A. Supplementary Material

In this supplementary material, §A.1 contains implementation details and §A.2 contains further results as well as ablations. In §A.3, we discuss paradigm differences of CiT from existing approaches.

A.1. Implementation Details

A.1.1 PyTorch Pseudo Code

To facilitate implementation of CiT, we provide the PyTorch pseudo-code in Algorithm 4 below.

Algorithm 4: CiT: PyTorch Pseudo Code

```
# b: maximum training steps as budget.
# d: iterator of raw data.
# t_meta: textual metadata.
# bsz: batch_size.
# t: threshold.
# gamma: target radio for curation.
# s: number of expected pairs.

c = 0
while c < b:
  x_meta = model(t_meta)
  x_meta = normalize(x_meta)
  d_c = []
  while len(d_c) < s:
    x_imgs, x_txts = next(d)
    x_txts = model(x_txts)
    x_txts = normalize(x_txts)
    v = x_txts @ x_meta.t()
    sel = max(v) > t
    b_ratio = sum(sel) / len(sel)
    if b_ratio < gamma:
      sel = max(v).topk(
                k=int(bsz*gamma), dim=0)
      d_c.extend((x_imgs[sel], x_txts[sel]))

  for (x_imgs, x_txts) in batchify(d_c):
    x_imgs, x_txts = model(x_imgs, x_txts)
    x_imgs, x_txts = normalize(x_imgs,x_txts)
    # scale: learnable log logit scale
    l = exp(scale) * x_imgs @ x_txts.t()
    labels = arange(bsz)
    loss = cross_entropy(l, labels)
    loss.backward()
    c += 1
```

A.1.2 Dataloader Implementation

For efficiency, we only load text during the curation loop and the training loop uses the curated indices to reload the full image-text pairs. Our implementation also supports in-memory storage of curated image-text pairs in case the data source is not randomly accessible for (re-)loading curated data, where all s pairs of training data can be stored in the CPU memory with image tensors represented as `uint8` data. We use a larger batch size for curation (compared to training) to speed up CiT.

Hyperparameter	Value
Optimizer	AdamW
Optimizer momentum	$\beta_1 = 0.9, \beta_2 = 0.999$
Optimizer ϵ	1e-8
Weight Decay (proj.)	1.0
Weight Decay (other)	0.2
Base Learning Rate	5e-4
Learning Rate Schedule	cosine decay
Minimum Learning Rate	1e-5
Gradient Clipping	None
Warm-up % of Train Steps	4%
Batch size	16,384
GPUs	16 Nvidia V100 32GB GPUs
precision	float16
Max BERT len.	32
Train Aug.	RandomResizedCrop(224, scale=(0.5, 1.0))
YFCC15M/YFCC100M Aug.	shuffle/join tags[38]
Eval Aug.	Resize(256), CenterCrop(224)
AugReg rgb Mean	(0.5, 0.5, 0.5)
AugReg rgb Std.	(0.5, 0.5, 0.5)
Other encoder rgb Mean	(0.485, 0.456, 0.406)
Other encoder rgb Std.	(0.229, 0.224, 0.225)

Table 9: Hyperparameters of CiT Training.

Data Source	Metadata	b	t	γ
YFCC15M	IN-1K	5K	0.55	0.003
YFCC15M	IN-21K	8K	0.55	0.003
YFCC15M	multi.	8K	0.55	0.003
YFCC100M	IN-1K	5K	0.7	0.01
LAION400M	IN-1K	5K	0.6	0.01
LAION400M	IN-21K	30K	0.65	0.01
LAION400M	multi.	16K	0.6	0.01
RAW IMG-TXT	IN-1K	8K	0.7	0.003
RAW IMG-TXT	IN-21K	60K	0.75	0.003
RAW IMG-TXT	multi.	30K	0.7	0.003

Table 10: Hyperparameters of CiT Curation.

A.1.3 Detailed Implementation Settings

The hyper-parameters of CiT training are shown in Table 9. We mostly follow [38, 20, 21]. CiT is trained on 16 GPUs with a global batch size of 16,384 (1024 per GPU).

Hyperparameters for CiT curation outlined in §3 of the main paper are shown in Table 10. We use different thresholds t and minimal ratios γ for each dataset/metadata combination to fit the training into a budget b shown in the table as well. We use the same values for all variants of vision encoders. Due to smaller size, we use a lower t for YFCC15M and CC12M, whereas for YFCC100M and Raw Img-Text Crawl we use a higher t to focus on high-quality data from the raw data source, in order to roughly meet the budget b.

Single GPU Setting. We provide more details on the implementation of the extremely efficient single GPU setup used for zero-shot evaluation on multiple tasks in Table 12. We can fit a batch size of 1,536 into a *single* 32GB V100

GPU and train for $b = 5000$ steps. To ensure the training can be finished quickly, we set $\gamma = 0.05$. Further to reduce the chance of using the minimal ratio during curation, we perform a pre-curation on YFCC15M for each task using BERT-SimCSE with a threshold of 0.45 to remove pairs with low relevance.

A.1.4 Implementation Differences from LiT

While we aim for a close reproduction of LiT [38], there are a few tricks that our implementation does not incorporate and we suspect the differences on our LiT reproduction on YFCC stem from those. Below we list some tricks known to us, but there could be more differences we are not aware of since we have no access to LiT's full preprocessing and training code.

Preprocessing. For the captions, LiT performs extra filtering and removes titles that start with "DSC", "IMG", "Picture". Also, LiT removes text consisting of only the word "image" or text that contains a large fraction of digits.

Joint Contrastive Loss. LiT adopts a joint contrastive loss over 3 text fields in YFCC15M and shows the gain in Figure 8 of the LiT paper [38]. Since this technique is specific to the type of captions in the specific YFCC data, we remove it from our implementation and randomly sample one of the three text fields to pair with a training image.

Text encoder. LiT adopts various text encoders such as $\text{BERT}_{\text{base}}$ and $\text{BERT}_{\text{large}}$. This work consistently uses $\text{BERT}_{\text{base}}$ for all main results to have a fair comparison.

A.1.5 Additional Ablations

This section extends ablations in Table 1 of the main paper to *(i)* evaluation prompts and *(ii)* training objectives.

Evaluation Prompts. We first verify the effects of LiT's extra prompts on CiT in Table 11a. We obtain a +0.2% gain by adding them to the CLIP prompts.

Training Objective. We ablate the $\mathcal{L}_{\text{img2txt}}$ training objective which our approach uses (see §3.2 of the main paper). In Table 11a we see that this variant provides a +0.2% gain over CLIP's objective that also incorporates a text2img loss.

Eval. Prompts	Acc
CLIP+LiT prompts	**61.4**
CLIP prompts only	61.2

(a) Evaluation Prompts

Objective	Acc
img2txt obj.	**61.4**
CLIP obj.	61.2

(b) Training Objective

Table 11: **Additional ablation experiments**. We use the default setup (MoCo-v3 / $\text{BERT}_{\text{base}}$-SimCSE) and YFCC15M as data source and report IN-1K Accuracy.

A.2. Additional Results

This section extends the results of CiT in the main paper to full results across 26 CLIP/SLIP benchmarks on YFCC15M and LAION400M and an extra ablation study.

A.2.1 Full Results on YFCC15M

We show the full results of Table 7 in main paper above in Table 12 below. On average, CiT-multi-meta (52.6) is slightly better than CiT-21K-meta (51.7), which is better than CiT-sep-meta and CiT-1K-meta (47.2). It appears that the broader ImageNet-21K wordnet taxonomy works well across datasets, and combining metadata from all downstream tasks is only slightly better than that. We note that training on the larger metadata does not introduce much extra curation compute since forwarding the raw examples takes the majority of computation. Nevertheless, we observe that larger metadata takes longer to converge and therefore increase the training budget to $b = 8000$ for CiT-21K-meta and CiT-multi-meta. We expect larger budgets will lead to even better results.

Besides what was already discussed in the main paper, we observe that CiT performs even better on larger models or models trained with supervised (AugReg IN-21K) or weakly supervised (SWAG) data than the unsupervisedly pre-trained MoCo-v3 on IN-1K. Out-of-domain issues (*e.g.* MNIST) are present even for larger vision encoders.

A.2.2 Full Results on LAION400M

In Table 13, we show the result of CiT trained on LAION400M and evaluated on 26 CLIP/SLIP benchmarks. With a larger data source, we realize CiT takes more time to converge especially with more metadata, which can be attributed to more data meeting the curation criteria. We set $b = 16000$ for CiT-multi-meta and $b = 30000$ for CiT-21K-meta. The trend is similar to YFCC15M but with better performance aross the benchmarks. Similar as in Table 12, CiT-multi-meta is better than CiT-21K-meta, but this time the gap is larger. In addition to the longer training, we believe that the combined metadata from 26 benchmarks are more effective on larger pre-training data.

A.2.3 Full Results on Raw Image-Text Crawl

In Table 14, we show the result of CiT trained on our raw image-text crawl and evaluated on 26 benchmarks. With a larger raw data source, we realize CiT takes more time to converge. We set b = 30000 for CiT-multi-meta and b = 60000 for CiT-21K-meta. The trend is similar to LAION-400M but raw Image-Text Crawl is not cleaned for vision-language association. Similar as in Table 13, CiT-multi-meta is better than CiT-21K-meta, but the gap is larger. We

Vis. Encoder	Init.	Hrs	Food-101	CIFAR10	CIFAR100	CUB	SUN397	Cars	Aircraft	DTD	Pets	Caltech-101	Flowers	MNIST	FER-2013	STL-10	EuroSAT	RESISC45	GTSRB	KITTI	Country211	PCAM	UCF101	Kinetics700	CLEVR	HatefulMemes	SST2	ImageNet	Avg
CLIP [21, 20]																													
ViT-B/16	scratch	27	50.6	66.0	34.5	38.8	51.1	4.0	5.4	21.2	28.5	60.9	53.3	8.4	17.3	90.5	30.2	21.5	6.1	35.1	10.5	53.5	28.5	22.1	10.8	52.4	50.7	37.6	34.2
ViT-L/16	scratch	189	59.5	72.9	41.5	40.3	53.6	6.9	6.4	20.6	27.9	65.4	55.0	10.3	34.5	94.2	22.7	28.8	5.8	41.4	12.6	54.9	34.3	24.0	12.9	54.3	50.1	40.4	37.4
SLIP[20]																													
ViT-B/16	scratch	41	59.5	78.6	45.2	38.7	53.4	5.4	5.7	26.1	31.1	71.0	56.6	9.8	19.6	94.4	20.3	28.9	14.5	34.0	11.6	55.4	37.7	26.9	17.5	52.8	51.1	42.8	38.0
ViT-L/16	scratch	284	64.4	87.8	56.4	39.8	58.9	8.6	7.8	26.8	32.0	76.6	59.4	13.2	36.0	96.6	27.7	36.5	7.2	28.8	15.6	54.4	42.6	30.0	14.1	53.4	50.1	46.2	41.2
CiT-1K-meta																													
ViT-B/16	MoCo-v3	5	45.6	81.0	49.9	30.4	44.9	6.3	8.3	26.8	80.0	71.2	25.1	7.3	26.0	95.2	19.1	14.3	6.9	22.2	6.2	54.1	34.7	24.7	**13.4**	50.7	50.1	61.2	38.5
ViT-B/16	AugReg	8	57.9	92.3	74.2	**36.9**	52.5	7.7	5.6	25.2	77.9	84.5	38.8	8.3	31.2	94.4	16.6	24.3	6.5	17.2	6.4	**59.1**	47.8	32.2	13.3	52.0	50.1	68.9	41.6
ViT-L/16	AugReg	8	60.0	**93.6**	**77.8**	36.3	54.0	9.0	5.7	25.6	79.8	**87.3**	45.2	**9.7**	29.2	96.1	20.9	32.8	7.0	**36.0**	7.6	52.8	51.5	35.2	12.6	**53.0**	49.7	71.6	43.8
ViT-H/14	SWAG	11	**79.0**	91.6	68.1	35.3	**56.9**	**26.2**	**12.5**	**30.0**	**88.8**	86.4	**47.6**	8.1	**31.3**	**97.8**	**27.6**	**46.4**	**7.3**	34.2	**14.5**	50.3	**54.7**	**43.8**	12.3	51.8	**51.0**	**73.3**	**47.2**
CiT-21K-meta																													
ViT-B/16	MoCo-v3	15	51.2	84.4	53.5	45.7	52.3	7.6	9.0	31.6	69.2	73.8	56.1	10.6	24.5	95.7	30.1	23.4	7.9	**28.5**	9.2	51.0	39.5	28.7	15.0	49.3	49.1	57.4	40.6
ViT-B/16	AugReg	23	75.3	93.8	75.7	57.8	59.8	9.7	10.1	35.4	68.3	87.9	74.3	12.1	27.4	97.1	30.8	30.6	7.3	24.3	9.9	50.5	54.7	37.4	13.6	**53.8**	**50.1**	63.7	46.6
ViT-L/16	AugReg	29	78.9	**95.1**	**78.6**	**60.5**	61.9	11.6	10.9	35.1	74.2	90.5	75.4	**14.8**	**34.8**	98.0	24.7	35.5	7.5	25.7	10.9	**50.8**	57.4	40.7	**14.8**	49.9	48.7	67.7	48.3
ViT-H/14	SWAG	39	**92.2**	92.9	70.9	59.0	**64.7**	**36.9**	**14.9**	**40.3**	**87.7**	**90.9**	**77.4**	10.1	32.7	**99.1**	**38.8**	**53.2**	**9.3**	15.9	**20.5**	50.7	**62.2**	**49.4**	12.9	46.8	44.2	**71.4**	**51.7**
CiT-multi-meta																													
ViT-B/16	MoCo-v3	11	51.3	81.8	50.5	50.7	51.6	9.5	14.6	30.8	75.6	73.3	58.7	10.3	26.2	95.6	23.2	19.1	7.8	14.6	9.4	50.8	39.7	28.0	14.7	52.8	50.0	58.8	40.4
ViT-B/16	AugReg	11	77.8	94.0	76.5	63.9	60.1	10.3	13.1	35.2	79.0	88.9	79.4	12.2	33.0	96.2	**31.6**	29.3	10.2	**17.4**	9.6	50.8	56.0	38.0	12.5	**55.8**	47.8	67.0	47.9
ViT-L/16	AugReg	16	80.4	**95.3**	**79.4**	65.6	61.9	13.3	11.3	35.1	79.9	90.6	80.1	10.7	**37.8**	97.4	29.3	35.0	7.8	13.8	10.7	49.7	59.5	41.3	13.0	54.5	**47.9**	70.5	48.9
ViT-H/14	SWAG	31	**91.8**	90.7	71.3	**65.6**	**62.4**	**47.9**	**19.7**	**40.8**	**91.7**	**91.3**	**81.2**	**10.7**	37.5	**98.0**	23.9	**46.4**	**11.0**	12.4	**20.2**	**51.3**	**64.3**	**50.2**	**13.5**	54.6	47.1	**73.4**	**52.6**
CiT-sep.-meta (single GPU)																													
ViT-B/16	MoCo-v3	4	59.1	82.2	55.2	56.6	50.7	13.0	13.1	32.8	74.8	77.6	65.9	16.9	13.8	96.3	17.1	21.6	7.6	**40.6**	9.4	53.5	42.7	27.8	14.2	52.2	50.9	50.7	42.2
ViT-B/16	AugReg	5	79.1	94.4	75.2	73.8	**60.6**	19.4	**17.4**	36.6	78.1	88.0	79.8	12.4	**39.2**	**97.0**	**31.1**	29.1	11.1	30.1	9.9	51.9	54.9	37.1	**19.2**	**52.5**	50.0	56.8	49.4
ViT-L/16	AugReg	7	83.8	**94.8**	**79.6**	**76.9**	60.4	19.6	17.2	36.0	77.8	89.6	82.2	12.1	39.0	96.7	24.8	**31.2**	9.7	26.9	10.7	57.6	59.1	39.9	14.9	46.8	51.2	60.1	**49.9**
ViT-H/14	SWAG	11	**92.1**	89.9	71.8	71.3	**65.4**	52.0	**20.9**	**38.7**	**90.6**	**90.4**	**84.8**	**15.1**	30.6	92.8	26.8	**47.1**	**13.4**	34.8	**20.8**	**59.4**	**65.8**	**50.1**	14.0	48.5	**51.7**	**67.0**	**54.1**

Table 12: CiT trained on YFCC15M and evaluated on 26 CLIP/SLIP benchmarks: we vary metadata on IN-1K, IN-21K and combined class names on 26 tasks (CiT-multi-meta) with a single training and run 26 separate training on each task with a single GPU (CiT-sep.-meta).

Vis. Encoder	Init.	Hrs	Food-101	CIFAR10	CIFAR100	CUB	SUN397	Cars	Aircraft	DTD	Pets	Caltech-101	Flowers	MNIST	FER-2013	STL-10	EuroSAT	RESISC45	GTSRB	KITTI	Country211	PCAM	UCF101	Kinetics700	CLEVR	HatefulMemes	SST2	ImageNet	Avg
CLIP (WIT400M) [21]																													
ViT-B/32	scratch	458	84.4	91.3	65.1	37.8	63.2	59.4	21.2	44.5	87.0	87.9	66.7	51.9	47.3	97.2	49.4	60.3	32.2	39.4	17.8	58.4	64.5	47.8	24.8	57.6	59.6	63.2	56.9
ViT-B/16	scratch	981	89.2	91.6	68.7	39.1	65.2	65.6	27.1	46.0	88.9	89.3	70.4	56.0	52.7	98.2	54.1	65.5	43.3	44.0	23.3	48.1	69.8	52.4	23.4	61.7	59.8	68.6	60.1
ViT-L/14	scratch	6803	92.9	96.2	77.9	48.3	67.7	77.3	36.1	55.3	93.5	92.6	78.7	87.2	57.5	99.3	59.9	71.6	50.3	23.1	32.7	58.8	76.2	60.3	24.3	63.3	64.0	75.3	66.2
OpenCLIP *																													
ViT-B-32	scratch	458	n/a	90.8	70.2	n/a	67.0	79.2	16.8	54.3	86.8	83.3	68.3	37.4	42.7	95.5	51.6	n/a	42.0	28.8	14.7	54.6	n/a	n/a	16.3	n/a	52.6	62.9	n/a
ViT-B-16	scratch	981	n/a	91.7	71.0	n/a	69.6	83.7	17.5	51.3	89.2	83.5	69.3	66.6	42.9	97.0	50.3	n/a	43.5	19.0	18.1	60.5	n/a	n/a	28.8	n/a	54.7	67.0	n/a
ViT-L-14	scratch	6803	n/a	94.7	77.4	n/a	72.6	89.6	25.1	60.3	91.9	84.2	75.4	76.4	50.1	98.0	61.8	n/a	50.0	20.8	23.1	48.6	n/a	n/a	24.2	n/a	56.3	72.7	n/a
CiT-1K-meta																													
ViT-B/16	MoCo-v3	26	31.2	80.7	56.7	29.5	41.7	12.6	3.9	35.2	85.9	82.3	19.1	16.3	25.0	89.7	20.0	19.7	**14.5**	42.2	3.7	55.3	34.8	23.0	14.4	49.5	49.3	67.0	38.6
ViT-B/32	AugReg	62	45.0	86.6	68.8	34.5	48.1	12.1	3.8	35.3	87.0	87.6	34.5	10.2	29.2	89.8	19.7	23.0	10.5	33.1	4.4	50.6	45.5	27.7	15.2	48.5	**50.4**	67.5	41.1
ViT-B/16	AugReg	63	45.4	87.8	70.9	33.7	50.8	12.4	3.3	38.0	86.2	89.0	31.5	9.7	26.4	90.0	25.3	25.3	13.2	**34.9**	5.2	**54.7**	50.0	31.5	14.7	50.4	49.3	73.0	42.4
ViT-L/16	AugReg	27	45.3	**90.6**	**76.3**	36.3	54.7	13.6	5.0	35.9	87.2	**92.1**	32.0	10.2	20.0	91.3	28.2	31.2	10.6	21.4	5.5	51.7	50.9	33.6	**16.1**	48.9	50.1	75.7	42.9
ViT-H/14	SWAG	26	**65.4**	89.8	68.7	**36.4**	**56.5**	**38.0**	**7.9**	**41.7**	**89.4**	88.5	**41.4**	**10.2**	**30.5**	**94.3**	**34.6**	**41.5**	12.0	19.1	**12.3**	49.5	**57.0**	**42.6**	13.2	**51.5**	46.5	**76.2**	**46.7**
CiT-21K-meta																													
ViT-B/16	MoCo-v3	70	64.8	85.0	63.1	59.5	56.3	26.2	8.1	40.2	87.6	87.1	60.6	17.8	34.5	95.9	29.4	30.3	10.9	33.0	6.4	**54.5**	48.8	31.2	15.1	47.9	50.1	64.1	46.5
ViT-B/32	AugReg	57	71.7	91.1	72.8	62.4	59.0	18.8	5.9	42.6	81.8	89.8	67.5	16.3	38.8	96.3	27.1	32.8	12.4	33.9	6.4	52.8	56.8	35.9	**16.4**	51.0	50.1	65.0	48.3
ViT-B/16	AugReg	72	77.1	92.8	74.7	**68.9**	61.9	20.6	8.3	41.5	85.7	91.2	73.8	**21.7**	38.3	97.0	26.2	36.4	**15.1**	**41.8**	7.1	52.4	56.8	38.3	12.1	51.0	**50.5**	71.2	50.5
ViT-L/16	AugReg	97	77.5	**93.5**	**79.1**	67.6	62.9	19.5	8.3	44.8	84.4	93.1	71.5	18.9	34.2	**98.0**	**29.6**	38.9	11.7	22.9	7.7	50.9	60.3	41.6	14.8	51.5	48.2	73.9	50.2
ViT-H/14	SWAG	135	**89.2**	91.5	72.1	68.2	**64.0**	**36.9**	**10.4**	**43.9**	88.2	**92.1**	**75.8**	7.1	**41.7**	97.4	29.2	**49.6**	10.7	34.6	**15.0**	50.9	**62.6**	**46.4**	13.2	**52.3**	49.7	**76.1**	**52.6**
CiT-multi-meta																													
ViT-B/16	MoCo-v3	31	68.1	84.3	62.0	63.7	56.9	65.7	16.0	40.3	90.0	87.8	61.1	6.8	26.6	92.1	27.6	35.9	**18.0**	38.6	7.2	50.9	56.0	35.2	17.2	46.0	49.7	65.8	48.8
ViT-B/32	AugReg	32	75.2	90.0	72.2	70.9	60.2	43.9	11.8	42.8	86.6	90.2	74.6	29.2	21.6	93.0	31.7	33.3	13.5	**44.7**	6.9	51.1	61.7	38.7	14.9	49.9	50.1	66.2	51.0
ViT-B/16	AugReg	51	80.2	91.5	74.4	**75.1**	62.3	53.7	15.5	40.1	87.2	90.8	76.3	12.3	31.2	92.4	28.1	38.3	13.2	18.6	7.8	60.5	66.0	42.5	14.0	50.3	50.0	71.7	51.7
ViT-L/16	AugReg	61	81.6	**92.7**	**79.2**	72.3	63.8	56.9	15.7	42.6	88.5	92.9	73.9	**22.6**	33.3	94.1	30.9	38.4	16.9	27.7	8.7	**56.7**	68.4	45.5	**16.4**	50.0	48.3	74.8	53.6
ViT-H/14	SWAG	54	**92.1**	91.0	71.8	71.7	**66.3**	**77.4**	**18.7**	**51.3**	**93.8**	**92.2**	**81.5**	14.9	**39.6**	**97.5**	**39.4**	**50.0**	15.0	19.1	**17.8**	50.9	**71.8**	**52.4**	14.7	**51.7**	**51.1**	**76.5**	**56.5**

Table 13: CiT trained on LAION400M and evaluated on 26 CLIP benchmarks: We vary metadata from IN-1K (CiT-1K-meta), IN-21K (CiT-21K-meta) and combined class names from 26 benchmarks (CiT-multi.-meta). We also list results from CLIP on WIT400M and OpenCLIP trained on LAION400M. *: from `https://github.com/LAION-AI/CLIP_benchmark`, with some results using VTAB benchmark evaluation/prompts.

Vis. Encoder	Init.	Hrs	Food-101	CIFAR10	CIFAR100	CUB	SUN397	Cars	Aircraft	DTD	Pets	Caltech-101	Flowers	MNIST	FER-2013	STL-10	EuroSAT	RESISC45	GTSRB	KITTI	Country211	PCAM	UCF101	Kinetics700	CLEVR	HatefulMemes	SST2	ImageNet	Avg
CLIP (WIT400M) [21]																													
ViT-B/32	scratch	458	84.4	91.3	65.1	37.8	63.2	59.4	21.2	44.5	87.0	87.9	66.7	51.9	47.3	97.2	49.4	60.3	32.2	39.4	17.8	58.4	64.5	47.8	24.8	57.6	59.6	63.2	56.9
ViT-B/16	scratch	981	89.2	91.6	68.7	39.1	65.2	65.6	27.1	46.0	88.9	89.3	70.4	56.0	52.7	98.2	54.1	65.5	43.3	44.0	23.3	48.1	69.8	52.4	23.4	61.7	59.8	68.6	60.1
ViT-L/14	scratch	6803	92.9	96.2	77.9	48.3	67.7	77.3	36.1	55.3	93.5	92.6	78.7	87.2	57.5	99.3	59.9	71.6	50.3	23.1	32.7	58.8	76.2	60.3	24.3	63.3	64.0	75.3	66.2
CiT-1K-meta																													
ViT-B/16	MoCo-v3	39	29.0	86.0	56.5	17.6	41.3	12.4	5.8	25.7	83.8	77.0	10.6	**10.8**	24.9	95.1	22.3	20.8	6.8	35.6	4.2	50.8	27.7	20.5	**17.2**	48.9	50.1	68.4	36.5
ViT-B/32	AugReg	69	42.8	92.2	70.5	**22.1**	49.0	11.4	5.5	27.0	83.8	81.1	16.5	8.2	**32.5**	94.3	**29.4**	22.2	8.5	**39.1**	4.9	51.3	37.6	26.7	16.4	48.0	50.1	67.8	40.0
ViT-B/16	AugReg	72	43.9	92.1	73.4	20.4	**50.0**	10.9	4.5	31.3	84.6	83.0	18.8	7.1	21.5	96.2	23.3	22.4	**11.2**	29.4	5.2	**52.3**	41.9	29.4	17.0	**50.6**	50.1	74.9	40.2
ViT-L/16	AugReg	105	47.8	**95.4**	**76.0**	18.5	49.4	11.4	5.6	30.9	**84.7**	**83.7**	22.4	6.4	25.6	96.8	24.7	29.7	8.9	36.3	5.3	50.9	45.9	31.0	16.3	46.5	50.1	**77.5**	41.4
ViT-H/14	SWAG	43	**57.2**	93.2	68.5	19.8	47.2	**25.6**	**5.9**	**32.4**	81.3	82.5	**25.3**	8.2	28.8	**97.4**	17.6	**42.2**	8.1	29.2	**10.3**	50.9	**53.7**	**38.8**	14.5	48.0	**53.2**	77.1	**43.0**
CiT-21K-meta																													
ViT-B/16	MoCo-v3	134	57.1	87.1	60.3	57.1	54.0	10.5	**6.0**	37.0	84.6	82.8	59.9	9.8	26.8	96.8	31.8	30.8	8.3	41.2	7.4	59.9	37.9	25.9	**20.8**	48.2	50.1	62.8	44.4
ViT-B/32	AugReg	148	64.4	93.2	71.7	49.5	56.8	10.8	5.7	35.4	76.2	85.8	60.9	9.5	29.1	95.4	27.1	25.2	9.3	39.8	7.7	51.3	45.8	32.1	14.1	**51.3**	50.1	62.2	44.6
ViT-B/16	AugReg	161	70.0	93.6	75.9	58.2	59.9	11.7	5.2	**37.7**	74.9	89.3	61.7	9.8	32.6	97.9	**29.5**	29.4	11.2	40.9	9.0	51.1	49.6	36.1	13.6	48.9	50.1	69.4	46.8
ViT-L/16	AugReg	228	71.7	**96.0**	**78.7**	56.7	**62.4**	12.2	5.9	37.4	77.0	**90.6**	65.3	**14.6**	**37.6**	98.3	27.8	34.0	8.5	**34.0**	9.4	44.2	54.7	39.0	15.5	47.9	50.1	72.6	47.8
ViT-H/14	SWAG	310	**80.4**	93.2	72.0	**58.4**	60.8	**25.6**	5.5	36.0	**78.4**	89.1	**70.6**	7.8	34.7	**98.9**	28.4	**41.7**	**10.8**	29.9	**14.0**	**50.8**	**57.5**	**41.9**	12.4	45.8	**52.6**	**75.5**	**49.0**
CiT-multi-meta																													
ViT-B/16	MoCo-v3	91	70.4	88.8	61.1	60.1	59.0	63.2	24.5	38.4	90.2	85.5	66.5	9.8	32.0	96.6	**35.4**	39.0	9.5	**35.8**	10.2	50.3	48.7	33.4	**17.1**	43.8	50.1	66.1	49.4
ViT-B/32	AugReg	62	72.7	92.9	71.0	51.0	58.9	30.9	10.9	36.3	86.6	87.4	67.5	9.8	36.3	94.5	29.1	29.4	8.5	33.4	8.6	54.9	51.6	36.3	14.8	**49.2**	50.0	64.4	47.6
ViT-B/16	AugReg	62	81.3	94.0	76.6	65.2	62.2	44.1	17.9	**41.3**	90.0	90.6	74.9	9.8	35.3	97.5	34.6	36.5	13.1	34.4	10.4	**56.8**	57.6	41.3	13.4	50.6	50.1	71.9	52.0
ViT-L/16	AugReg	62	82.4	**96.1**	**79.2**	62.4	64.1	44.5	15.8	41.2	89.3	91.3	74.9	9.8	34.7	98.2	27.9	38.7	8.9	33.4	11.1	55.9	61.3	44.0	11.9	48.9	50.1	74.4	51.9
ViT-H/14	SWAG	203	**93.7**	93.5	73.2	**75.7**	**65.1**	**79.5**	**25.2**	40.3	**95.8**	**92.1**	**85.0**	**11.6**	**38.9**	**98.3**	30.5	**51.9**	**10.1**	28.7	**21.8**	52.5	**68.9**	**52.9**	15.9	45.7	**50.1**	**77.6**	**56.7**

Table 14: CiT trained on Raw Image-Text Crawl and evaluated on 26 CLIP benchmarks: We vary metadata from IN-1K (CiT-1K-meta), IN-21K (CiT-21K-meta) and combined class names from 26 benchmarks (CiT-multi.-meta). The budget $b = 60000$ for IN-21K and $b = 30000$ for combined class names. We also list results from CLIP on WIT400M.

YFCC15M	Food-101	CIFAR10	CIFAR100	CUB	SUN397	Cars	Aircraft	DTD	Pets	Caltech-101	Flowers	MNIST	FER-2013	STL-10	EuroSAT	RESISC45	GTSRB	KITTI	Country211	PCAM	UCF101	Kinetics700	CLEVR	HatefulMemes	SST2	ImageNet
# of classes	101	10	100	200	397	196	100	47	37	102	102	10	7	10	10	45	43	4	211	2	101	700	8	2	2	1000
$t > 0.55$																										
# pairs per class (k)	5.32	16.64	9.27	3.28	5.63	0.81	4.35	3.12	3.66	6.71	6.51	11.9	5.43	18.21	8.59	10.57	2.22	23.63	2.33	0.53	4.28	3.9	20.96	5.48	0.66	3.69
total keep rates (%)	3.66	1.13	6.31	4.46	15.2	1.08	2.96	0.999	0.922	4.66	4.52	0.81	0.259	1.24	0.585	0.324	0.65	0.644	3.34	0.007	2.95	18.6	1.14	0.075	0.009	25.1

Table 15: Statistics of YFCC15M (title and description) coverage on 26 tasks of CLIP evaluation: Low coverage could explain the root cause of the poor performance of zero-shot transfer (*e.g.* Cars, PCAM, etc.).

expect better accuracy for longer training.

A.2.4 Early Detection of Task Coverage

One extra benefit of curation is being able to detect the task coverage of the training data. Although existing scaled pretrainings have huge success, the coverage of pre-training data distribution for downstream tasks is largely unknown. We discuss the coverage for CiT on YFCC15M below.

Task Coverage. We obtain the statistics of curated data (offline in Table 1 (a) of the main paper) for the 26 tasks and show it in Table 15. We consider a sample with a maximum cosine similarity for one class as one sample belonging to that class/task. We note that this is a hard-matching which does not necessarily cover the full class to sample correlation. Breaking down YFCC15M for different tasks partially explains the low performance on some. For example, SST2 (a binary classification task) has low image-text pair matches, explaining the low performance (close to random) for all models.

A.3. Difference in Learning Paradigm

CiT incorporates data curation into its training process, thereby altering the learning paradigm from existing pretraining models that rely on human-driven offline filtering. A comparison of CiT with other approaches is presented in Table 16. The main difference between CiT and other approaches is that it accepts raw image-text pairs. Unlike CLIP/LiT, which require human-filtered datasets due to performance constraints, CiT can curate data during training. Although CLIP uses search queries, which is close to CiT's metadata, it is done offline on a larger scale.

Image-text pairs used in CLIP/LiT/CiT are much noisier than human-annotated data, such as an image-label pair commonly used in supervised learning, due to the nature of language that may or may not describe the image (e.g., file names). The goal of active learning is to selectively obtain labels for images from a fixed dataset and semi-supervised learning aims to create pseudo-labels for images. Therefore, established supervised/semi-supervised learning techniques may also select poor (noisy) examples (e.g., via active learning), while CiT tries to select quality pairs.

In Table 17, we apply some semi-supervised/deep active

	CiT	CLIP [21]	LiT [38]	DAL
Paradigm	curation&pre-training	pre-training	pre-training	sup. learning/fine-tuning
Data Type	**raw(online)** img-txt pairs	filtered img-txt pairs	filtered img-txt pairs	human annotated images
Offline Filtering	✗	✓	✓	✓
Initialization	uni-modal	from scratch	uni-modal	from scratch/uni-modal

Table 16: Comparison of CiT with existing approaches on learning paradigm.

Strategy	IN-1K Acc.
CiT [5]	61.4
Self-training	26.2
Active Learning	
Entropy Sampling	19.5
Least Confidence Sampling	23.0
Margin Sampling	50.5
BALD	53.7

Table 17: Comparison of CiT with different training paradigms, all using MoCo-v3 backbones, on YFCC15M.

learning (DAL) strategies[4] on YFCC15M, using the text as class names. The results show that these strategies are sub-optimal and CiT is much more effective in handling the noisy image-text data.

[4] https://github.com/ej0cl6/deep-active-learning

References

[1] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *arXiv preprint arXiv:2204.14198*, 2022.

[2] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9650–9660, 2021.

[3] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3558–3568, 2021.

[4] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709*, 2020.

[5] Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9640–9649, 2021.

[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[7] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2020.

[8] Tianyu Gao, Xingcheng Yao, and Danqi Chen. Simcse: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910, 2021.

[9] Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A Smith. Don't stop pretraining: adapt language models to domains and tasks. *arXiv preprint arXiv:2004.10964*, 2020.

[10] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. *arXiv preprint arXiv:2111.06377*, 2021.

[11] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9729–9738, 2020.

[12] 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.

[13] Krishnateja Killamsetty, Durga Sivasubramanian, Ganesh Ramakrishnan, and Rishabh Iyer. Glister: Generalization based data subset selection for efficient and robust learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 8110–8118, 2021.

[14] Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240, 2020.

[15] Junlong Li, Zhuosheng Zhang, Hai Zhao, Xi Zhou, and Xiang Zhou. Task-specific objectives of pre-trained language models for dialogue adaptation. *arXiv preprint arXiv:2009.04984*, 2020.

[16] Yangguang Li, Feng Liang, Lichen Zhao, Yufeng Cui, Wanli Ouyang, Jing Shao, Fengwei Yu, and Junjie Yan. Supervision exists everywhere: A data efficient contrastive language-image pre-training paradigm. *arXiv preprint arXiv:2110.05208*, 2021.

[17] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

[18] Antoine Miech, Jean-Baptiste Alayrac, Lucas Smaira, Ivan Laptev, Josef Sivic, and Andrew Zisserman. End-to-end learning of visual representations from uncurated instructional videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9879–9889, 2020.

[19] Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. In *Proceedings of the IEEE international conference on computer vision*, pages 2630–2640, 2019.

[20] Norman Mu, Alexander Kirillov, David Wagner, and Saining Xie. Slip: Self-supervision meets language-image pre-training. *arXiv preprint arXiv:2112.12750*, 2021.

[21] 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. *arXiv preprint arXiv:2103.00020*, 2021.

[22] Shibani Santurkar, Yann Dubois, Rohan Taori, Percy Liang, and Tatsunori Hashimoto. Is a caption worth a thousand images? a controlled study for representation learning. *arXiv preprint arXiv:2207.07635*, 2022.

[23] Christoph Schuhmann, Romain Beaumont, Cade W Gordon, Ross Wightman, Theo Coombes, Aarush Katta, Clayton Mullis, Patrick Schramowski, Srivatsa R Kundurthy, Katherine Crowson, et al. Laion-5b: An open large-scale dataset for training next generation image-text models.

[24] 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.

[25] 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 the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2556–2565, Melbourne, Australia, July 2018. Association for Computational Linguistics.

[26] Amanpreet Singh, Ronghang Hu, Vedanuj Goswami, Guillaume Couairon, Wojciech Galuba, Marcus Rohrbach, and Douwe Kiela. Flava: A foundational language and vision alignment model. *arXiv preprint arXiv:2112.04482*, 2021.

[27] Mannat Singh, Laura Gustafson, Aaron Adcock, Vinicius de Freitas Reis, Bugra Gedik, Raj Prateek Kosaraju, Dhruv Mahajan, Ross Girshick, Piotr Dollár, and Laurens van der Maaten. Revisiting weakly supervised pre-training of visual perception models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 804–814, 2022.

[28] Andreas Steiner, Alexander Kolesnikov, Xiaohua Zhai, Ross Wightman, Jakob Uszkoreit, and Lucas Beyer. How to train your vit? data, augmentation, and regularization in vision transformers. *arXiv preprint arXiv:2106.10270*, 2021.

[29] Bart Thomee, David A Shamma, Gerald Friedland, Benjamin Elizalde, Karl Ni, Douglas Poland, Damian Borth, and Li-Jia Li. Yfcc100m: The new data in multimedia research. *Communications of the ACM*, 59(2):64–73, 2016.

[30] Kai Wei, Rishabh Iyer, and Jeff Bilmes. Submodularity in data subset selection and active learning. In *International conference on machine learning*, pages 1954–1963. PMLR, 2015.

[31] Mitchell Wortsman, Gabriel Ilharco, Jong Wook Kim, Mike Li, Simon Kornblith, Rebecca Roelofs, Raphael Gontijo Lopes, Hannaneh Hajishirzi, Ali Farhadi, Hongseok Namkoong, et al. Robust fine-tuning of zero-shot models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7959–7971, 2022.

[32] Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3733–3742, 2018.

[33] Hu Xu, Gargi Ghosh, Po-Yao Huang, Dmytro Okhonko, Armen Aghajanyan, Florian Metze, Luke Zettlemoyer, and Christoph Feichtenhofer. Videoclip: Contrastive pre-training for zero-shot video-text understanding. In *Proceedings of*

the 2021 Conference on Empirical Methods in Natural Language Processing, pages 6787–6800, 2021.

[34] Hu Xu, Bing Liu, Lei Shu, and S Yu Philip. Bert post-training for review reading comprehension and aspect-based sentiment analysis. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2324–2335, 2019.

[35] Andrew Yates, Rodrigo Nogueira, and Jimmy Lin. Pretrained transformers for text ranking: Bert and beyond. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*, pages 1154–1156, 2021.

[36] Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. Coca: Contrastive captioners are image-text foundation models. *arXiv preprint arXiv:2205.01917*, 2022.

[37] Lu Yuan, Dongdong Chen, Yi-Ling Chen, Noel Codella, Xiyang Dai, Jianfeng Gao, Houdong Hu, Xuedong Huang, Boxin Li, Chunyuan Li, et al. Florence: A new foundation model for computer vision. *arXiv preprint arXiv:2111.11432*, 2021.

[38] Xiaohua Zhai, Xiao Wang, Basil Mustafa, Andreas Steiner, Daniel Keysers, Alexander Kolesnikov, and Lucas Beyer. Lit: Zero-shot transfer with locked-image text tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18123–18133, 2022.

[39] Xuan Zhang, Pamela Shapiro, Gaurav Kumar, Paul McNamee, Marine Carpuat, and Kevin Duh. Curriculum learning for domain adaptation in neural machine translation. *arXiv preprint arXiv:1905.05816*, 2019.

[40] Yuhao Zhang, Hang Jiang, Yasuhide Miura, Christopher D Manning, and Curtis P Langlotz. Contrastive learning of medical visual representations from paired images and text. *arXiv preprint arXiv:2010.00747*, 2020.