HYPE: Hyperbolic Entailment Filtering for Underspecified Images and Texts

Wonjae Kim Sanghyuk Chun Taekyung Kim Dongyoon Han Sangdoo Yun

NAVER AI Lab

Abstract. In an era where the volume of data drives the effectiveness of self-supervised learning, the specificity and clarity of data semantics play a crucial role in model training. Addressing this, we introduce HYPerbolic Entailment filtering (HYPE), a novel methodology designed to meticulously extract modality-wise meaningful and well-aligned data from extensive, noisy image-text pair datasets. Our approach leverages hyperbolic embeddings and the concept of entailment cones to evaluate and filter out samples with meaningless or underspecified semantics, focusing on enhancing the specificity of each data sample. HYPE not only demonstrates a significant improvement in filtering efficiency but also sets a new state-of-the-art in the DataComp benchmark when combined with existing filtering techniques. This breakthrough showcases the potential of HYPE to refine the data selection process, thereby contributing to the development of more accurate and efficient self-supervised learning models. Additionally, the image specificity ϵ_i can be independently applied to induce an image-only dataset from an image-text or image-only data pool for training image-only self-supervised models and showed superior performance when compared to the dataset induced by CLIP score.

1 Introduction

Recent studies have shown that a machine learning model performance is highly correlated to the training dataset scale and the dataset quality; carefully human-validated high-quality training data leads to a better model performance than the same size of noisy data [30, 37]. However, human-validated dataset construction is labor-intensive, making its scale-up expensive and impractical. As an alternative, there have been attempts to scale up noisy data points until reaching the performance garnered from carefully collected high-quality training datasets [11, 33, 51]. However, this approach requires more than billion-scale data points that introduces another challenges in computational costs and storage size. To mitigate the problem, researchers have begun to study inexpensive automatic data filtering approaches to the noisy billion-scale data points.

The large datasets being created today, except private in-house datasets [54, 62, 74, 79], rely heavily on web-crawled documents by CommonCrawl. As the scale of images and texts obtained from the web is tremendously large, each dataset employs different heuristics for reducing the size of the dataset. These heuristics include, for example, whether the text is a title from Wikipedia,

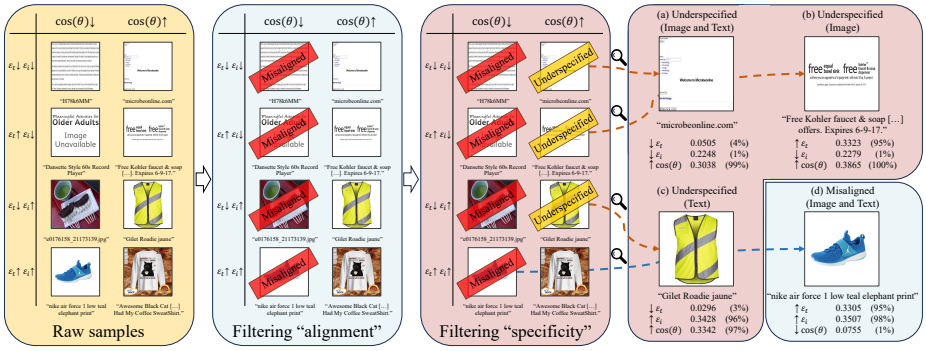

Fig. 1: Example of HYPE filtering on the Datacomp small pool [20]. HYPE leverages both uni-modal specificity (text specificity ϵ_t and image specificity ϵ_i) and cross-modal similarity (CLIP similarity $cos(\theta)$ as in this figure or negative Lorentzian distance $-d_{\mathcal{L}}$ can be used instead) to effectively identify and eliminate misalignment and underspecification issues on noisy image-text pairs. Figures (a-c) show instances where image-text pairs exhibit high alignment yet are flagged for exclusion due to insufficient specificity: (b) demonstrates low image specificity ϵ_i, (c) illustrates low text specificity ϵ_t, and (a) indicates low specificity in both aspects. Conversely, Figure (d) presents a scenario with high ϵ_i and ϵ_t but low $cos(\theta)$, highlighting a misalignment between the image and text, evidenced by the absence of an "elephant print".

whether it is written in English, and whether the image resolution is large enough [20, 53, 57, 58, 73]. Another rule-of-thumb is model-based filtering, usually based on the pre-trained CLIP [53] model, which determines if the given image and text are semantically aligned [20, 57, 58], or if the given image is similar to high-quality images from human-validated datasets, such as ImageNet [20].

Although CLIP-based filtering helps verify the semantic *alignment* between images and texts, we argue that *alignment* alone is insufficient for high-quality data filtering. More specifically, we should consider *specificity* of each data point. Here, we (informally) define *alignment* as whether a given image-text pair is sufficiently similar and *specificity* as whether a given unimodal data point contains sufficient information to be uniquely defined (*i.e.*, specificity measures how each data point has semantically overlapping with other data points). A more formal definition will be described in Sec. 3.3. Figure 1 illustrates the concept of alignment and specificity. In the figure, the website screenshot and the URI are well-aligned, but the information of the screenshot and the URI are not sufficiently enough to be uniquely defined. Unfortunately, as CLIP-based filtering only considers alignment, it cannot filter out underspecified images and texts.

To consider both *alignment* and *specificity*, we employ the pre-trained CLIP [53] and its hyperbolic embedding version, MERU [16]. By employing both alignment and specificity metrics, our data filtering, named HYPerbolic Entailment filtering (HYPE), can successfully handle underspecified samples and misaligned pairs at the same time. More specifically, we propose a novel specificity measurement based on the property of hyperbolic embeddings, the image specificity ϵ_i

and the text specificity ϵ_t. We employ four metrics for HYPE: the cosine similarity ($cos(\theta)$) between the two CLIP embeddings, the negative Lorentzian distance ($-d_{\mathcal{L}}$) [40] between the two MERU embeddings, and the specificity measure of each modality using the entailment cone defined by MERU: ϵ_i (how specific the image is) and ϵ_t (how specific the text is). HYPE utilize all four metrics: ϵ_i, ϵ_t, $-d_{\mathcal{L}}$, and $cos(\theta)$ for filtering, making sure that the samples like shown in Figure 1 are eliminated, which is not possible by alignment-only filtering. By considering various aspects of data points rather than only alignment, HYPE is ranked in the first place on the Datacomp filtering track [20] for small and medium scales by combining with DFN [19]. Our contribution can be summarized as follows.

1. We propose HYPE, a novel method that enhances the training of CLIP models beyond what is possible with traditional CLIP-based filtering techniques by leveraging uni-modal *specificity* along with cross-modal *alignment*.
2. HYPE can be effectively used independently or in conjunction with other filtering methods. When combined, it achieves a new state-of-the-art in the DataComp benchmark, indicating its ability to filter datasets using distinct properties compared to other methods.
3. ϵ_i can be independently applied to induce a dataset for training image-only self-supervised models, showing superior performance compared to alignment-based filtering.

2 Background

2.1 Hyperbolic Embeddings

Despite the usefulness of Euclidean embeddings, they cannot capture additional instance-wise information, such as specificity. In this paper, we employ hyperbolic embeddings to capture additional information for data filtering. A hyperbolic space maps data that needs to be close to many positives at the same time (*i.e.*, more generic data) into closer to the origin, while maps data with fewer positive pairs (*i.e.*, more specific data) into farther away from the origin [40, 49]. Conceptually, the distance from the origin corresponds to the uncertainty represented by Euclidean Gaussian embeddings [63]. Thus, hyperbolic embeddings can naturally capture how the uncertainty of inputs caused by inherent noisy image-text pairs [12]. Practical implementations of $\mathbb{R}^n$ hyperbolic spaces include the Poincaré ball model [1, 2, 17, 18, 21, 22, 35, 49], which distorts the distances in $\mathbb{R}^n$, and the hyperboloid model (Lorentz model), which is defined as a submanifold of $\mathbb{R}^{n+1}$ [16, 38]. A recent study, MERU [16], has successfully extended this concept to image-text contrastive models, showing better performance than CLIP in cross-modal retrieval and illustrating interesting applications of image traversal. In this paper, we focus on noisy pair filtering leveraging the *specificity* we can gather from the hyperbolic model, which was not addressed in MERU, and show the advantages of using hyperbolic CLIP as a filtering network. To be self-contained, we will describe the details of hyperbolic embeddings and how specificity can be measured by hyperbolic embeddings in Section 3.2.

2.2 DataComp Benchmark

For recent years, several evaluation benchmark suites have been proposed to evaluate *models* on various modalities, including text [68, 69], images [78], video [42], and multimodal models [60, 80]. These model-driven benchmarks include evaluation datasets and tasks, but they do not limit models and training datasets. Namely, the three factors of the scaling law [28, 29, 30, 61] –the size of the model, the amount of data, and the number of training steps– cannot be controlled through the benchmarks. It makes fair quantitative comparisons between different algorithms or methods difficult beyond the effect of scale. To address this, DataComp [20] has been proposed as a data-driven, rather than model-driven, benchmark where the size of the model and the number of training steps (the number of samples seen) are controlled and fixed. The Datacomp evaluation consists of 38 tasks, mainly grouped into four task groups: ImageNet, 6 ImageNet distribution shifts [4, 26, 27, 55], 13 VTAB [78], and 3 retrievals [6, 43, 75]. The main evaluation metric of DataComp is computed by the average score of these tasks, and additional benchmarks from CLIP [53] and WILDS [36]. In this paper, we consider the **DataComp filtering track**, a benchmark for evaluating the effectiveness of filtering methods. There are four different scales of datasets in terms of fixed model size, training budget, and the number of seen samples (`small`, `medium`, `large`, and `xlarge`). For example, the number of seen samples of `small` is 12.8M, growing 10 times for each scale (*e.g.*, `large` has 1.28B seen samples). Therefore, for each filtering track, the training method, budget, and the number of seen samples are fixed, but only the seen samples are changed. We note that the training method is fixed as CLIP training and the evaluation protocol is fixed as the average zero-shot score on the 38 tasks of Datacomp evaluation suite. It is because CLIP demonstrates a better scaling trade-off than other methods [37, 73], and there exist well-founded open software [11, 32] and open datasets [7, 48, 57, 58] for the training.

3 Method

This section outlines the overview of HYPE filtering, the theoretical background, and the practical implementation of hyperbolic embeddings, presenting HYPE as an effective method for dataset filtering in image-text contrastive learning.

3.1 Overview of HYPE

While CLIP-based filtering captures *alginment* well, it cannot effectively measure the *specificity* of each data point. More specifically, as CLIP is trained with noisy-contrastive estimation [25, 50] using random samples as negative pairs, CLIP enforces to make each embedding located to a more distinct subspace rather than having semantic overlaps between each other. For example, consider a photo of a dog and a cat together and captions *"A dog"*, *"A cat"*, and *"A dog and a cat"* in Figure 2. In this case, as shown in Figure 2 (a), the best Euclidean space

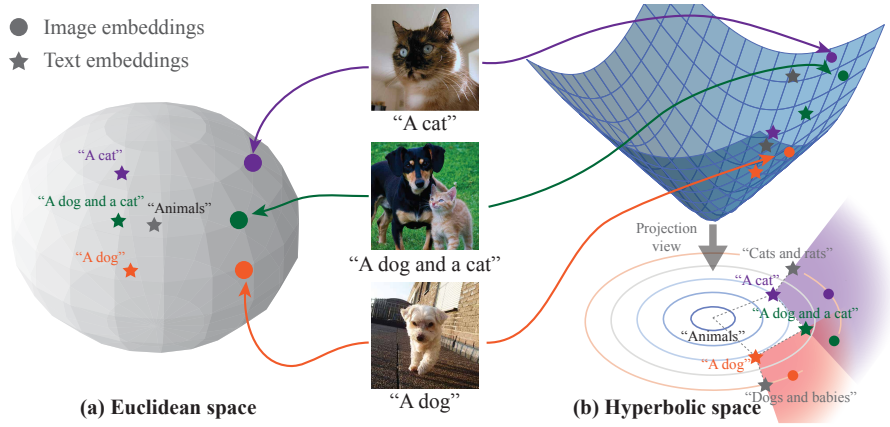

Fig. 2: Conceptual comparisons of Euclidean embeddings and hyperbolic embeddings.

mapping will map the dog and cat photo to the midpoint between the embedding of *'A dog'* and *'A cat'*, because the dog and cat photo should be matched with both dog and cat embeddings. However, the actual semantic meaning is more complex than the average of the two embeddings. As pointed out by Desai et al. [16], it is because CLIP uses the same distance metric at every point.

Hyperbolic embedding, on the other hand, can capture more complex semantics by letting each point have different distance metrics. As shown in Figure 2 (b), hyperbolic embedding space can represent more complex information than Euclidean embedding space. Conceptually, a more generic data point (*i.e.*, potentially matched with more samples) will be mapped into a point close to the center point in hyperbolic space. For example, the textual embeddings of *"A cat"* and *"A dog"* are closer to the center (*"Animals"*) than that of *"A dog and a cat"* and *"Cats and rats"*. Moreover, using the property of hyperbolic embedding space, we can define an "entailment" of each modality, *i.e.*, whether the given data sample can be matched with the other data samples. For example, Figure 2 (b) also illustrates the projected view of the hyperbolic space. In the projected view, we can observe that the "A dong and a cat" caption embedding is placed where the "cones" of caption embeddings "A cat" and "A dog" (shown in purple and red areas, respectively) are overlapped. In other words, by using the concept of the "entailment cone", we can define the entailment of the given input.

Using the entailment cones, we define the "entailment loss" $\mathcal{L}_e(\mathbf{x}, \mathbf{y})$ for the given image-text pair that measures whether the image $\mathbf{y}$ is correctly placed on the entailment cone of the corresponding text $\mathbf{x}$. Then, we define the "specificity" of each input by computing the average entailment loss on the dataset. The image specificity ϵ_i is defined as the average entailment loss, *i.e.*, $\sum_{\mathbf{x}} \frac{\mathcal{L}_e(\mathbf{x},\mathbf{y})}{M}$, and the text specificity ϵ_t is defined similarly, $\sum_{\mathbf{y}} \frac{\mathcal{L}_e(\mathbf{x},\mathbf{y})}{M}$. ϵ_i and ϵ_t measure whether the learned hyperbolic embedding space describes the given input well. We will provide a more rigorous mathematical definition in the latter section. Figure 3 shows examples of images and texts with low and high specificity values (*i.e.*, ϵ_i

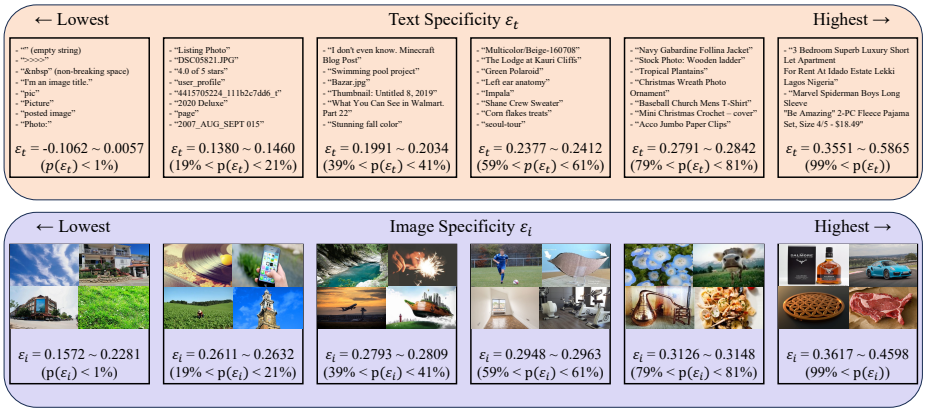

Fig. 3: We show examples of low and high ϵ_i and ϵ_t from the 12.8M Datacomp small pool, where each percentile group spanned with 20% intervals. Here, a higher value denotes that the instance is more specific (see Section 3.3 for details of ϵ_i and ϵ_t). The range absolute value and their percentile $p(\cdot)$ of ϵ_i and ϵ_t are also shown. For texts, the lowest ϵ_t texts are empty sentences or the least specific texts that could fit any image, such as *"Picture"*, while the higher ϵ_t texts are generally longer sentences that describe some object in detail. For images, images with low ϵ_i are either background images with no objects or with too many objects, while images with higher ϵ_i are so-called *iconic* images, which contain a single object that can be described with precision.

and ϵ_t, respectively). As shown in the figure, samples with smaller specificities are more generic and underspecified. For example, the low ϵ_i values of mobile phone or tower images denote their abundant potential relative captions in the dataset. Conversely, Dalmore whisky in the "Highest" category highlights the scarcity of descriptive texts without directly mentioning "Dalmore", underscoring the metric's effectiveness in distinguishing specificity. Similarly, the captions *"pic"* and *"Picture"* have low ϵ_t values as they are vague to describe a specific image.

In this paper, we propose to use not only CLIP alignment score, $cos(\theta)$, but the specificity scores ϵ_i and ϵ_t. Also, as the CLIP embedding space is not sufficient to represent complex image-text representations (as shown in Figure 2), we use the alignment score measured by our hyperbolic CLIP, $-d_{\mathcal{L}}$. Finally, following the baseline DataComp filtering, we additionally employ the ImageNet clustering filter c_{IN}, which denotes whether the given image belongs to ImageNet classes. Our HYPE score is defined as follows:

$$\text{HYPE}_{\text{score}} = \epsilon_i + \epsilon_t - d_{\mathcal{L}} + cos(\theta) + c_{\text{IN}} \tag{1}$$

In the following subsections, we will provide the details of the hyperbolic CLIP [16] and more formal theoretical explanations of the meaning of ϵ_i and ϵ_t.

3.2 Hyperbolic CLIP

In this subsection, we provide a brief introduction to hyperbolic embeddings and its multimodal version, MERU [16]. Hyperbolic embeddings have been actively

studied on diverse modalities, such as images [77] or text [65]. Recently, Desai et al. [16] applied hyperbolic embeddings to image-text joint embedding space based on CLIP, named MERU. MERU is based on the Lorentz model, which uses the upper half of a two-sheeted hyperboloid in the $n+1$-dimensional Euclidean space $\mathbb{R}^{n+1}$ to represent the n-dimensional hyperbolic space $\mathcal{L}^n$. The $\mathbf{x} \in \mathbb{R}^{n+1} = [\mathbf{x}_{space}, x_{time}]$ in this space consists of two components [46]: One is $\mathbf{x}_{space} \in \mathbb{R}^n$ in the n-dimensional *space* dimension and the other is $x_{time} \in \mathbb{R}$ in the one-dimensional *time* axis. This hyperboloid is symmetric with respect to the time axis and has a *Lorentzian inner product* $\langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}} = \langle \mathbf{x}_{space}, \mathbf{y}_{space} \rangle - x_{time}\, y_{time}$, which is different from the Euclidean inner product. From this inner product, the *Lorentzian norm* is $\|\mathbf{x}\|_{\mathcal{L}} = \sqrt{|\langle \mathbf{x}, \mathbf{x} \rangle_{\mathcal{L}}|}$ is derived. Since the Lorentz model is defined to have a constant curvature of $-c$ at all points: $\mathcal{L}^n = \{\mathbf{x} \in \mathbb{R}^{n+1} : \langle \mathbf{x}, \mathbf{x} \rangle_{\mathcal{L}} = {}^{-1}/_{c}\,,\ c > 0\}$, we can derive x_{time} from $\mathbf{x}_{space}$:

$$x_{time} = \sqrt{{}^{1}/_{c} + \|\mathbf{x}_{space}\|^2} \tag{2}$$

MERU is built upon the Lorentz model and the CLIP architecture. MERU does not L^2 normalize $\mathbf{v}_{enc} \in \mathbb{R}^n$, the embedding that passed the last linear projection in CLIP. Instead, MERU uses $\mathbf{v}_{space} = \mathbf{v}_{enc}$ to define $\mathbf{v} = [\mathbf{v}_{enc}, 0] \in \mathbb{R}^{n+1}$ and uses it as a point in the tangent space on the hyperboloid origin $\mathbf{O} = [\mathbf{0}, \sqrt{{}^{1}/_{c}}]$ (this is because $\langle \mathbf{O}, \mathbf{v} \rangle_{\mathcal{L}} = 0$ holds). MERU multiplies $\mathbf{v}$ by a learnable scalar α initialized as $\sqrt{{}^{1}/_{n}}$. The *negative Lorentzian distance*, which we will use as a similarity for the contrastive learning is defined as:

$$-d_{\mathcal{L}}(\mathbf{x}, \mathbf{y}) = -\sqrt{{}^{1}/_{c}} \cdot \cosh^{-1}(-c\, \langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}}) \tag{3}$$

As $-d_{\mathcal{L}}$ can only be calculated on a manifold, not the tangent space, we need to map $\mathbf{v}$ in the tangent space to the manifold. Luckily, as MERU only deals with the tangent space of the origin $\mathbf{O}$, this *exponential map* can be simplified into:

$$\mathbf{x}_{space} = \frac{\sinh(\sqrt{c}\, \|\mathbf{v}_{space}\|)}{\sqrt{c}\, \|\mathbf{v}_{space}\|} \mathbf{v}_{space} \tag{4}$$

By applying the exponential map to text and image embeddings, MERU can find the $-d_{\mathcal{L}}$ between positive and negative pairs in a batch, which can be simply used instead of the cosine similarity of CLIP's InfoNCE loss to train the model. MERU simplifies the exponential map by using the tangent space of the origin, thus minimizing potential numerical instability in the model's computation.

3.3 Entailment Cone and Specificity

Now, we describe how we can measure specificity using hyperbolic embeddings. Note that $-d_{\mathcal{L}}$ also can perform as a filtering metric as a better alignment measure rather than the vanilla CLIP distance $cos(\theta)$. However, $-d_{\mathcal{L}}$ can only measure *alignment* between images and texts but cannot measure how each image or text is *specific*. This paper proposes a new instance-wise filtering metric named *specificity* based on the concept of *entailment*. The concept of *entailment* has its roots in logic and linguistics, long before its incorporation into machine learning [64, 66]. In logic, entailment is a fundamental relationship where the

truth of one statement guarantees the truth of another. In natural language processing (NLP), a number of tasks have been created to verify that the language model can properly capture this entailment relationship (*i.e.*, semantic containment and exclusion): RTE [3, 5, 14, 24], MNLI [70], WNLI [41], etc., and these tasks form a significant part of the GLUE benchmark [68, 69]. Beyond NLP, tasks have also been created in the vision-and-language domain, such as SNLI-VE [71, 72], to evaluate cross-modal entailment relationships between images and text. The concept of an *entailment cone* emerges when we consider how entailment relationships can be represented in a vector space. The idea is that for a given concept represented by a vector, there exists a *cone* in the vector space within which all vectors that are semantically entailed by the original term fall.

While the implementation of entailment cones in the vision-and-language context can be done through order embedding [67], Desai et al. [16] borrows the concepts of Ganea et al. [21] and Le et al. [38] to train MERU using entailment loss, which is involved in the training of the model. In the hyperboloid space drawn by MERU, an entailment cone is defined as a half-aperture with $K = 0.1$:

$$\text{aper}(\mathbf{x}) = \sin^{-1}\left(\frac{2K}{\sqrt{c}\,\|\mathbf{x}_{space}\|}\right) \tag{5}$$

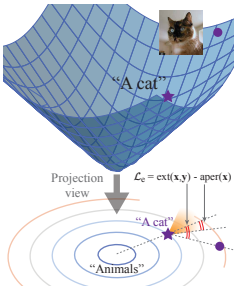

Fig. 4: Visual example of aper Eqn. 5, ext Eqn. 6 and entailment loss Eqn. 7.

Desai et al. [16] empirically demonstrated that *text always entails an image*. This concept can be taken for granted because text, with its symbolic representation, is always less specific than an image with pixel-level specificity. Thus, entailment loss makes the model learn such that the image embedding of a positive image-text pair falls within the cone of its paired text (See Figure 2 (b) as an example). The acute angle that the image embedding $\mathbf{y}$ makes with the text embedding $\mathbf{x}$ can be found following hyperbolic trigonometry:

$$\text{ext}(\mathbf{x},\mathbf{y}) = \cos^{-1}\left(\frac{y_{time} + x_{time}\; c\,\langle \mathbf{x},\mathbf{y}\rangle_{\mathcal{L}}}{\|\mathbf{x}_{space}\|\sqrt{\left(c\,\langle \mathbf{x},\mathbf{y}\rangle_{\mathcal{L}}\right)^2 - 1}}\right) \tag{6}$$

Entailment loss is then determined by the difference between this deviation and the size of the cone:

$$\mathcal{L}_e(\mathbf{x},\mathbf{y}) = \max(0,\ \text{ext}(\mathbf{x},\mathbf{y}) - \text{aper}(\mathbf{x})) \tag{7}$$

The visual explanation of Eqn. 5, 6 and 7 is illustrated in Figure 4. The $\mathcal{L}_e$ alone still requires image-text pairs. To use this value independently to measure uni-modal specificity, we first sorted all samples from the DataComp medium in descending order of CLIP similarity, and then selected the top N samples. We then measured the $\mathcal{L}_e$ for each image and text MERU embedding in the DataComp medium against the MERU embeddings of the opposite modality in the N samples and averaged these values. We used the M images and M texts with the highest average $\mathcal{L}_e$ as our reference set: $\mathcal{S}_i$ and $\mathcal{S}_t$, respectively. We

Table 1: DataComp statistics. We have fewer samples than the original release of DataComp small (12.8M) and medium (128M) due to inaccessible URLs. We confirmed that the overall metric statistics of the samples remain largely unchanged for both scales. Hence, we expect that using these metrics as filtering will achieve almost similar results even when the scale goes beyond DataComp medium. Also, ϵ_t is significantly lower than ϵ_i, namely, *text always entails an image* empirically.

Dataset	Size	ϵ_t	ϵ_i	$-d_{\mathcal{L}}$	$cos(\theta)$	c_{IN}
DataComp Small	11.6M	0.211 ± 0.082	0.289 ± 0.030	-0.726 ± 0.053	0.208 ± 0.064	6.110 ± 4.875
DataComp Medium	115.6M	0.210 ± 0.082	0.289 ± 0.030	-0.726 ± 0.053	0.208 ± 0.064	5.957 ± 4.908

Table 2: ImageNet-1k [56] zero-shot classification accuracy (IN1K) and MS-COCO [43] text-to-image (T2I) and image-to-text (I2T) retrieval recalls on Karpathy test split [34] and mAP on ECCV Caption [13] performances of CLIP and MERU models. Note that the results reported in Desai et al. [16] used COCO 2017 validation split instead of Karpathy test split. The results marked with an asterisk ($*$) are the official checkpoints from [16], and the unmarked ones are the ones we reproduced. The best scores are in **bold** and the second best scores are in underlined.

Model	Method	Dataset Size	# Samples Seen	IN1K	COCO T2I				COCO I2T			
					R1	R5	R10	mAP	R1	R5	R10	mAP
B/16	CLIP $*$	12M	245M	37.9	15.4	34.3	44.4	18.5	21.2	43.4	54.1	10.3
	MERU $*$	12M	245M	37.5	15.1	33.8	44.8	18.6	21.2	43.0	53.9	10.0
	MERU	27M	128M	42.3	24.6	49.0	60.8	28.8	37.9	63.4	75.0	18.3
L/16	CLIP $*$	12M	245M	38.4	14.2	32.1	42.6	17.6	21.2	41.9	52.2	9.8
	MERU $*$	12M	245M	38.8	14.7	33.2	43.4	18.5	21.2	42.1	52.7	10.2
L/14	MERU	12M	128M	38.2	13.6	31.2	41.0	17.6	21.2	44.2	54.6	10.3
	MERU	27M	256M	**50.2**	**30.2**	**55.4**	**66.9**	**32.8**	**43.3**	**69.5**	**79.7**	**21.0**

set N and M to 20,000 as the value of ϵ_* converged when calculated over 3,000 samples. The relatively low variance of metrics shown in Table 1 shows that the specificity values remain consistent across different reference sets, suggesting that it is invariant to the choice of dataset and not subject to bias. Now, given any image, we can calculate its $\mathcal{L}_e$ with the M text reference set, and we define this value as image specificity ϵ_i. Similarly, we can calculate the $\mathcal{L}_e$ value for text, and define this value as text specificity ϵ_t:

$$\epsilon_t(\mathbf{x}) = \sum_{\mathbf{y} \in \mathcal{S}_i} \mathcal{L}_e(\mathbf{x}, \mathbf{y})/M \text{ and } \epsilon_i(\mathbf{y}) = \sum_{\mathbf{x} \in \mathcal{S}_t} \mathcal{L}_e(\mathbf{x}, \mathbf{y})/M \tag{8}$$

3.4 Hyperbolic Entailment Filtering (HYPE)

Here, we describe the details of HYPE. We first train a MERU model with ViT-B/16 and ViT-L/14 backbones on CC3M [59] and CC12M [8] in addition to RedCaps [15]. Both models were trained on 8 V100s with a batch size of 2048. The models were optimized using AdamW [44], with a weight decay of 0.2, $(\beta_1, \beta_2) = (0.9, 0.98)$, and a learning rate of 5×10^{-4}. After a warm-up of 4,000 steps, ViT-B/16 was trained for 62,500 steps and ViT-L/14 for 125,000 steps using a cosine decay learn rate schedule. Our implementation is based on the OpenCLIP codebase [32]. Training of ViT-B/16 and ViT-L/14 MERU models took approximately 10 hours and 61 hours, respectively.

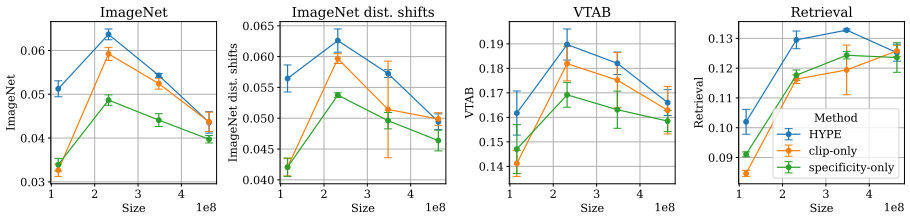

Fig. 5: Comparisons with baseline filtering methods and HYPE. We show the subsampled Datacomp training set from 10% to 40% and evaluate them across four Datacomp benchmark task groups. Each model was trained four times with varied seeds. 10% and 30% results are the same as Table 4.

Note that the original MERU by Desai et al. [16] was trained solely on the RedCaps [15] dataset. We added more clean data points to allow better filtering capability, as the findings of DFN [19] and our discussion in Sec. 4.1. We also note that the original MERU uses ViT-B/16 and ViT-L/16 backbones with their textual encoder having a hidden size of 512. Since DataComp [20] uses ViT-B/16 and ViT-L/14 for its baseline CLIP filtering method, we retrained MERU on ViT-B/16, which has a 512 textual encoder hidden size, and ViT-L/14, which has a 768 textual encoder hidden size, with the expanded dataset. The results of MERU re-training are shown in Table 2. Surprisingly, even when all the training hyperparameters, including the batch size, were the same as in the original MERU, and the training was done with fewer steps (ViT-B/16), the zero-shot performance of the reproduced MERU model was significantly better than that of the original MERU. All results in this paper are based on the hyperbolic embeddings obtained by our reproduced ViT-L/14 MERU.

We extract ϵ_i, ϵ_t, and $-d_{\mathcal{L}}$ for every sample in the target image-text dataset using our MERU model. For each sample, we also compute and store the ImageNet clustering-based image filter used by DataComp and the CLIP score $cos(\theta)$ of the ViT-L/14 CLIP. The clustering-based filter c_{IN} is quantified as a value of 10 if included and 0 if not, enabling preferential use. Table 1 summarizes the statistics for the datasets tested in this paper. The $\text{HYPE}_{\text{score}}$ is obtained by linearly combining all the metrics with equal weight as defined in Eqn. 1.

Note that the metrics used for HYPE have the same computation complexity as the CLIP distance. On the other hand, a number of the existing filtering methods need more complex computations, such as the OCR engine (T-MARS [45]) and additional clustering operations (CIDS [76]). Also, we argue that our method is data-efficient compared to the previous model-driven filtering methods (our method: 27M, CLIP: 400M, DFN [19]: 2B) Our method is simple yet archives the first place in small and medium DataComp leaderboards.

3.5 Ablation study

In this subsection, we provide ablation studies of HYPE design choices. First, we show that using our metric outperforms solely using CLIP similarity or solely

using specificity in Figure 5. Across sample sizes from 10% to 40% and across four Datacomp benchmark task groups, HYPE consistently outperformed each metric alone. Note that the gaps can be small in 40% samples because they share more samples, thus less filtering effect. In 10% or 20%, where filtering works more sensitively, HYPE always outperforms the baseline methods with large gaps. This demonstrates that, as suggested in Figure 1, each metric, when used in isolation, is limited in its ability to filter out data that adversely affects image-text contrastive learning.

We also examined the effect of each component of HYPE in Table 3. Our findings confirm that while c_{IN} enhances IN zero-shot, omitting c_{IN} yielded superior average performance (1st vs. 2nd rows). The results of removing $cos(\theta)$ (3rd row) are inspiring: our model is trained with 1/15 data points and 1/5 seen training samples than OpenAI CLIP but performs better than the CLIP baseline (4th row). We also found that there is no single weight combination for Eqn. 1 that performs best for all datasets. In this paper, we set all weights as 1 (*i.e.*, 1st row), considering the importance of the ImageNet benchmark and the relatively low importance of small datasets, such as SVHN in the VTAB benchmark.

Table 3: Ablation study

Method	IN	VTAB	Ret	Avg
HYPE	**0.338**	0.357	**0.286**	0.343
HYPE − c_{IN}	0.322	**0.369**	0.273	**0.349**
HYPE − c_{IN} − $cos(\theta)$	0.320	0.358	0.278	0.345
$cos(\theta)$ only [20]	0.260	0.326	0.235	0.322
$cos(\theta)$ + c_{IN} [20]	0.297	0.346	0.231	0.328

4 Experiments

In this section, we will show and discuss the results of using HYPE for the image-text contrastive learning benchmark DataComp's small and medium, and ϵ_i for image-only contrastive learning by itself. Before that, we will discuss the methods we used as a baseline for filtering in image-text contrastive learning.

4.1 Comparison Methods

In this paper, "filtering" refers to the process of excluding samples from the training data, while "sampling" refers to how often each sample is used for training. Here, we introduce the major baselines of the DataComp filtering benchmark. The most simple baseline filters the dataset by language (*e.g.*, leaving only English text), text length (*e.g.*, more than two words or five characters), and image size (*e.g.*, aspect ratio of 3 or less and shorter axis more than 200 pixels). There are two methods that empirically perform well. One is image-based clustering, which groups the CLIP embeddings 100K centroids and then filters the samples in centroids based on whether one of the images in ImageNet is closest to the centroid of the cluster to which each sample belongs. The other is CLIP score filtering we explained before. Recently, three notable approaches have been proposed for Datacomp medium scale: DFN [19], CIDS [76], and T-MARS [45].

Data Filtering Networks (DFN) [19] is a model-centric approach without multi-step filtering; they directly train a network determines whether filtering

Table 4: We have compared HYPE with concurrent works challenging the Datacomp benchmark. Methods with an asterisk (∗) are our reproductions given their sample IDs for a fair comparison, as we were able to download fewer samples than the original models. HYPE on the Datacomp small scale reports values from the average of four models trained with different seeds. The uniform column stands for whether or not each method uses the given sample IDs with equal probability during training. The best scores are in **bold**, and the second best scores are in underlined.

Method	Datacomp Scale	Sample Size	Uniform	ImageNet	ImageNet Dist. Shift	VTAB	Retrieval	Average
CLIP L/14 30% [20]	Small	3.8M	Yes	0.051	0.055	0.190	0.119	0.173
WS [31]	Small	4.1M	Yes	0.056	0.061	0.196	0.132	0.180
HYPE 10%	Small	1.2M	Yes	0.051	0.056	0.162	0.102	0.150
HYPE 20%	Small	2.3M	Yes	0.064	0.063	0.190	0.130	0.176
HYPE 30%	Small	3.5M	Yes	0.054	0.057	0.182	0.133	0.170
CLIP L/14 30% [20]	Medium	38.0M	Yes	0.273	0.230	0.338	0.251	0.328
WS [31]	Medium	24.8M	Yes	0.305	0.253	0.363	0.270	0.342
T-MARS [45]	Medium	23.0M	No	0.338	0.274	0.371	0.231	0.357
CIDS [76] ∗	Medium	21.3M	No	0.326	0.262	0.372	0.258	0.365
DFN [19] ∗	Medium	17.1M	Yes	0.376	0.300	0.384	0.284	0.372
HYPE 10%	Medium	11.6M	Yes	0.327	0.257	0.365	0.246	0.340
HYPE 20%	Medium	23.1M	Yes	0.338	0.269	0.357	0.286	0.343
HYPE 30%	Medium	34.7M	Yes	0.300	0.243	0.337	0.276	0.332
HYPE 10% + CIDS [76] ∗	Medium	18.9M	No	0.346	0.276	0.390	0.264	0.373
HYPE 10% + DFN [19] ∗	Medium	21.5M	No	**0.382**	**0.303**	**0.393**	**0.306**	**0.379**

out the given data. The authors showed that CLIP cosine similarity-based DFN performs the best among the other possible variants, such as, autoencoder [23]. The DFN paper also observes that training DFN with high-quality training samples (*i.e.*, a proprietary dataset, such as HQITP-357M [19, 54]) is crucial for better filtering, compared to low-quality and large-scale training samples. Note that the best-performing DFN is trained on HQITP-357M, whose scale is already beyond the DataComp medium of 128M, making it very resource-intensive.

Cluster-Importance-based Data Selection (CIDS) [76] uses the 38 Datacomp evaluation datasets to filter out samples dissimilar to the target evaluation datasets and then duplicates the samples with similar distributions for more extensive training sampling. While this method does not require a significant amount of additional computing resources, it has a notable drawback: the model needs to know on which dataset the CLIP will be evaluated before filtering.

Text-Masking and Re-Scoring (T-MARS) [45] reveals that many samples in noisy datasets, like DataComp's dataset pool, are simple OCR samples (image-text pairs that simply transcribe the text in the image). This helps CLIP focus on learning visual semantics by retaining only those images in the data pool that still have high CLIP scores after masking the text in the images. However, removing all OCR-like samples would harm the performance of tasks like MNIST [39], SVHN [47], and RenderedSST2 [53] in DataComp's evaluation dataset; therefore, they still require CLIP to read the text in the images.

4.2 Image-Text Contrastive Pre-training

Table 4 includes the DataComp filtering track results of the main competitors (*i.e.*, DFN [19], CIDS [76] and T-MARS [45]) and the ensemble filtering with

weak supervision [31]. As mentioned in Table 1, we only use the subset of the official DataComp due to the dissipated URLs (about 10% samples were lost). For a fair comparison, we obtained the sample IDs used by the two best-performing methods on DataComp medium: CIDS [76] and DFN [19], and reproduced the model with only those belonging to our pool – denoted with asterisk ($*$).

In the table, we observe that HYPE performs extremely well in retrieval scores, *e.g.*, HYPE 20% Medium shows 0.286 retrieval, which outperforms all the baselines. We believe that it is because hyperbolic embeddings significantly improve retrieval performances compared to the CLIP embedding (as observed in Table 2), making the filtered data samples by HYPE more suitable for retrieval tasks. This is especially noteworthy given that DFN used 357M high-quality proprietary image-text pairs while HYPE is achievable with a much smaller 27M publicly accessible dataset. Note that DataComp only contains 3 retrieval task groups out of 38 tasks; therefore, if we add more retrieval tasks for the evaluation benchmark, HYPE will achieve a higher average score than others.

Second, HYPE can be combined with the other methods. As our specificity metric is single-modality filtering and orthogonal to cross-modality filtering, such as CLIP filtering, all other baselines rely on, it can properly filter underspecified examples as shown in Figure 1. This characteristic helps us to mark the first rank in the DataComp small and medium track by combining HYPE with DFN.

Additionally, we trained two more B/16 models with different training seen samples, 128M and 256M, whose IN-ZS accuracies are 42.3% and 47.6%. With these models, we report their correlation to DataComp medium's IN-ZS accuracy as described in [19]. As shown in Figure 6, a better-performing MERU model consistently induces a more effective filtering network, evidenced by improved zero-shot accuracy. This upward trend suggests that further improvements in MERU could lead to even more effective filtering, which is not observed in the downward trend of Euclidean CLIP.

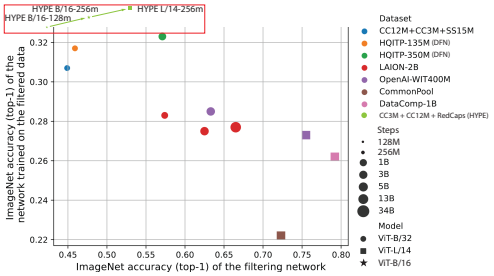

Fig. 6: Filtering network IN-ZS acc. vs. Induced networks IN-ZS acc. (overlaid on the figure of the DFN paper [19])

4.3 Image-Only Contrastive Pre-training

As our specificity metric is an uni-modal metric, unlike the CLIP similarity, we can apply our filtering method to uni-modal datasets. Specifically, we investigate the image specificity metric (ϵ_i)-based filtering for image-only self-supervised learning (SSL) methods, as previous works highlight the significance of iconic images in SSL training [52]. Since ϵ_i can efficiently identify iconic images (as shown in Figure 3), we can expect that ϵ_i-based filtering will lead to better SSL performances. We filter out underspecified images from the DataComp medium dataset and measure the SSL performances using two methods, SimCLR [9], and

Table 5: ImageNet-1K linear probing classification accuracies of ViT-S. The table compares $cos(\theta)$ and ϵ_i on inducing various size image-only datasets (DataComp Medium) for image-only self-supervised learning methods: SimCLR [9] and MoCo-v3 [10].

Model	Filtering Metric	Dataset Size			
		0.13M	0.32M	0.64M	1.28M
SimCLR [9]	$cos(\theta)$	47.06	45.47	52.31	49.68
	ϵ_i	**53.30**	**50.89**	**57.49**	**54.55**
MoCo-v3 [10]	$cos(\theta)$	39.00	45.00	51.10	53.40
	ϵ_i	**44.70**	**51.60**	**56.80**	**59.70**

MoCo-v3 [10]. We also provide CLIP similarity $cos(\theta)$-based filtering, recognized for inducing well-aligned images from noisy datasets, based on image-caption pairs in the DataComp medium dataset.

Table 5 shows the results from 1.28M images (comparable to ImageNet [56]) to 0.13M (10% of ImageNet). For the comparison, we use the established hyperparameters searched on ImageNet. Following the practice of the DataComp filtering track, we keep the number of seen samples fixed for every dataset size, *i.e.*, we use more epochs for smaller dataset sizes. We report the linear probing performances on the ImageNet validation set following the standard SSL evaluation protocol. Table 5 reveals that ϵ_i consistently outperforms $cos(\theta)$ across all dataset sizes and models. Note that MoCo-v3 trained with a dataset induced by ϵ_i outperforms SimCLR trained with a dataset induced by $cos(\theta)$ for the most of dataset sizes. This result shows that the lower-performing SSL method can outperform the higher-performing ones by simply replacing the data.

5 Discussion and Future Work

We conclude this paper by discussing the limitations of our method and outlining future research directions. A notable limitation is that our experiments did not include the larger DataComp subsets, specifically the large and xlarge scales. Considering that HYPE shows an increasing performance gap as the dataset size grows—from small to medium—it is reasonable to hypothesize that HYPE might demonstrate exceptional performance when applied to these larger datasets.

Furthermore, HYPE was designed with a hyperbolic CLIP size set to L/14, aligning with Datacomp's standards. However, there is a strong basis to believe that employing a larger hyperbolic CLIP architecture could significantly enhance performance metrics. Additionally, our research solely utilized ϵ_i to create an image-only dataset. We posit that employing ϵ_t to generate a text dataset could result in a visually meaningful text corpus. This new corpus could be instrumental in training a language model capable of rapidly adapting to visual inputs. Finally, we recognize the potential for extensive ablation studies, especially regarding the coefficient used in merging metrics for HYPE, such in-depth analysis could yield further insights into the filtering model's behavior and performance, thereby enhancing its overall efficacy.

References

[1] Mina Ghadimi Atigh, Julian Schoep, Erman Acar, Nanne Van Noord, and Pascal Mettes. Hyperbolic image segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4453–4462, 2022. 3

[2] Yushi Bai, Zhitao Ying, Hongyu Ren, and Jure Leskovec. Modeling heterogeneous hierarchies with relation-specific hyperbolic cones. *Advances in Neural Information Processing Systems*, 34:12316–12327, 2021. 3

[3] Roy Bar Haim, Ido Dagan, Bill Dolan, Lisa Ferro, Danilo Giampiccolo, Bernardo Magnini, and Idan Szpektor. The second PASCAL recognising textual entailment challenge. 2006. 8

[4] Andrei Barbu, David Mayo, Julian Alverio, William Luo, Christopher Wang, Dan Gutfreund, Josh Tenenbaum, and Boris Katz. Objectnet: A large-scale bias-controlled dataset for pushing the limits of object recognition models. *Advances in neural information processing systems*, 32, 2019. 4

[5] Luisa Bentivogli, Ido Dagan, Hoa Trang Dang, Danilo Giampiccolo, and Bernardo Magnini. The fifth PASCAL recognizing textual entailment challenge. 2009. 8

[6] Yonatan Bitton, Nitzan Bitton Guetta, Ron Yosef, Yuval Elovici, Mohit Bansal, Gabriel Stanovsky, and Roy Schwartz. Winogavil: Gamified association benchmark to challenge vision-and-language models. *Advances in Neural Information Processing Systems*, 35:26549–26564, 2022. 4

[7] Minwoo Byeon, Beomhee Park, Haecheon Kim, Sungjun Lee, Woonhyuk Baek, and Saehoon Kim. Coyo-700m: Image-text pair dataset. https://github.com/kakaobrain/coyo-dataset, 2022. 4

[8] Soravit Changpinyo, Piyush Sharma, Nan Ding, and Radu Soricut. Conceptual 12M: Pushing web-scale image-text pre-training to recognize long-tail visual concepts. In *CVPR*, 2021. 9

[9] 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. 13, 14

[10] Xinlei Chen*, Saining Xie*, and Kaiming He. An empirical study of training self-supervised vision transformers. *arXiv preprint arXiv:2104.02057*, 2021. 14

[11] Mehdi Cherti, Romain Beaumont, Ross Wightman, Mitchell Wortsman, Gabriel Ilharco, Cade Gordon, Christoph Schuhmann, Ludwig Schmidt, and Jenia Jitsev. Reproducible scaling laws for contrastive language-image learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2818–2829, 2023. 1, 4

[12] Sanghyuk Chun. Improved probabilistic image-text representations. In *International Conference on Learning Representations*, 2024. 3

[13] Sanghyuk Chun, Wonjae Kim, Song Park, Minsuk Chang Chang, and Seong Joon Oh. ECCV Caption: Correcting false negatives by collecting machine-and-human-verified image-caption associations for MS-COCO. In *European Conference on Computer Vision (ECCV)*, 2022. 9

[14] Ido Dagan, Oren Glickman, and Bernardo Magnini. The PASCAL recognising textual entailment challenge. In *Machine learning challenges. evaluating predictive uncertainty, visual object classification, and recognising tectual entailment*, pages 177–190. Springer, 2006. 8

[15] Karan Desai, Gaurav Kaul, Zubin Aysola, and Justin Johnson. RedCaps: Web-curated image-text data created by the people, for the people. In *NeurIPS Datasets and Benchmarks*, 2021. 9, 10

[16] Karan Desai, Maximilian Nickel, Tanmay Rajpurohit, Justin Johnson, and Ramakrishna Vedantam. Hyperbolic Image-Text Representations. In *Proceedings of the International Conference on Machine Learning*, 2023. 2, 3, 5, 6, 7, 8, 9, 10

[17] Ankit Dhall, Anastasia Makarova, Octavian Ganea, Dario Pavllo, Michael Greeff, and Andreas Krause. Hierarchical image classification using entailment cone embeddings. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 836–837, 2020. 3

[18] Aleksandr Ermolov, Leyla Mirvakhabova, Valentin Khrulkov, Nicu Sebe, and Ivan Oseledets. Hyperbolic vision transformers: Combining improvements in metric learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7409–7419, 2022. 3

[19] Alex Fang, Albin Madappally Jose, Amit Jain, Ludwig Schmidt, Alexander Toshev, and Vaishaal Shankar. Data filtering networks. *arXiv preprint arXiv:2309.17425*, 2023. 3, 10, 11, 12, 13

[20] Samir Yitzhak Gadre, Gabriel Ilharco, Alex Fang, Jonathan Hayase, Georgios Smyrnis, Thao Nguyen, Ryan Marten, Mitchell Wortsman, Dhruba Ghosh, Jieyu Zhang, et al. Datacomp: In search of the next generation of multimodal datasets. *arXiv preprint arXiv:2304.14108*, 2023. 2, 3, 4, 10, 11, 12

[21] Octavian Ganea, Gary Bécigneul, and Thomas Hofmann. Hyperbolic entailment cones for learning hierarchical embeddings. In *International Conference on Machine Learning*, pages 1646–1655. PMLR, 2018. 3, 8

[22] Songwei Ge, Shlok Mishra, Simon Kornblith, Chun-Liang Li, and David Jacobs. Hyperbolic contrastive learning for visual representations beyond objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6840–6849, 2023. 3

[23] Xinyang Geng, Hao Liu, Lisa Lee, Dale Schuurmans, Sergey Levine, and Pieter Abbeel. Multimodal masked autoencoders learn transferable representations. *arXiv preprint arXiv:2205.14204*, 2022. 12

[24] Danilo Giampiccolo, Bernardo Magnini, Ido Dagan, and Bill Dolan. The third PASCAL recognizing textual entailment challenge. In *Proceedings of the ACL-PASCAL workshop on textual entailment and paraphrasing*, pages 1–9. Association for Computational Linguistics, 2007. 8

[25] Michael Gutmann and Aapo Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 297–304. JMLR Workshop and Conference Proceedings, 2010. 4

[26] Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, et al. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8340–8349, 2021. 4

[27] Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15262–15271, 2021. 4

[28] Danny Hernandez, Jared Kaplan, Tom Henighan, and Sam McCandlish. Scaling laws for transfer. *arXiv preprint arXiv:2102.01293*, 2021. 4

[29] Joel Hestness, Sharan Narang, Newsha Ardalani, Gregory Diamos, Heewoo Jun, Hassan Kianinejad, Md Mostofa Ali Patwary, Yang Yang, and Yanqi Zhou. Deep learning scaling is predictable, empirically. *arXiv preprint arXiv:1712.00409*, 2017. 4

[30] Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. An empirical analysis of compute-optimal large language model training. *Advances in Neural Information Processing Systems*, 35: 30016–30030, 2022. 1, 4

[31] Tzu-Heng Huang, Changho Shin, Sui Jiet Tay, Dyah Adila, and Frederic Sala. Multimodal data curation via object detection and filter ensembles. 2023. 12, 13

[32] Gabriel Ilharco, Mitchell Wortsman, Ross Wightman, Cade Gordon, Nicholas Carlini, Rohan Taori, Achal Dave, Vaishaal Shankar, Hongseok Namkoong, John Miller, Hannaneh Hajishirzi, Ali Farhadi, and Ludwig Schmidt. Openclip, 2021. If you use this software, please cite it as below. 4, 9

[33] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *International conference on machine learning*, pages 4904–4916. PMLR, 2021. 1

[34] Andrej Karpathy and Li Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3128–3137, 2015. 9

[35] Valentin Khrulkov, Leyla Mirvakhabova, Evgeniya Ustinova, Ivan Oseledets, and Victor Lempitsky. Hyperbolic image embeddings. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6418–6428, 2020. 3

[36] Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, et al. Wilds: A benchmark of in-the-wild distribution shifts. In *International Conference on Machine Learning*, pages 5637–5664. PMLR, 2021. 4

[37] Skanda Koppula, Yazhe Li, Evan Shelhamer, Andrew Jaegle, Nikhil Parthasarathy, Relja Arandjelovic, João Carreira, and Olivier Hénaff. Where should i spend my flops? efficiency evaluations of visual pre-training methods. *arXiv preprint arXiv:2209.15589*, 2022. 1, 4

[38] Matt Le, Stephen Roller, Laetitia Papaxanthos, Douwe Kiela, and Maximilian Nickel. Inferring concept hierarchies from text corpora via hyperbolic embeddings. *arXiv preprint arXiv:1902.00913*, 2019. 3, 8

[39] Yann LeCun. The mnist database of handwritten digits. *http://yann. lecun. com/exdb/mnist/*, 1998. 12

[40] John M Lee. *Introduction to Riemannian manifolds*. Springer, 2018. 3

[41] Hector J Levesque, Ernest Davis, and Leora Morgenstern. The Winograd schema challenge. In *AAAI Spring Symposium: Logical Formalizations of Commonsense Reasoning*, page 47, 2011. 8

[42] Linjie Li, Jie Lei, Zhe Gan, Licheng Yu, Yen-Chun Chen, Rohit Pillai, Yu Cheng, Luowei Zhou, Xin Eric Wang, William Yang Wang, et al. Value: A multi-task benchmark for video-and-language understanding evaluation. *arXiv preprint arXiv:2106.04632*, 2021. 4

[43] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014. 4, 9

[44] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2017. 9

[45] Pratyush Maini, Sachin Goyal, Zachary C Lipton, J Zico Kolter, and Aditi Raghunathan. T-mars: Improving visual representations by circumventing text feature learning. *arXiv preprint arXiv:2307.03132*, 2023. 10, 11, 12

[46] Hermann Minkowski. *Raum und zeit*. Springer, 1988. 7

[47] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. Reading digits in natural images with unsupervised feature learning. 2011. 12

[48] Thao Nguyen, Gabriel Ilharco, Mitchell Wortsman, Sewoong Oh, and Ludwig Schmidt. Quality not quantity: On the interaction between dataset design and robustness of clip. *Advances in Neural Information Processing Systems*, 35:21455–21469, 2022. 4

[49] Maximillian Nickel and Douwe Kiela. Poincaré embeddings for learning hierarchical representations. *Advances in neural information processing systems*, 30, 2017. 3

[50] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. 4

[51] Hieu Pham, Zihang Dai, Golnaz Ghiasi, Kenji Kawaguchi, Hanxiao Liu, Adams Wei Yu, Jiahui Yu, Yi-Ting Chen, Minh-Thang Luong, Yonghui Wu, et al. Combined scaling for zero-shot transfer learning. *Neurocomputing*, 555:126658, 2023. 1

[52] Senthil Purushwalkam and Abhinav Gupta. Demystifying contrastive self-supervised learning: Invariances, augmentations and dataset biases. *Advances in Neural Information Processing Systems*, 33:3407–3418, 2020. 13

[53] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *Int. Conf. Machine Learning*, pages 8748–8763. PMLR, 2021. 2, 4, 12

[54] Kanchana Ranasinghe, Brandon McKinzie, Sachin Ravi, Yinfei Yang, Alexander Toshev, and Jonathon Shlens. Perceptual grouping in contrastive vision-language models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5571–5584, 2023. 1, 12

[55] Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do ImageNet classifiers generalize to ImageNet? In *Proceedings of the 36th International Conference on Machine Learning*, pages 5389–5400. PMLR, 2019. 4

[56] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *IJCV*, 115:211–252, 2015. 9, 14

[57] Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarczyk, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, and Aran Komatsuzaki. Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. *arXiv preprint arXiv:2111.02114*, 2021. 2, 4

[58] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in Neural Information Processing Systems*, 35: 25278–25294, 2022. 2, 4

[59] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of ACL*, 2018. 9

[60] Suwon Shon, Ankita Pasad, Felix Wu, Pablo Brusco, Yoav Artzi, Karen Livescu, and Kyu J Han. Slue: New benchmark tasks for spoken language understanding evaluation on natural speech. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7927–7931. IEEE, 2022. 4

[61] Ben Sorscher, Robert Geirhos, Shashank Shekhar, Surya Ganguli, and Ari Morcos. Beyond neural scaling laws: beating power law scaling via data pruning. *Advances in Neural Information Processing Systems*, 35:19523–19536, 2022. 4

[62] Chen Sun, Abhinav Shrivastava, Saurabh Singh, and Abhinav Gupta. Revisiting unreasonable effectiveness of data in deep learning era. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017. 1

[63] Alexandru Tifrea, Gary Bécigneul, and Octavian-Eugen Ganea. Poincar\'e glove: Hyperbolic word embeddings. 2019. 3

[64] Víctor Manuel Sánchez Valencia. *Studies on natural logic and categorial grammar*. Universiteit van Amsterdam, 1991. 7

[65] Marco Valentino, Danilo S. Carvalho, and André Freitas. Multi-relational hyperbolic word embeddings from natural language definitions, 2023. 7

[66] Johan Van Benthem et al. *Essays in logical semantics*. Springer, 1986. 7

[67] Ivan Vendrov, Ryan Kiros, Sanja Fidler, and Raquel Urtasun. Order-embeddings of images and language. *arXiv preprint arXiv:1511.06361*, 2015. 8

[68] Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium, 2018. Association for Computational Linguistics. 4, 8

[69] Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. Superglue: A stickier benchmark for general-purpose language understanding systems. *Advances in neural information processing systems*, 32, 2019. 4, 8

[70] Adina Williams, Nikita Nangia, and Samuel Bowman. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122. Association for Computational Linguistics, 2018. 8

[71] Ning Xie, Farley Lai, Derek Doran, and Asim Kadav. Visual entailment task for visually-grounded language learning. *arXiv preprint arXiv:1811.10582*, 2018. 8

[72] Ning Xie, Farley Lai, Derek Doran, and Asim Kadav. Visual entailment: A novel task for fine-grained image understanding. *arXiv preprint arXiv:1901.06706*, 2019. 8

[73] Hu Xu, Saining Xie, Xiaoqing Ellen Tan, Po-Yao Huang, Russell Howes, Vasu Sharma, Shang-Wen Li, Gargi Ghosh, Luke Zettlemoyer, and Christoph Feichtenhofer. Demystifying clip data. *arXiv preprint arXiv:2309.16671*, 2023. 2, 4

[74] I. Zeki Yalniz, Hervé Jégou, Kan Chen, Manohar Paluri, and Dhruv Mahajan. Billion-scale semi-supervised learning for image classification, 2019. 1

[75] Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2:67–78, 2014. 4

[76] Haichao Yu, Yu Tian, Sateesh Kumar, Linjie Yang, and Heng Wang. The devil is in the details: A deep dive into the rabbit hole of data filtering. *arXiv preprint*

arXiv:2309.15954, 2023. 10, 11, 12, 13

[77] Yun Yue, Fangzhou Lin, Kazunori D Yamada, and Ziming Zhang. Hyperbolic contrastive learning, 2023. 7

[78] Xiaohua Zhai, Joan Puigcerver, Alexander Kolesnikov, Pierre Ruyssen, Carlos Riquelme, Mario Lucic, Josip Djolonga, Andre Susano Pinto, Maxim Neumann, Alexey Dosovitskiy, et al. A large-scale study of representation learning with the visual task adaptation benchmark. *arXiv preprint arXiv:1910.04867*, 2019. 4

[79] Xiaohua Zhai, Alexander Kolesnikov, Neil Houlsby, and Lucas Beyer. Scaling vision transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12104–12113, 2022. 1

[80] Wangchunshu Zhou, Yan Zeng, Shizhe Diao, and Xinsong Zhang. Vlue: A multi-task multi-dimension benchmark for evaluating vision-language pre-training. In *International Conference on Machine Learning*, pages 27395–27411. PMLR, 2022. 4