

A SETUP AND EXPERIMENTAL DETAILS

A.1 DATASETS

In Appendix Figure 6, we present (random) samples from the MS-COCO (Lin et al., 2014), Conceptual Captions (Sharma et al., 2018) and YFCC datasets (Thomee et al., 2016). We use the 2017 version of COCO, which contains five human-written captions along with multi-object image labels for each image.



- "A table topped with plates and glasses with eating utensils.."
- "a fork is laying on a small white plate"
- "dirty dishes on a table, and a bottle of something."
- "a table top with some dishes on top of it",
- "A table full of dirty dishes is pictured in this image."



- "An All Nippon Airways 777 sitting at a gate on the tarmac."
- "a large air plane on a run way"
- "A jumbo jet being serviced at an airport."
- "A large blue and white jetliner sitting on top of a tarmac.",
- "A large airliner preparing for departure at an airport."



- "A man jumping a horse over an obstacle."
- "A person jumping a horse over an object."
- "An equestrian competitor and his horse jumping over a stile"
- "A horse and jockey jump over bush hurdles .",
- "A rider and horse jump over a wooden brush obstacle."



- "A couple of zebra standing on top of a dirt field."
- "Some zebras walking around in a field looking around"
- "Some very cute zebras in a big dusty field."
- "A small zebra standing next to a bigger zebra.",
- "The baby Zebras stripes are much closer together than an adults."

Figure 6: Dataset examples: MS-COCO (Lin et al., 2014)

Licenses. These datasets were obtained by scraping images from online hosting services (e.g. Flickr). Thus, the ownership of the images lies with the respective individuals that uploaded them. Nevertheless, as per their terms of agreement, the images can be used for research purposes.



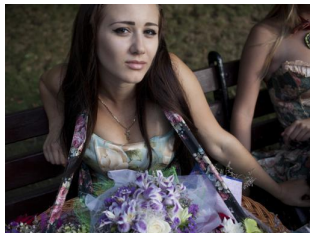
“paratroopers load onto a helicopter.”



“Close up hands of woman typing text message on smart phone in a cafe.”



“woman in a bathrobe is smiling to camera in the forest”



“Girls in old time dresses selling flowers are pictured taking a rest of a bench.”



“A shrimp has pairs of legs.”

Figure 6: Dataset examples: Conceptual Captions (Sharma et al., 2018)



“Kenneth Phan #7 A Day in the Life of DC is a photo project meant to capture a flavor of the region through the eyes of the participants. Participants submitted twelve photos taken on May 30, 2009. Photos by Kenneth Phan”



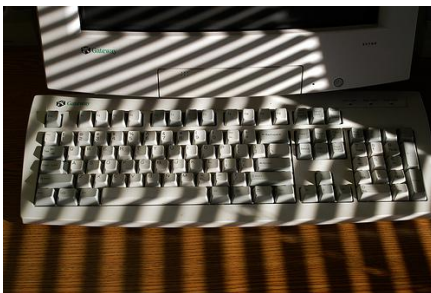
“Pombas New York - USA 27 de Setembro 2013”



“Um, the girls that live at the house I lived in 13 years ago are huge @foursquare fans! #amazing @ 600 euclid 4sq.commPKNZv (posted via FlickrSquare)”



“squares Created for dA Users Gallery Challenge #43 – Winter Stock 1 Model with thanks to Reine-Haru”



“Stripes and Squares Love the contrast of the light on the keyboard, stripes and squares”

Figure 6: Dataset examples: YFCC (Thomee et al., 2016)



Random examples of “full” captions:

- “A light bulb. A brown bulletin board. The bookcases are full. Books and papers on the floor. The carpet is brown. Books on the shelf. A flexible desk lamp. Magazines on the bench. Corded telephone mounted on wall. a book on a book shelf. Books on the desk. Sticker on the box is white and red. The white cord of the phone.”
- “Bright light on the ceiling. A short brown wallpaper. Desktop monitor and keyboard. A set of items on the floor. The bookcases are full. Books and papers on the floor. The carpet is brown. Large clock on crowded bookshelf. A black table lamp not lighted. Black desk chair. Picture frame with three pictures. Sticker on the box is white and red. The wall phone is white. The white cord of the phone.”
- “A brown bulletin board. Monitor on the table. Pile of books on floor. The brown wallpaper. Black padded piano bench. The carpet is brown. Books on the shelf. Magazines under a black psp. Corded telephone mounted on wall. White and black clock. Chair by the desk. Picture frame with three pictures. Sticker on the box is white and red. The wall phone is white. The white cord of the phone.”

Random examples of “quadrant” captions:

- “A green plastic object on the table. Magazines under a black psp. Cardboard box with red and white sticker. Magazines on the bench.”
- “Magazines on the bench.”
- “Green package on top of cardboard box. Magazines sitting on top of stool. Magazines under a black psp. Magazines on the bench.”



Random examples of “full” captions:

- “A blue double decker bus. a window on a building. a window on a building. Yellow and blue door. White clouds in blue sky. Cross on top of the building. a window on a bus. Tire has blue rim. Rectangular window with curve on sides. Number 35 on the back of the bus.”
- “White clouds in blue sky. Bus. Building has a window. Varta printed in blue. Stickers on the window. White clouds in blue sky. Cross on top of the building. a window on a bus. Rectangular window with curve on sides. The bus number is 35. Number 220 on the back of the bus.”
- “Building has a window. Building has a window. a window on a building. A parked double decker bus. Varta printed in blue. Folding door of the bus. White clouds in blue sky. Cross on top of the building. a window on a bus. Rectangular window with curve on sides. The bus number is 35. 220 printed in gray.”

Random examples of “quadrant” captions:

- “David printed in black.”
- “Logo on the side of the bus. Windows with papers on it. Stickers on the window..”
- “Building has a window. Building has a window. David printed in black. Windows with papers on it.”

Figure 6: Dataset examples: VisualGenome (Krishna et al., 2017)

Like most large-scale datasets, COCO, CC and YFCC have not been extensively vetted, and may contain identifying information or offensive content. Characterizing the pervasiveness of these issues is an important and active area of research. That being said, we do not redistribute the data, our work is unlikely to significantly further the risks from these datasets.

A.2 MODELS

We rely on existing open source implementations of CLIP (Ilharco et al., 2021) and SimCLR (Falcon & the PyTorch Lightning team, 2019) for all our experiments, with a ResNet-50 image encoder (feature dimension=2048), and a linear and MLP projection head respectively. We use the transformer architecture from Radford et al. (2021) for encoding captions in CLIP. Unless otherwise specified, we use five captions per image to train CLIP_s. For downstream transfer, we train a linear probe using task data.

Supervised baseline on COCO. We trained a ResNet-50 classifier from scratch on the COCO dataset. The classifier was trained to predict the presence/absence of each of the 80 object classes per image (i.e., 80 binary classification tasks) as COCO has multi-object labels per image. We then evaluate the accuracy of the model by aggregating (in a class balanced manner) the correctness of each of these binary predictions.

A.3 HYPERPARAMETERS

We ran an extensive hyperparameter grid for CLIP and SimCLR on MS-COCO and used the same configuration in the rest of our experiments. These defaults are stated in Appendix Table 3.

Model	Batch Size	Epochs	Warmup	lr	wd
Supervised	1024	200	10	10^{-3}	10^{-6}
SimCLR	1024	200	10	10^{-2}	10^{-6}
CLIP	1024	200	10	10^{-3}	0.1

Table 3: Default hyperparameters for model training.

We use the Adam optimizer with a cosine lr schedule for all the models. All other hyperparameters are defaults from standard implementations of SimCLR ⁴ and CLIP.⁵

Exceptions. We train CLIP/SimCLR on CC/YFCC-2M for 100 epochs due to computational restrictions. For corpora smaller than 100K (Figure 2), we scale up the number of epochs to keep the number of iterations roughly comparable.

Data augmentations. The PyTorch pseudo-code for the default SimCLR and CLIP data from prior work augmentation are as follows:

$$\begin{aligned}
 T_{SimCLR} &= \{ \text{RandomResizedCrop}(\text{size} = 224), \\
 &\quad \text{RandomHorizontalFlip}(p = 0.5) \\
 &\quad \text{RandomApply}(\text{ColorJitter}(0.8, 0.8, 0.8, 0.2), p = 0.8) \\
 &\quad \text{RandomGrayscale}(p = 0.2), \\
 &\quad \text{GaussianBlur}(\text{kernel_size} = 23, p = 0.5) \} \\
 T_{CLIP} &= \{ \text{RandomResizedCrop}(\text{size} = 224, \\
 &\quad \text{scale} = (0.9, 1.0), \\
 &\quad \text{interpolation} = \text{BICUBIC}) \}
 \end{aligned}$$

Note that for our experiments, unless otherwise specified, we use the standard SimCLR set for the Supervised/SimCLR/CLIP models.

⁴https://pytorch-lightning-bolts.readthedocs.io/en/latest/self_supervised_models.html

⁵https://github.com/mlfoundations/open_clip

Linear probe. We train the probe using cross-entropy loss on CLIP/SimCLR features of dimensionality 2048. In cases where the downstream task data is imbalanced, we re-weight the loss to account for it. We also evaluate class balanced accuracy at test time. For each downstream task, we train the probe for 250 epochs using an SGD optimizer. We use a batch size of 256, weight decay of 10^{-6} and momentum 0.9. We perform a grid search for the best learning rate (using the validation set), considering values between 3×10^{-2} and 10. We also consider 3 random seeds.

Confidence intervals. We report 95% confidence intervals obtained via bootstrapping over the test set, as well as the three random seeds used for the linear probe. Due to space constraints, we do not always report them in the main paper, but include a detailed table for all our experiments with confidence intervals in the Appendix.

A.4 COMPUTE

We train each of our models of 4 NVIDIA A100 GPUs. Training both CLIP and SimCLR models takes on the order of 8-10 hours for a pre-training corpus of size $\sim 100K$.

A.5 BLIP RECAPTIONING

To generate BLIP captions for images from the CC and YFCC datasets, we use the BLIP captioning model (Li et al., 2022). In particular, we use the provided⁶ ViT-Base with nucleus sampling (top_p of 0.9, repetition penalty of 1.1, and text length range [5, 40]), varying the random seed to generate multiple captions per image. (Random) image-BLIP caption pairs are shown in Appendix Figure 7.



Dataset: CC

- "portrait of a young boy sitting in the leaves in a park - stock image."
- "toddler boy in a coat sitting on leaves with arms up to the air, smiling and laughing - stock photo."
- "An image of a little boy sitting on the leaves in a park - stock image."



Dataset: CC

- "The men are walking on a dirt ground with equipment in the background."
- "military soldiers in uniforms carrying weapons and soldiers on their back in a desert"
- "Officials and soldiers stand in the desert, looking at a vehicle with missiles."



Dataset: YFCC

- "Signs of various silhouettes of people dancing, standing, and laying on the street."
- "lot of bronze colored women holding their arms up with their hands together in front of a metal wall art"
- "Sculptures on a wall of various silhouettes and dance positions."



Dataset: YFCC

- "I love the white swan in the foreground with the water behind him."
- "an image group of white birds in a green area near some water"
- "the swan is standing on the green grass near the water"

Figure 7: Random images from CC and YFCC alongside BLIP captions.

⁶checkpoint https://storage.googleapis.com/sfr-vision-language-research/BLIP/models/model_base_caption_capfilt_large.pth from <https://github.com/salesforce/BLIP>

A.6 SYNTHETIC COCO CAPTIONS

We construct these captions for MS-COCO using the available multi-object labels (see Figure 8 for examples). A synthetic caption is complete (incomplete) if it describes all (a random subset) of objects in the image. It is consistent (inconsistent) if it describes a given object using a single consistent term throughout the dataset (one from a set of manually curated synonyms) and whether we use a fixed template (one of a set of templates). In every case, we randomly order the objects that we describe.

As an example of this procedure, consider two images X_1 with labels [“plate”, “cup”, “cup”, “refrigerator”] and X_2 with labels [“refrigerator”, “cup”, “potted plant”, “tennis racket”].

- A consistent and complete caption would describe *all* the image objects using a single descriptor/object in random order. For instance:
Caption(X_1) = “An image of a plate, two cups, and a refrigerator”.
Caption(X_2) = “An image of a refrigerator, potted plant, tennis racket and cup”.
- On the other hand, consistent and incomplete captions would still use a single descriptor/object, but might (randomly) omit certain image objects. For instance,
Caption(X_1) = “An image of a plate and cup”.
Caption(X_2) = “An image of a refrigerator, potted plant and tennis racket”.
- Finally, an inconsistent caption uses one of a set of descriptors/object in the image. For instance, a cup might be described (randomly) as a “cup” or “glass”. We also randomly vary the template. Concretely, inconsistent and incomplete captions might look like:
Caption(X_1) = “An image of a plate and glass”.
Caption(X_2) = “An photo of a fridge, tennis racket and a cup”.

For inconsistent captions, the set of templates we consider is: “A photo of {}”, “There are {}”, “{} together”, “I see {}”, “Shown here are {}”, “You can see {}”. The set of synonyms per COCO object are:

```

person: ['human', 'man', 'woman', 'individual', 'person']
bicycle: ['bike', 'two-wheeler', 'cycle']
car: ['auto', 'car', 'motorcar', 'automobile']
motorcycle: ['bike', 'two-wheeler', 'motorbike', 'moped',
'scooter']
airplane: ['plane', 'airplane', 'aeroplane', 'air plane', 'jet',
'airplane']
bus: ['vehicle', 'coach']
train: ['locomotive', 'engine', 'carriage', 'wagon']
truck: ['hand truck', 'truck', 'motortruck', 'motor truck']
boat: ['ship', 'boat', 'vessel', 'watercraft']
traffic light: ['light', 'stoplight', 'traffic signal', 'signal',
'stop light']
fire hydrant: ['hydrant', 'firehydrant']
stop sign: ['stop', 'road sign', 'sign']
parking meter: ['meter', 'parking', 'pay station', 'parking
kiosk']
bench: ['seat', 'park bench', 'seat']
bird: ['bird', 'birdie']
cat: ['cat', 'kitten']
dog: ['puppy', 'pooch', 'canine']
horse: ['horse', 'stallion']
sheep: ['sheep']
cow: ['cow']
elephant: ['elephant']
bear: ['bear']
zebra: ['zebra']
giraffe: ['giraffe']
backpack: ['knapsack', 'backpack', 'rucksack', 'haversack',
'packsack']

```

umbrella: ['umbrella']
handbag: ['bag', 'handbag', 'purse']
tie: ['tie', 'necktie']
suitcase: ['bag', 'traveling bag', 'suitcase']
frisbee: ['Frisbee']
skis: ['ski']
snowboard: ['snowboard']
sports ball: ['ball', 'basketball', 'football']
kite: ['kite']
baseball bat: ['bat']
baseball glove: ['glove']
skateboard: ['skateboard', 'skate board']
surfboard: ['surfboard', 'surf board']
tennis racket: ['tennis racket', 'racket']
bottle: ['bottle', 'jar']
wine glass: ['glass']
cup: ['mug', 'cup', 'shot glass']
fork: ['fork']
knife: ['knife']
spoon: ['spoon']
bowl: ['bowl', 'plate']
banana: ['banana']
apple: ['apple', 'granny smith']
sandwich: ['sandwich', 'burger']
orange: ['Orange', 'mandarin', 'clementine']
broccoli: ['broccoli']
carrot: ['carrot']
hot dog: ['hot dog']
pizza: ['pie', 'pizza']
donut: ['donut', 'doughnut']
cake: ['pastry', 'dessert', 'cake']
chair: ['chair']
couch: ['couch', 'lounge', 'sofa']
potted plant: ['plant', 'houseplant']
bed: ['bed']
dining table: ['table']
toilet: ['crapper', 'toilette', 'potty', 'lavatory', 'lav',
'can', 'bathroom', 'privy', 'toilet']
tv: ['television receiver', 'telly', 'television', 'TV', 'tv
set', 'tv']
laptop: ['laptop', 'laptop computer', 'notebook']
mouse: ['computer mouse', 'mouse']
remote: ['remote control', 'remote']
keyboard: ['keyboard']
cell phone: ['phone', 'mobile', 'cell phone', 'cell']
microwave: ['microwave oven', 'microwave']
oven: ['oven']
toaster: ['toaster']
sink: ['basin', 'sink']
refrigerator: ['icebox', 'refrigerator', 'fridge']
book: ['book', 'novel', 'textbook', 'story book']
clock: ['clock', 'watch', 'wall clock']
vase: ['vase', 'jar']
scissors: ['scissors']
teddy bear: ['stuffed toy', 'toy']
hair drier: ['dryer', 'blow dryer']
toothbrush: ['toothbrush']

A.7 VISUAL GENOME CAPTIONS

In our experiments, we also consider training CLIP models on the Visual Genome dataset (Krishna et al., 2017). Here, we construct image captions using the available region descriptions. Concretely, for each image, we construct:

- 10 captions by randomly subsampling *all* the available region descriptions and concatenating them.
- 10 captions by randomly subsampling the available region descriptions in the *first quadrant alone* and concatenating them. We do so in order to further understand the effect of incompleteness (i.e., describing only a part of the image) in captions on CLIP’s performance.

Examples of the generated captions are shown in Appendix Figure 6

Using the two sets of captions per image (“full” or “quadrant”) we train CLIP (using a single caption per image) and CLIP_S (stochastically sampling one of the ten captions) models. We then measure the transfer performance of the models in Appendix Table 9.

A.8 MEASURING CAPTION VARIABILITY

In Section 3.3, we attempt to quantify the variability of captions corresponding to a given real-world dataset. To do so, we look to prior work in linguistics and natural language processing (discussed below) which has sought to study similar quantities. In our setting, we treat the entire set of dataset captions as a single document so that we can measure variability across them.

Measure of Lexical Diversity (MTLD). In linguistics research, the lexical diversity of a given text—the range of words used within it—has been long studied (see (Jarvis, 2013) for a discussion). As discussed in (Baese-Berk et al., 2021), “...samples with (lexical) greater diversity, may also include less repetition, more switches among topics, and use of multiple lexical items to refer to the same concept”. A classical measure of lexical diversity is the *type-token ratio (TTR)*: the ratio of unique words with respect to the total number of words in the text. However, this metric suffers from certain drawbacks including sensitivity to corpus length. To mitigate these drawbacks, McCarthy & Jarvis (2010) proposed the notion of *measure of lexical diversity* or MTLD. Intuitively, this measure captures the average length of words (within the given text) for which the TTR remains constant. A higher MTLD score indicates that the document contains less repetitive (more variable) tokens. We use a standard implementation⁷ to measure MTLD over the entire set of captions.

Unique n-grams. In the context of conversational agents in natural language processing, the number of unique n-grams is often used to assess the diversity of the generated text Li et al. (2015); Fung et al. (2020). We count the sum of unique 1, 2, 3-grams for the set of captions in a given dataset.

A.9 FILTERING CAPTIONS

In Section 4, we introduce a methodology to filter poor quality captions from a given source dataset. Using the fastText library,⁸ we train a linear classifier bag-of-n-grams sentence embeddings (n=2) to distinguish a subset of source captions from those in the COCO validation set. We then use the classifier to filter the source dataset (CC/YFCC), only selecting the ones that are (mis)classified as being COCO like.

In Appendix Figure 9, we present a (random) subset of filtered examples from the YFCC dataset. Compared to random YFCC samples (cf. Appendix Figure 6), the ones in Appendix Figure 9 have much shorter captions—often without attributes such as dates, urls and hashtags. This difference is more apparent if we consider, for instance, the top-30 most frequent 1-grams mentioned in YFCC captions before and after filtering.

Before filtering: -, new, photo, day, taken, one, &, photos, see, like, 2013, view, 2012, park, first, 2011, around, old, part, 2010, \,

⁷https://github.com/jennafrens/lexical_diversity

⁸<https://github.com/facebookresearch/fastText>

*Complete and Consistent:*

- "A photo of four bowls, a oven, seven cups, a refrigerator, two persons, a spoon, two cakes"

Incomplete and consistent:

- "A photo of a person, six cups, three bowls, two cakes, a oven."

Incomplete and Inconsistent:

- "A photo of a kitchen, two women, two shot glasses."
- "I see a oven, a kitchen, two mugs, a kitchen, a man."

*Complete and Consistent:*

- "A photo of a person, a tennis racket, a sports ball, a car."

Incomplete and consistent:

- "A photo of a car, a person, a tennis racket."

Incomplete and Inconsistent:

- "a sports ball, a motorcar together."
- "There is a woman."

Figure 8: Random image samples from MS-COCO alongside our synthetic captions.

city, @, street, national, time, please, york, state, :, house, center, san, may, visit, go, research, near, use, back, get, 2008, 2, south, great, lake, two, central, north, little

After filtering: street, looking, two, water, people, one, beach, white, man, view, road, train, near, black, sign, front, blue, side, red, old, snow, small, tree, bridge, next, playing, river, top, day, little, room, walking, light, back, three, station, around, sitting, big, window, car, table, outside, dog, green, park, food, taken, ready, picture

A.10 AUGMENTING CAPTIONS WITH GPT-J

We propose a methodology to augment captions contained in existing datasets by using a pre-trained language model (in our case GPT-J-6b, referred to as GPT-J) to paraphrase them (Section 4). To this end, we rely on in-context learning, wherein we provide GPT-J with some (four) paired caption-paraphrase examples (using the five human-provided COCO captions) as the context. We then ask GPT-J to paraphrase a given target caption. For instance, a query to the model might look like:

Paraphrase the sentence below

Input: A little boy standing next to a dog in a field.

Output: A dog parked filled with people and a bunch of different dogs.

Paraphrase the sentence below

Input: Some people are on the sandy beach flying kites.

Output: a sunny day at the beach with colorful kites in the sky

Paraphrase the sentence below



“Orc/Troll There’s a face only a mother could love.”



“Pedal Board 9 Back Camera.”



“Kittens Morrissey and Marr relax on the bed.”



“Park in Santa Clara Complete with Che graffiti.”

Figure 9: Filtered image-caption pairs from the YFCC dataset (see Section 4).

Input: A living room filled with furniture and a table.

Output: A living room with a nice couch and a coffee table.

Paraphrase the sentence below

Input: A couple of people on a surfboard in the ocean.

Output: A dog is lying on the surfboard as it surfs on a wave.

Paraphrase the sentence below

Input: a handful of snowmen make this with a glove .

Output:

We use temperature sampling to generate multiple diverse captions for a given image-caption pair from the dataset. Examples for the CC dataset are shown in Appendix Figure 10.

CC caption: “young beautiful businesswoman holding the documents and crossing the road in financial district .”

GPT-J paraphrases:

- “A women in business attire crossing the road in financial district.”
- “A young beautiful businesswoman is crossing the road.”
- “A young woman crossing the road in financial district.”

CC caption: “Island with waterfalls and rainbows floating above clouds in a dreamy sky – stock photo #”

GPT-J paraphrases:

- “An island with waterfalls and rainbows floating above clouds in a dreamy sky.”
- “A rainbow in the sky with a cloud and a rainbow in the water”
- “A rainbow floating in a dreamy sky and a waterfall”

CC caption: “Young friends eating pizza in the park”

GPT-J paraphrases:

- “The young friends are eating pizza in the park”
- “A group of young people eating pizza in the park”
- “Some kids are eating pizza in the park”

CC caption: “Actors attend the spring fashion show during fashion week.”

GPT-J paraphrases:

- “A fashion show that’s attended by actors.”
- “People attending a spring fashion show during fashion week.”
- “The spring fashion show was held during fashion week.”

Figure 10: Caption paraphrases generated using in-context learning with GPT-J.

B ADDITIONAL EXPERIMENTS

In Appendix Tables 4-13, we report per-task performance for all our experiments. In Appendix Table 4, we also illustrate the performance of SimCLR/CLIP models trained using the simpler data augmentations typically used for CLIP training (cf. Appendix A.3). One can see that both models perform worse with this modification—with the performance of CLIP dropping by 10% and that of SimCLR by 50%.

For COCO, we also consider a variant of SimCLR, which we refer to as SimCLR_{+lab} , that factors in label information in the transformation $T(x)$. Specifically, for a given image x , x_+ is a data augmented version of another COCO image which has at least one object in common with x . We see that factoring label information does improve SimCLR’s performance considerably, putting it between vanilla CLIP and CLIP_S . However, note for typical pre-training datasets such as CC and YFCC, we do not have access to such “expert” object labels. Instead, we can take advantage of captions to improve the equivalences learned by the model.

Model	SUP	SimCLR ₋	SimCLR	SimCLR _{+lab}	CLIP ₋	CLIP	CLIP _S
COCO	90.5 ± 1.5	60.4 ± 2.4	88.9 ± 1.6	89.3 ± 1.5	84.9 ± 1.9	88.4 ± 1.7	89.8 ± 1.6
Aircraft	31.6 ± 0.9	2.3 ± 0.3	40.6 ± 1.0	47.0 ± 1.0	30.3 ± 1.0	41.4 ± 1.0	46.4 ± 1.0
Birdsnap	11.8 ± 0.4	0.7 ± 0.1	18.5 ± 0.5	20.8 ± 0.5	14.0 ± 0.4	17.6 ± 0.5	20.0 ± 0.5
Cal101	65.8 ± 0.7	3.8 ± 0.3	71.5 ± 0.7	80.4 ± 0.6	53.6 ± 0.8	73.2 ± 0.7	78.4 ± 0.6
Cal256	53.7 ± 0.5	3.1 ± 0.2	58.6 ± 0.4	65.7 ± 0.4	41.5 ± 0.5	60.4 ± 0.5	65.6 ± 0.5
Cars	21.7 ± 0.5	1.2 ± 0.1	31.4 ± 0.6	39.3 ± 0.7	23.4 ± 0.5	35.8 ± 0.6	41.5 ± 0.6
CIFAR-10	74.8 ± 0.5	23.2 ± 0.5	82.1 ± 0.4	81.5 ± 0.5	74.0 ± 0.5	83.6 ± 0.4	84.6 ± 0.4
CIFAR-100	46.7 ± 0.6	6.0 ± 0.3	57.3 ± 0.6	56.8 ± 0.6	50.4 ± 0.6	60.8 ± 0.6	62.5 ± 0.6
DTD	55.9 ± 1.4	6.2 ± 0.6	61.7 ± 1.3	60.3 ± 1.3	48.2 ± 1.4	65.7 ± 1.3	66.7 ± 1.3
Flowers	63.5 ± 0.7	4.6 ± 0.3	77.4 ± 0.6	81.4 ± 0.6	68.2 ± 0.7	80.5 ± 0.6	84.0 ± 0.6
Food	47.1 ± 0.4	4.0 ± 0.1	58.7 ± 0.3	56.4 ± 0.4	51.8 ± 0.4	60.9 ± 0.4	65.3 ± 0.4
Pets	45.9 ± 1.0	6.3 ± 0.5	57.3 ± 0.9	63.0 ± 0.9	44.6 ± 0.9	57.0 ± 0.9	61.2 ± 0.9
SUN	44.5 ± 0.4	1.3 ± 0.1	51.9 ± 0.4	52.2 ± 0.4	37.6 ± 0.4	50.8 ± 0.4	54.9 ± 0.4
μ_{Tx}	47.2 ± 0.2	5.2 ± 0.1	56.0 ± 0.2	58.7 ± 0.2	44.8 ± 0.2	57.5 ± 0.1	61.3 ± 0.2

Table 4: Extended comparison of transfer performance of supervised, SimCLR and CLIP pre-trained models. Here SimCLR₋ and CLIP₋ denote models trained with the default CLIP data augmentation transforms instead of the SimCLR ones (cf. Appendix A.3). SimCLR_{+lab} refers to SimCLR models trained by picking x_+ to be a different image with the same label as x .

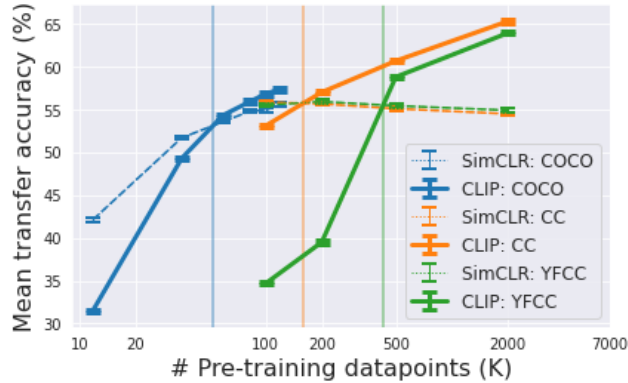


Figure 11: Transfer performance of SimCLR, CLIP and CLIP_S as we vary the number of COCO samples used for pre-training.

Model	CLIP	CLIP	CLIP _S	CLIP	CLIP _S
Complete	✓	✗	✗	✗	✗
Consistent	✓	✗	✗	✓	✓
COCO	88.8 ± 1.7	88.4 ± 1.7	89.3 ± 1.6	88.3 ± 1.7	89.2 ± 1.5
Aircraft	46.6 ± 1.0	44.5 ± 1.0	46.6 ± 1.0	45.6 ± 1.0	45.8 ± 1.0
Birdsnap	18.9 ± 0.5	17.2 ± 0.5	18.6 ± 0.5	18.5 ± 0.5	19.1 ± 0.5
Cal101	77.3 ± 0.6	75.3 ± 0.7	76.8 ± 0.6	76.1 ± 0.7	76.0 ± 0.6
Cal256	63.3 ± 0.5	59.9 ± 0.5	63.0 ± 0.4	61.4 ± 0.5	63.6 ± 0.5
Cars	42.4 ± 0.6	41.6 ± 0.6	42.7 ± 0.7	41.2 ± 0.6	42.8 ± 0.6
CIFAR-10	83.3 ± 0.4	82.4 ± 0.4	82.9 ± 0.4	83.7 ± 0.4	83.2 ± 0.4
CIFAR-100	60.5 ± 0.6	59.0 ± 0.6	58.9 ± 0.5	59.9 ± 0.5	60.1 ± 0.6
DTD	64.3 ± 1.3	63.7 ± 1.3	66.1 ± 1.3	63.4 ± 1.2	65.2 ± 1.2
Flowers	82.1 ± 0.5	78.3 ± 0.6	79.5 ± 0.6	79.3 ± 0.6	80.6 ± 0.6
Food	61.4 ± 0.3	57.6 ± 0.4	60.9 ± 0.4	59.0 ± 0.3	61.9 ± 0.4
Pets	60.0 ± 0.9	57.1 ± 1.0	58.8 ± 0.9	59.8 ± 0.9	60.6 ± 1.0
SUN	52.1 ± 0.4	49.6 ± 0.4	53.1 ± 0.4	50.6 ± 0.4	52.7 ± 0.4
μ_{Tx}	59.2 ± 0.1	56.6 ± 0.2	58.9 ± 0.2	57.7 ± 0.2	59.3 ± 0.2

Table 5: The impact of intra-dataset variations in captions on CLIP’s transfer performance. Here, we use synthetic captions for pre-training, constructed using COCO multi-object image labels. We vary whether these captions are consistent (i.e., do they use a single term to describe a given object?) and complete (i.e., do they describe all image objects?). We also consider a variant of CLIP, CLIP_S, which uses multiple captions per image.

	Aircraft	Birdsnap	Ctech101	Ctech256	Cars	CIFAR10	CIFAR100	DTD	Flowers	Food-101	Pets	SUN937
SimCLR	40.6	18.5	71.5	58.6	31.5	82.1	57.3	61.7	77.4	58.7	57.3	51.9
CLIP	41.4	17.6	73.2	60.4	35.8	83.6	60.8	65.7	80.5	60.9	57.0	50.8
CLIP + SimCLR loss	40.4	17.8	81.5	61.6	36.3	84.3	62.1	67.1	79.2	62.0	58.1	53.2

Table 6: Effect of incorporating SimCLR loss into CLIP on downstream transfer performance.

Model Dataset size	SimCLR				CLIP			
	100K	200K	500K	2M	100K	200K	500K	2M
Aircraft	40.5 \pm 1.0	40.3 \pm 1.0	39.3 \pm 0.9	37.9 \pm 1.0	35.5 \pm 1.0	39.9 \pm 1.0	41.6 \pm 1.0	45.1 \pm 1.0
Birdsnap	20.2 \pm 0.5	20.7 \pm 0.5	20.6 \pm 0.5	20.6 \pm 0.5	15.1 \pm 0.5	17.5 \pm 0.5	19.8 \pm 0.5	24.0 \pm 0.6
Cal101	70.7 \pm 0.7	70.3 \pm 0.7	70.3 \pm 0.7	69.0 \pm 0.7	67.7 \pm 0.8	73.5 \pm 0.7	79.0 \pm 0.6	84.8 \pm 0.6
Cal256	57.7 \pm 0.5	57.3 \pm 0.5	57.3 \pm 0.5	56.7 \pm 0.5	54.4 \pm 0.4	60.0 \pm 0.5	65.9 \pm 0.4	73.9 \pm 0.4
Cars	33.3 \pm 0.6	31.2 \pm 0.6	29.6 \pm 0.6	27.5 \pm 0.6	29.8 \pm 0.6	33.8 \pm 0.6	37.7 \pm 0.6	42.6 \pm 0.7
CIFAR-10	81.0 \pm 0.5	80.4 \pm 0.5	79.3 \pm 0.5	79.8 \pm 0.5	82.5 \pm 0.4	83.9 \pm 0.4	85.6 \pm 0.4	86.8 \pm 0.4
CIFAR-100	58.1 \pm 0.6	57.4 \pm 0.6	56.4 \pm 0.5	56.4 \pm 0.6	59.7 \pm 0.6	63.2 \pm 0.5	64.8 \pm 0.5	67.8 \pm 0.6
DTD	62.8 \pm 1.3	63.9 \pm 1.2	64.5 \pm 1.2	64.3 \pm 1.3	63.7 \pm 1.3	67.6 \pm 1.3	70.3 \pm 1.3	74.7 \pm 1.2
Flowers	80.8 \pm 0.6	80.3 \pm 0.6	80.2 \pm 0.6	79.4 \pm 0.6	76.5 \pm 0.6	80.8 \pm 0.6	85.0 \pm 0.5	88.8 \pm 0.5
Food	57.6 \pm 0.3	58.3 \pm 0.4	57.0 \pm 0.4	56.7 \pm 0.4	56.6 \pm 0.4	59.4 \pm 0.4	62.7 \pm 0.3	68.1 \pm 0.3
Pets	58.2 \pm 0.9	57.9 \pm 1.0	56.8 \pm 0.9	55.9 \pm 0.9	49.7 \pm 1.0	53.5 \pm 0.9	60.2 \pm 0.9	65.2 \pm 0.9
SUN	49.4 \pm 0.4	49.8 \pm 0.4	49.7 \pm 0.4	49.6 \pm 0.4	45.9 \pm 0.4	50.9 \pm 0.4	55.3 \pm 0.4	61.8 \pm 0.4
μ_{Tx}	55.9 \pm 0.2	55.3 \pm 0.2	55.1 \pm 0.2	54.5 \pm 0.2	53.1 \pm 0.2	57.0 \pm 0.2	60.7 \pm 0.2	65.3 \pm 0.2

Table 7: Transfer performance of SimCLR and CLIP models after pre-training on CC subsets.

Model Dataset size	SimCLR				CLIP			
	100K	200K	500K	2M	100K	200K	500K	2M
Aircraft	39.5 \pm 0.9	39.3 \pm 1.0	38.0 \pm 0.9	36.3 \pm 0.9	17.0 \pm 0.7	21.2 \pm 0.8	41.5 \pm 0.9	43.0 \pm 0.9
Birdsnap	19.2 \pm 0.5	18.9 \pm 0.5	19.7 \pm 0.5	19.0 \pm 0.5	8.3 \pm 0.4	10.4 \pm 0.4	19.8 \pm 0.5	26.2 \pm 0.6
Cal101	71.0 \pm 0.7	71.1 \pm 0.7	70.3 \pm 0.7	68.4 \pm 0.7	42.7 \pm 0.7	51.4 \pm 0.7	75.2 \pm 0.7	82.1 \pm 0.6
Cal256	56.9 \pm 0.5	58.5 \pm 0.5	58.6 \pm 0.5	57.7 \pm 0.5	32.9 \pm 0.4	38.2 \pm 0.5	62.4 \pm 0.5	70.5 \pm 0.4
Cars	33.1 \pm 0.6	29.8 \pm 0.6	28.1 \pm 0.6	26.8 \pm 0.6	11.8 \pm 0.4	15.5 \pm 0.5	36.1 \pm 0.7	37.4 \pm 0.6
CIFAR-10	80.4 \pm 0.5	80.6 \pm 0.4	80.2 \pm 0.5	79.7 \pm 0.5	71.1 \pm 0.5	72.9 \pm 0.5	83.5 \pm 0.4	86.0 \pm 0.4
CIFAR-100	56.8 \pm 0.5	58.2 \pm 0.5	56.8 \pm 0.5	57.2 \pm 0.6	46.6 \pm 0.6	47.7 \pm 0.6	62.3 \pm 0.6	66.2 \pm 0.5
DTD	64.8 \pm 1.2	67.0 \pm 1.2	67.3 \pm 1.2	67.0 \pm 1.3	41.9 \pm 1.3	49.9 \pm 1.3	69.1 \pm 1.2	74.3 \pm 1.1
Flowers	80.9 \pm 0.6	81.2 \pm 0.6	80.5 \pm 0.6	80.5 \pm 0.6	47.6 \pm 0.7	54.9 \pm 0.8	83.4 \pm 0.5	89.4 \pm 0.4
Food	57.4 \pm 0.4	57.9 \pm 0.4	56.9 \pm 0.4	57.4 \pm 0.4	36.6 \pm 0.4	43.0 \pm 0.3	61.7 \pm 0.3	67.4 \pm 0.4
Pets	54.8 \pm 1.0	55.4 \pm 1.0	55.6 \pm 0.9	55.6 \pm 0.9	30.4 \pm 0.9	34.0 \pm 0.9	55.7 \pm 1.0	61.9 \pm 0.9
SUN	51.4 \pm 0.4	52.9 \pm 0.4	53.1 \pm 0.4	53.2 \pm 0.4	29.2 \pm 0.4	34.7 \pm 0.4	54.6 \pm 0.4	62.8 \pm 0.4
μ_{Tx}	55.5 \pm 0.2	55.9 \pm 0.2	55.4 \pm 0.2	54.9 \pm 0.2	34.7 \pm 0.2	39.5 \pm 0.2	58.8 \pm 0.2	63.9 \pm 0.2

Table 8: Transfer performance of SimCLR and CLIP models after pre-training on YFCC subsets.

Model	Captions	Aircraft	Birdsnap	Ctech101	Ctech256	Cars	CIFAR10	CIFAR100	DTD	Flowers	Food-101	Pets	SUN937
CLIP	quadrant	29.9	11.3	64.0	46.3	21.0	78.6	53.5	56.4	61.4	47.8	39.4	39.9
CLIP _S (10)	quadrant	42.8	16.9	79.8	59.2	38.5	81.0	57.4	62.8	77.6	56.0	54.5	50.0
CLIP	full	42.1	15.9	75.9	57.9	36.2	82.5	59.4	62.2	76.7	56.4	52.3	48.8
CLIP _S (10)	full	44.2	18.2	82.4	62.5	39.4	83.0	59.7	65.4	81.2	60.3	56.1	54.6

Table 9: Linear probe accuracy for CLIP models trained on VisualGenome (Krishna et al., 2017).

Method	Size	Epochs	Aircraft	Birdsnap	Ctech101	Cars	CIFAR10	CIFAR100	DTD	Flowers	Food-101	Pets	SUN937
BYOL ((Tian et al., 2021))	100M	1000	47.5	31.3	84.0	44.3	85.0	63.9	75.2	93.4	67.9	71.1	63.4
MoCLR ((Tian et al., 2021))	100M	1000	45.6	29.4	85.6	41.1	87.8	69.9	75.8	92.9	67.7	67.7	63.4
CLIP	2M	100	43.0	26.2	82.1	37.4	86.0	66.2	74.3	89.4	67.4	61.9	62.8

Table 10: Comparison of our results to (Tian et al., 2021).

Model	CLIP				CLIP _S	
Dataset	CC		YFCC		CC	YFCC
Dataset size	100K	500K	100K	500K	100K	100K
Aircraft	35.5 ± 1.0	41.6 ± 1.0	35.4 ± 0.9	42.8 ± 0.9	40.1 ± 1.0	41.3 ± 0.9
Birdsnap	15.1 ± 0.5	19.8 ± 0.5	15.9 ± 0.5	20.7 ± 0.6	17.8 ± 0.5	19.2 ± 0.5
Cal101	67.7 ± 0.8	79.1 ± 0.6	67.9 ± 0.7	79.7 ± 0.6	75.1 ± 0.6	75.8 ± 0.6
Cal256	54.4 ± 0.5	65.9 ± 0.5	55.8 ± 0.5	67.6 ± 0.4	61.8 ± 0.5	62.6 ± 0.5
Cars	29.8 ± 0.6	37.7 ± 0.6	29.6 ± 0.6	37.8 ± 0.6	37.3 ± 0.6	38.1 ± 0.6
CIFAR-10	82.5 ± 0.5	85.6 ± 0.4	82.9 ± 0.4	85.6 ± 0.4	83.6 ± 0.4	82.7 ± 0.4
CIFAR-100	59.7 ± 0.6	64.8 ± 0.5	60.9 ± 0.6	65.2 ± 0.5	62.2 ± 0.6	60.9 ± 0.6
DTD	63.7 ± 1.2	70.2 ± 1.2	64.1 ± 1.3	71.0 ± 1.3	67.7 ± 1.3	68.7 ± 1.2
Flowers	76.5 ± 0.6	85.0 ± 0.5	77.7 ± 0.6	86.2 ± 0.5	81.4 ± 0.6	83.7 ± 0.6
Food	56.6 ± 0.4	62.7 ± 0.3	57.3 ± 0.4	64.0 ± 0.3	59.6 ± 0.4	61.3 ± 0.3
Pets	49.7 ± 1.0	60.2 ± 0.9	49.6 ± 0.9	61.6 ± 0.9	54.7 ± 0.9	56.6 ± 0.9
SUN	45.9 ± 0.4	55.3 ± 0.4	47.7 ± 0.4	57.1 ± 0.4	52.5 ± 0.4	54.9 ± 0.4
μ_{T_x}	53.7 ± 0.2	60.7 ± 0.2	54.8 ± 0.2	61.8 ± 0.2	57.8 ± 0.2	58.8 ± 0.2

Table 11: Effect of using BLIP captions for CC/YFCC images in CLIP training.

Dataset	CC	YFCC
Dataset size	100K	500K
Aircraft	37.0 ± 1.0	41.0 ± 1.0
Birdsnap	15.5 ± 0.5	21.1 ± 0.5
Cal101	71.1 ± 0.7	78.2 ± 0.7
Cal256	55.9 ± 0.5	64.9 ± 0.4
Cars	30.9 ± 0.6	35.2 ± 0.6
CIFAR-10	82.9 ± 0.5	85.1 ± 0.4
CIFAR-100	59.3 ± 0.6	63.4 ± 0.6
DTD	63.8 ± 1.3	71.8 ± 1.2
Flowers	76.3 ± 0.7	84.3 ± 0.5
Food	57.4 ± 0.4	64.1 ± 0.3
Pets	52.7 ± 0.9	59.3 ± 0.9
SUN	47.4 ± 0.4	56.4 ± 0.4
μ_{Tx}	54.2 ± 0.2	60.4 ± 0.2

Table 12: Effect of caption filtering on CLIP’s transfer performance.

Dataset	CC	COCO
Dataset size	200K	120K
Aircraft	41.9 ± 0.9	44.7 ± 1.0
Birdsnap	18.8 ± 0.5	18.6 ± 0.5
Cal101	77.4 ± 0.7	75.9 ± 0.6
Cal256	63.5 ± 0.4	62.8 ± 0.4
Cars	38.2 ± 0.6	40.8 ± 0.6
CIFAR-10	84.0 ± 0.4	84.1 ± 0.4
CIFAR-100	62.5 ± 0.6	61.3 ± 0.6
DTD	68.1 ± 1.2	65.3 ± 1.3
Flowers	82.4 ± 0.6	81.9 ± 0.6
Food	60.4 ± 0.4	62.0 ± 0.4
Pets	56.0 ± 1.0	59.6 ± 1.0
SUN	53.1 ± 0.4	51.9 ± 0.4
μ_{Tx}	58.8 ± 0.3	58.9 ± 0.3

Table 13: Training CLIP_S models using additional captions generated via GPT-J paraphrasing.

B.1 DESCRIPTIVE VS. NOISY CAPTIONS

In Section 3.2, we study the effect of the average descriptiveness of dataset captions on CLIP’s transfer performance. We find that the CC and YFCC datasets tend to have captions with lower descriptiveness than the manually-sourced COCO captions. A natural question to ask is whether these captions are just irrelevant noise. After all, they have been collected via automated scraping with little or no post-processing. In prior work, Alikhani et al. (2020) provide a taxonomy of relevant, non-noisy captions: Visible (“presents information that is intended to recognizably characterize what is depicted in the image”), Subjective (“describes the speaker’s reaction to, or evaluation of, what is depicted in the image”), Action (“describes an extended, dynamic process of which the moment captured in the image is a representative snapshot”), Story (“providing a free-standing description of the circumstances depicted in the image”) and Meta (“allows the reader to draw inferences not just about the scene depicted in the image but about the production and presentation of the image itself”). A noisy or irrelevant caption would be one that does not fall into the aforementioned categories.

They then recruit expert annotators to categorize CC images based on this taxonomy. It turns out that CC captions are actually well-aligned with the corresponding images: only 3% of the captions are irrelevant noise. The remainder of the captions are relevant, but might not always fall into the “Visible” category (their analogue to our notion of “descriptive”). To demonstrate that the same is holds for YFCC, we manually annotate random dataset samples using this taxonomy—see Figure 12.

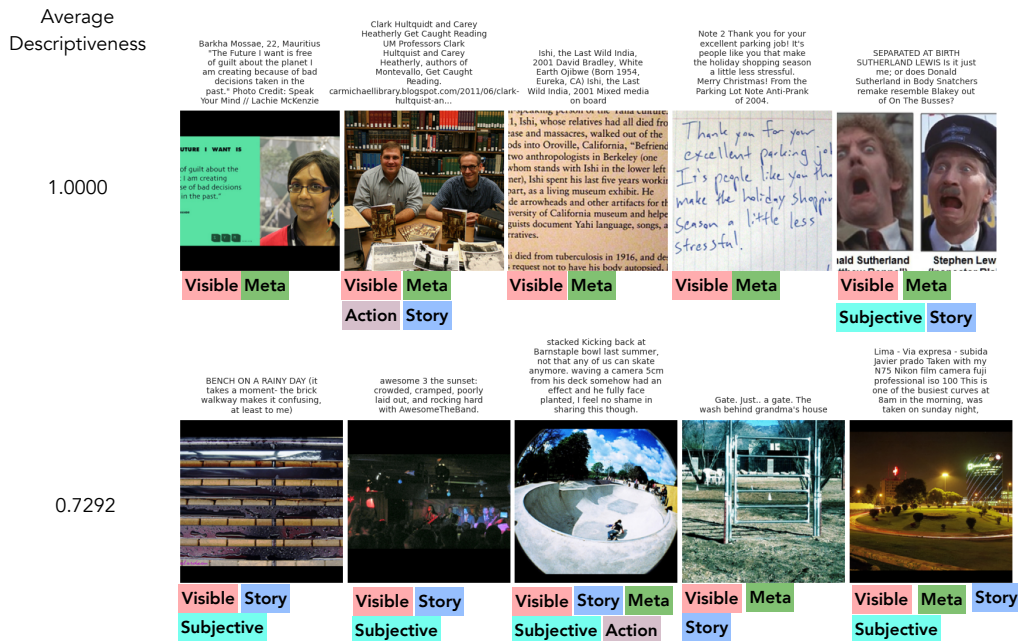


Figure 12: Random samples from the YFCC dataset with different average descriptiveness levels. We manually categorize the corresponding captions into the taxonomy of (Alikhani et al., 2020).