Embedding Reliability: On the Predictability of Downstream Performance

Shervin Ardeshir
Netflix
shervina@netflix.com

Navid Azizan
Massachusetts Institute of Technology
azizan@mit.edu

Abstract

In (self-)supervised (pre-)training, such as in contrastive learning, often a network is presented with correspondent (positive) and non-correspondent (negative) pairs of datapoints, and is trained to find an embedding vector for each datapoint, i.e., a representation, which can be further fine-tuned for various downstream tasks. To safely deploy these models in critical decision-making systems, it is crucial to equip them with a measure of their *reliability*. Here we study whether such measures can be quantified for a datapoint in a meaningful way. In other words, we explore if the downstream performance on a given datapoint is predictable, directly from a few characteristics of its pre-trained embedding. We study whether this goal can be achieved by directly estimating the distribution of the training data in the embedding space, and accounting for the local consistency of the representations. Our experiments show that these notions of reliability often strongly correlate with its downstream accuracy. For a more detailed version of this study, please refer to [Ardeshir and Azizan, 2022].

1 Introduction

While deep learning approaches are capable of finding useful representations that have demonstrably enabled breakthroughs in a wide variety of tasks, one cannot wishfully assume that their predictions will always be accurate when queried on various inputs. There have been many examples of these systems making wrong predictions, which in some cases have led to fatal accidents [NTS, 2017, Varshney and Alemzadeh, 2017] and unacceptable errors [Guynn, 2015]. Many such failures may be prevented if the system could supplement its predictions with a level of uncertainty or reliability in those predictions [Dietterich, 2017], which is crucial for building societal trust in such systems.

Despite the recent developments in uncertainty quantification, the majority of the literature[1] has been focused on supervised settings, in which a single input is mapped to an absolute target value. On the other hand, most high-performing models are (pre-)trained using embedding-learning objectives such as contrastive, in which a single input is mapped to a non-absolute abstract embedding vector[2]. Due to this fundamental difference, earlier approaches are not readily applicable to such models.

Figure 1 shows an overview of our setup. A back-box model f is pre-trained on a training dataset, resulting in an embedding vector representing each datapoint. The goal is to study whether there are any notions of reliability for an embedding vector which is indicative of how it would later perform downstream. Given the non-triviality of predicting downstream performance solely from a pre-trained embedding vector, we explore the possibility of such prediction using a few intuitive measures. To

[1] For a more in-depth literature review, please refer to [Ardeshir and Azizan, 2022].

[2] During a typical training regime of a contrastive model, pairs of datapoints are provided as positives or negatives; the contrastive objective then aims to find a data representation in which the positive pairs "attract," (i.e., fall close to each other with an appropriate notion of distance) and the negative pairs "repulse" each other in the embedding space.

ML Safety Workshop, 36th Conference on Neural Information Processing Systems (NeurIPS 2022).

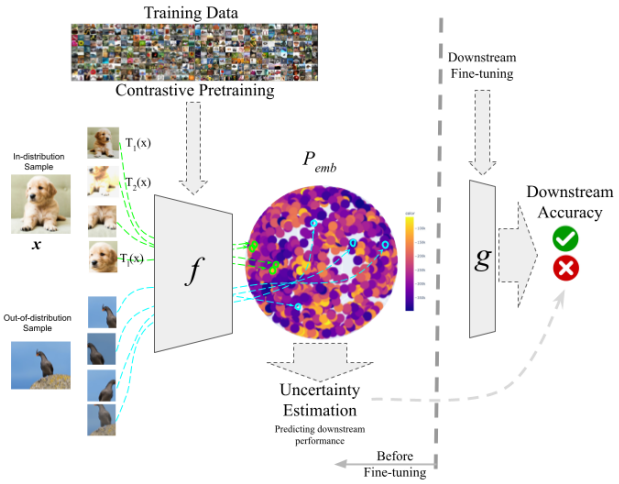

Figure 1: Model f is pre-trained in a supervised or self-supervised manner on a training dataset. Given a test image x, we measure the reliability of its resulting embedding $f(x)$ using notions such as embedding-variance (given different augmentations), and the distribution of the training data embeddings, to quantify this reliability. We show this reliability is not only capable of detecting out of distribution samples, but also correlates with the performance on a datapoint downstream.

that end, given an input and a pre-trained model, we measure the reliability of the resulting embedding in three aspects: (1) *How certain the model is about the location of an embedding vector*. This is computed by introducing variations to the input datapoint and measuring variations in its embedding vector. (2) *How familiar the model is with that area of the embedding space*. In other words, has the model seen training examples with similar embeddings. This notion is computed by directly estimating the distribution of the embedding vectors of the training data. (3) *How well does the model perform in that region of the embedding space*. This is measured by calculating the local retrieval performance of the model. We study whether these intuitive notions meaningfully correlate with the downstream performance on a given input. In the following section, we provide more details on these notions.

2 Framework and Proposed Method

Here we describe our framework and how our various notions of reliability are constructed. Let us consider a model $f : \mathbb{R}^n \to \mathcal{S}^{m-1}$, which maps an n-dimensional input datapoint (e.g., an image) x to the ℓ_2-normalized m-dimensional feature vector $f(x)$ (on the unit hypersphere). Given the model f, we aim to measure the reliability of the embedding vector $f(x)$ for any given input x. As discussed earlier, we do so based on quantifying the uncertainty in the location of the point in the embedding space as well as the consistency of the model's prediction in that region.

Per-Sample Feature Variation (δ). This measure aims to capture *how certain the model is about the location of an embedding vector*. Given a set of transformations (augmentations) of an input datapoint (image) $\{T_1, T_2, \ldots, T_l\}$ used in the training of the contrastive model, we measure the variation across $\{z_1, z_2, \ldots, z_l\}$, where $z_i = f(T_i(x))$ is the embedding vector corresponding to the i-th transformation of the input x. More specifically, we define $\delta(x)$ as the sum of the variances for different dimensions, i.e., the trace of the sample covariance matrix for the observation vectors $\{z_1, z_2, \ldots, z_l\}$. An important characteristic of this metric is that it *does not require access to the training data* and would work on any black-box model. Note that the underlying assumption here is that the downstream task is invariant to the pre-training data transformations (augmentations).

Density (p_{emb}). The density of the embedding space at a point z would intuitively capture *how much data has the model observed* around z during training, which is the transformation of the training data distribution under f. We fit a Gaussian mixture model (GMM) to the m-dimensional embeddings of the training data. Computing this density function requires access to the pre-trained model and an *unsupervised* training dataset, i.e., only the input datapoints and not the labels.

	Model	Training Inputs	Training Labels	Downstream Classifier
Per-Sample Feature Variation: δ	✓	×	×	×
Embedding Density: p_{emb}	✓	✓	×	×
Ensembled Embedding Density: $p_{\text{emb-ens}}$	✓	✓	×	×
Embedding Consistency: $p^{k,\tau}_{\text{emb}}$	✓	✓	✓	×
Ensembled Embedding Consistency: $p^{k,\tau}_{\text{emb-ens}}$	✓	✓	✓	×
Entropy	✓	✓	✓	✓
Max Score	✓	✓	✓	✓

Table 1: Our different reliability measures make different assumptions on access to the model and the training data. A ✓ indicates requiring access. Note that entropy and max score require access to the downstream classifier and are thus not applicable in our setting.

Consistency ($p^{k,\tau}_{\text{emb}}$). The consistency of the model at z measures whether the training datapoints mapped closest to z have consistent labels. This notion would capture *how accurate the model is* at z, based on the fact that a more accurate contrastive model should have a more pure local correspondence. Note that unlike the density-only distribution mentioned above, estimating this distribution requires access to both training data and training labels (correspondences). For each training datapoint, we calculate the fraction of its k nearest neighbors (k-NN) in the embedding space whose class labels are consistent with that datapoint. We then filter out the datapoints based on their k-NN accuracy with a threshold τ, and fit a Gaussian mixture model to the datapoints whose k-NN consistency is above the threshold τ. We denote this distribution by $p^{k,\tau}_{\text{emb}}$. This notion would require access to the model and a *supervised* training dataset, and is thus only applicable to the supervised contrastive learning setup [Khosla et al., 2020]. Note that setting τ to zero yields $p^{k,0}_{\text{emb}}(\cdot) = p_{\text{emb}}(\cdot)$, which would solely capture the density of each datapoint in the training data.

Per-Sample Feature Variance + Embedding Distribution ($p^{k,\tau}_{\text{emb-ens}}$). One could also combine the two notions of per-sample variance and embedding distribution, which has the interpretation of a stochastic embedding [Wang and Isola, 2020]. More specifically, we have an ensemble of probabilities through the l transformations $\{T_1, T_2, ..., T_l\}$, and using the law of total probability we have $p^{k,\tau}_{\text{emb-ens}} = \sum_{i=1}^{l} p^{k,\tau}_{\text{emb}}(f(T_i(x)))p(T_i) = \frac{1}{l}\sum_{i=1}^{l} p^{k,\tau}_{\text{emb}}(f(T_i(x)))$.

The measures mentioned above have different requirements, ranging from access to the black-box model only (feature-variation measure), to requiring access to a fully supervised training dataset (consistency measure). Table 1 summarizes the requirements for each measure. Note that the last two measures (entropy and max score) require the full observation of the downstream task and are solely defined as a baseline. In the next section, we describe our experimental setup and how we evaluate the aforementioned measures.

3 Experimental Results

We pre-train self-supervised (SimCLR) [Chen et al., 2020] and supervised (SupCon) [Khosla et al., 2020] contrastive models with ResNet18 [He et al., 2016] backbones, and on the training set of CIFAR10 or CIFAR100 [Krizhevsky et al., 2009] datasets. We then perform inference on their test sets, alongside test sets of CUBS2011 [Wah et al., 2011] and SVHN [Netzer et al., 2011] as other out-of-distribution datasets. We follow the pre-training and linear fine-tuning protocols in accordance to Khosla et al. [2020].

We evaluate different notions of reliability to cover different aspects of downstream predictability, and out-of-distribution detection. To evaluate our different metrics, we treat them as retrieval instances and compute their AUROC (Area Under the Receiver Operating Characteristic curve). The ground-truth label of the retrieval instance could be derived as a function of the downstream accuracy of a datapoint and whether the model has been exposed to the datapoint's semantic class during pre-training. In what follows, we discuss the details of this evaluation for each reliability notion.

Setup	Dataset	δ	p_{emb}	$p_{\text{emb-ens}}$	$p^{k,\tau}_{\text{emb}}$	$p^{k,\tau}_{\text{emb-ens}}$	Entropy	Max score
SimCLR	CIFAR10	0.652	0.702 ± 0.023	$\mathbf{0.719 \pm 0.004}$	-	-	0.883	0.836
SimCLR	CIFAR100	**0.647**	0.559 ± 0.009	0.564 ± 0.017	-	-	0.816	0.761
SupCon	CIFAR10	0.830	0.805 ± 0.025	0.858 ± 0.004	0.808 ± 0.026	$\mathbf{0.862 \pm 0.002}$	0.916	0.892
SupCon	CIFAR100	**0.766**	0.735 ± 0.017	0.764 ± 0.004	0.720 ± 0.017	0.743 ± 0.002	0.879	0.852

Table 2: In-distribution performance prediction (quantified based on AUROC as described in 3.1) is done on each datapoint in the downstream task of image classification. As it can be observed, all of our notions meaningfully capture sample difficulty as measured by correctness on in-distribution datapoints.

Setup	In-dist	Out-of-dist	δ	p_{emb}	$p_{\text{emb-ens}}$	$p^{k,\tau}_{\text{emb}}$	$p^{k,\tau}_{\text{emb-ens}}$	Entropy	Max score
SimCLR	CIFAR10	CUBS2011	**0.766**	0.59 ± 0.004	0.602 ± 0.074	-	-	0.689	0.745
SimCLR	CIFAR10	SVHN	0.393	0.960 ± 0.000	$\mathbf{0.975 \pm 0.018}$	-	-	0.890	0.918
SimCLR	CIFAR10	CIFAR100	0.645	0.773 ± 0.003	$\mathbf{0.793 \pm 0.035}$	-	-	0.851	0.858
SimCLR	CIFAR100	CUBS2011	**0.783**	0.598 ± 0.0032	0.608 ± 0.020	-	-	0.775	0.783
SimCLR	CIFAR100	SVHN	0.365	0.810 ± 0.002	$\mathbf{0.846 \pm 0.022}$	-	-	0.761	0.789
SimCLR	CIFAR100	CIFAR10	**0.610**	0.515 ± 0.0023	0.516 ± 0.010	-	-	0.692	0.670
SupCon	CIFAR10	CUBS2011	0.580	0.644 ± 0.005	$\mathbf{0.660 \pm 0.031}$	0.640 ± 0.006	0.655 ± 0.022	0.671	0.690
SupCon	CIFAR10	SVHN	0.548	0.977 ± 0.003	$\mathbf{0.995 \pm 0.000}$	0.976 ± 0.003	0.995 ± 0.001	0.962	0.964
SupCon	CIFAR10	CIFAR100	0.765	0.878 ± 0.003	$\mathbf{0.918 \pm 0.002}$	0.877 ± 0.003	0.916 ± 0.002	0.903	0.900
SupCon	CIFAR100	CUBS2011	0.853	0.727 ± 0.006	$\mathbf{0.762 \pm 0.084}$	0.718 ± 0.004	0.747 ± 0.026	0.877	0.884
SupCon	CIFAR100	SVHN	0.546	0.904 ± 0.003	$\mathbf{0.940 \pm 0.015}$	0.867 ± 0.002	0.903 ± 0.017	0.845	0.852
SupCon	CIFAR100	CIFAR10	**0.720**	0.667 ± 0.005	0.689 ± 0.016	0.644 ± 0.002	0.661 ± 0.028	0.739	0.723

Table 3: Out-of-distribution detection (quantified based on AUROC as described in 3.2): As expected, $p_{\text{emb}-\text{ens}}$, which aims to directly estimate the embedding distribution and benefits from the ensemble effect, outperforms other measures in most instances.

3.1 In-distribution Performance Prediction

We evaluate our proposed reliability measures on in-distribution test-set datapoints, and in terms of their capability in retrieving samples that are correctly classified in a downstream classifier. Table 2 shows this metric for our different reliability measures. It can be observed that all of our notions strongly correlate with the correctness of the model's predictions on in-distribution datapoints.

3.2 Out-of-distribution Detection

We evaluate our reliability measures on images from the in-distribution (pre-training dataset) and an out-of-distribution dataset and quantify their performance in terms of retrieving the in-distribution embeddings. In other words, a model pre-trained (supervised or self-supervised) on the training set of dataset A is fed test datapoints from datasets A and B. Then, the effectiveness of the reliability measures are evaluated in terms of distinguishing datapoints of dataset A from those of B. Table 3 contains the performance of our different measures on this task. It can be observed that in most cases, $p_{\text{emb}-\text{ens}}$ has the best performance, whose definition is also more consistent with out-of-distribution detection tasks, as it directly estimates the embedding distribution that comes from the training data. Another observation would be the failure of the feature variation measure in detecting out-of-distribution samples of SVHN in the self-supervised setups. We hypothesize this could be due to the fact that SVHN is a less diverse dataset, which results in its images being mapped close to one another in a CIFAR10 or CIFAR100 pre-trained model. As a result, feature variation would not be a good notion for distinguishing such samples. On the other hand, the probability-based measures result in very high AUROC scores, alluding that these measures capture complementary notions of reliability. For a more detailed version of our experiments, including more insights and ablation studies, please refer to [Ardeshir and Azizan, 2022].

4 Conclusion

In this effort, we explored the possibility of measuring the reliability of abstract embeddings obtained, e.g., from contrastive learning. We show that such measures not only are able to meaningfully detect out-of-distribution samples but also are predictive of performance in downstream tasks. We believe that having such notions of reliability can particularly be insightful, e.g., for deciding between different options of pre-trained models, or for deciding on specific sample weighting policies in downstream fine-tuning tasks.

References

Shervin Ardeshir and Navid Azizan. Uncertainty in contrastive learning: On the predictability of downstream performance. *arXiv preprint arXiv:2207.09336*, 2022.

Collision between a car operating with automated vehicle control systems and a tractor-semitrailer truck near Williston, Florida, May 7, 2016. highway accident report NTSB/HAR-17/02. Technical report, National Transportation Safety Board, 2017.

Kush R Varshney and Homa Alemzadeh. On the safety of machine learning: Cyber-physical systems, decision sciences, and data products. *Big data*, 5(3):246–255, 2017.

Jessica Guynn. Google photos labeled black people'gorillas'. *USA Today*, 1, 2015.

Thomas G Dietterich. Steps toward robust artificial intelligence. *AI Magazine*, 38(3):3–24, 2017.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. *Advances in Neural Information Processing Systems*, 33, 2020.

Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International Conference on Machine Learning*, pages 9929–9939. PMLR, 2020.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

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

Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. Reading digits in natural images with unsupervised feature learning. 2011.