally metric structure – among the latent classes. Connections to meta-learning and transfer learning may arise.

We use experiments primarily to illustrate and support the new framework. But one experiment on sentence embeddings already illustrates how fresh insights derived from our framework can lead to improvements upon state-of-the-art models in this active area. We hope that further progress will follow, and that our theoretical insights will begin to influence practice, including design of new heuristics to identify semantically similar/dissimilar pairs.

Acknowledgments

This work is supported by NSF, ONR, the Simons Foundation, the Schmidt Foundation, Mozilla Research, Amazon Research, DARPA, and SRC. We thank Michael Collins, Rong Ge, Elad Hazan, Sham Kakade, Karthik Narasimhan, Karan Singh, and Yi Zhang for helpful discussions and suggestions.

References

Arora, S. and Risteski, A. Provable benefits of representation learning. arXiv, 2017.

Bellet, A., Habrard, A., and Sebban, M. Similarity learning for provably accurate sparse linear classification. *arXiv preprint arXiv:1206.6476*, 2012.

Bellet, A., Habrard, A., and Sebban, M. A survey on metric learning for feature vectors and structured data. *CoRR*, abs/1306.6709, 2013.

Blum, A. and Mitchell, T. Combining labeled and unlabeled data with co-training. In *Proceedings of the Eleventh Annual Conference on Computational Learning Theory*, COLT' 98, 1998.

Chung, J., Gulcehre, C., Cho, K., and Bengio, Y. Gated feedback recurrent neural networks. In *Proceedings of the 32Nd International Conference on International Conference on Machine Learning - Volume 37*, ICML'15. JMLR.org, 2015.

Cortes, C., Mohri, M., and Rostamizadeh, A. Two-stage learning kernel algorithms. 2010.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 10 2018.

Dyer, C. Notes on noise contrastive estimation and negative sampling. *CoRR*, abs/1410.8251, 2014. URL http://arxiv.org/abs/1410.8251.

Gutmann, M. and Hyvärinen, A. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pp. 297–304, 2010.

Hazan, E. and Ma, T. A non-generative framework and convex relaxations for unsupervised learning. In *Neural Information Processing Systems*, 2016.

Jean, N., Wang, S., Samar, A., Azzari, G., Lobell, D., and Ermon, S. Tile2vec: Unsupervised representation learning for spatially distributed data. *arXiv preprint arXiv:1805.02855*, 2018.

Kiros, R., Zhu, Y., Salakhutdinov, R., Zemel, R. S., Torralba, A., Urtasun, R., and Fidler, S. Skip-thought vectors. In *Neural Information Processing Systems*, 2015.

Krizhevsky, A. Learning multiple layers of features from tiny images. Technical report, 2009.

Kumar, A., Niculescu-Mizil, A., Kavukcoglu, K., and Daumé, H. A binary classification framework for two-stage multiple kernel learning. In *Proceedings of the 29th International Coference on International Conference on Machine Learning*, ICML'12, 2012.

Logeswaran, L. and Lee, H. An efficient framework for learning sentence representations. In *Proceedings of the International Conference on Learning Representations*, 2018.

Ma, Z. and Collins, M. Noise contrastive estimation and negative sampling for conditional models: Consistency and statistical efficiency. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2018.

Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., and Potts, C. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the ACL: Human Language Technologies*, 2011.

Maurer, A. A vector-contraction inequality for rademacher complexities. In *International Conference on Algorithmic Learning Theory*, pp. 3–17. Springer, 2016.

Maurer, A., Pontil, M., and Romera-Paredes, B. The benefit of multitask representation learning. *J. Mach. Learn. Res.*, 2016.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. Distributed representations of words and phrases and their compositionality. In *Neural Information Processing Systems*, 2013.

Mohri, M., Rostamizadeh, A., and Talwalkar, A. *Foundations of machine learning*. MIT press, 2018.

Pagliardini, M., Gupta, P., and Jaggi, M. Unsupervised learning of sentence embeddings using compositional n-gram features. Proceedings of the North American Chapter of the ACL: Human Language Technologies, 2018.

Pennington, J., Socher, R., and Manning, C. D. GloVe: Global vectors for word representation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2014.

Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., and Zettlemoyer, L. Deep contextualized

word representations. In *Proceedings of NAACL-HLT*, 2018.

Simonyan, K. and Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

Snell, J., Swersky, K., and Zemel, R. Prototypical networks for few-shot learning. In *Advances in Neural Information Processing Systems 30*. 2017.

Sohn, K. Improved deep metric learning with multi-class n-pair loss objective. In *Advances in Neural Information Processing Systems*, pp. 1857–1865, 2016.

Wang, X. and Gupta, A. Unsupervised learning of visual representations using videos. In *Proc. of IEEE International Conference on Computer Vision*, 2015.