Data Collection and Preparation

The data we need to collect has to be both unimodal and multimodal. The requirement for multimodal data is obvious: We aim to align image and text, which requires a dataset of image-text pairs. Unimodal data is required for preliminary tests of classic unimodal knowledge distillation, on which we can then build. Further, we will utilize unimodal data for the evaluation of multimodal models on downstream tasks. After all, a multimodal model should not only excel in aligning modalities, but also in tasks that only involve on of the aligned modalities. This also gives us the opportunity to compare the performance of unimodal and multimodal distilled models on the same tasks.

Unimodal Data

Collecting unimodal data does not pose an obstacle, as there are many highly curated and large datasets available. For image data, we select ImageNet-1K [1], which is an intuitive choice, as is features a high variety of content, is widely used for image classification, and, with 1.2 million training images, can be considered a medium-sized dataset. For comparison, current (August 2024) SOTA vision-language models have been trained on datasets spanning at least 14 million samples [2]–[4].

We will use this dataset for both knowledge distillation, and, most importantly, for the evaluation of image models using the ImageNet-1K validation accuracy metric, which is the most popular benchmark for computer vision models by far. We utilize the full dataset of the 2012 version, which contains 1.2 million images for training and 50,000 for validation [1]. The data can be downloaded from Huggingface's dataset hub[1] without any costs, merely requiring an account.

For raw text data, used in unimodal knowledge distillation of our text model, we select OpenWebText (OWT) [5]. This data was developed to replicate the datasets used to train GPT-2, and is also publicly available on HuggingFace[2]. The dataset consists of raw unstructured text, without any labels, which are not necessary for our distillation process. It is published as 21 chunks, and we select the first 6 for training and the 7th for validation, which is around 33% of the data. We do not collect the full dataset, as the training data, when slicing it into sections of 192 tokens, which each slice being one training example, already consists of more than 2.5 billion tokens, which we consider sufficient for our purposes. Even though the data is already preprocessed and cleaned, we further preprocess it by removing empty lines and null bytes, which we found to be quite common and lead to problems during encoding and training, as they provide no learnable information.

For benchmarking language models, including our multimodal models, on downstream tasks, we will use the GLUE benchmark [6]. GLUE, short for **G**eneral **L**anguage **U**nderstanding **E**valuation, is a collection of NLP datasets spanning four different tasks: Sentiment analysis (SST-2), grammar error detection (CoLA), sentence similarity (STS-B, MRPC, QQP), and natural language understanding (QNLI, MNLI, RTE). All 8 datasets are publicly available, and can also be accessed through HuggingFace[3].

The 8 datasets measure the performance of language models on the following tasks:

SST-2

Sentence classification of rotten tomatoes movie reviews into "negative" (1), "somewhat negative" (2), "somewhat positive" (3), and "positive" (4) [7].

[1] https://huggingface.co/datasets/ILSVRC/imagenet-1k
[2] https://huggingface.co/datasets/Skylion007/openwebtext
[3] https://huggingface.co/datasets/nyu-mll/glue

CoLA
Is a binary classification tasks to test a models understanding of grammar: Model should output whether a sentence is grammatically correct (label: “acceptable” -> 1) or not (label: “unacceptable” -> 0) [8].

STS-B
A regression task. The model is tasked with predicting the similarity between two sentences. The similarity score is in the interval $[0, 5] \subset \mathbb{R}$ [9].

MRPC
Is a binary classification taks. The training objective is paraphrase detection, meaning whether two sentences describe the same semantic concept [10].

QQP
The same as MRPC, instead of a simple sentence pair, the goal is to detect wethere two questions are semantic duplicates, i.e. ask the same thing[4].

QNLI
A binary classification task, where the model has to predict whether one sentence is the answer to a question represented by another sentence. Examples are of the form (question, sentence) [11], [6].

RTE
A dataset of text pairs, where the model has to predict whether a hypothesis (sentence 2) can be inferred from a text (sentence 1) [12]–[15], [6]. The task is binary classification (hypothesis can be inferred, or not).

MNLI
A classification task, where the model has to predict whether a hypothesis can be inferred from the premise (entailment), contradicts the premise (contradiction), or is neutral (neutral). There a two versions available, MNLI matched and MNLI mismatched. Both consists of the same training dataset, but test set of MNLI mismatched consists of out-of-domain data, so sentence pairs about concepts not seen during training. It is therefore a better measure of generalization, compared to MNLI matched [16].

Concrete examples can be found in (TODO: cite glue examples) in the Appendix.

Dataset	Training Examples
ImageNet-1K [1]	1.28M
OpenWebText (subset) [5]	13M
GLUE [6]	990K (total)
Total	15.27M

Table 1: Unimodal datasets and their sizes used in this work. While the amount of training examples from OpenWebText is indeed correct, it is important to note that collecting text data is significantly cheaper to obtain, and requires less disk space than image data, which is why we were able to collect that much text data without any problems.

Multimodal Data
For training multimodal models, specifically image-text models, datasets containing image-text pairs are required. We orient ourselves on the BEiTv3 paper, currently achieving SOTA performance on

[4]https://quoradata.quora.com/First-Quora-Dataset-Release-Question-Pairs

multimodal benchmarks [3]. The paper uses the datasets COCO [17], Visual Genome [18], Conceptual Captions 3M [19] and 12M [20], and SBU captions [21], of which we select COCO, being the most popular and widely used dataset and therefore an intuitive choice, and a subset of both Conceptual Captions 3M and 12M.

While the COCO dataset can be downloaded in its entirety from the COCO website[5], both variants of Conceptual Captions, developed by Google, only provide urls and the caption for each image. This is because the images used come from a variety of sources on the internet, and have been uploaded by humans all over the world. The images stem from blog posts, news articles, social media, and other sources. Since Google does not own the rights to the images, they cannot provide them in a dedicated dataset, which is why there is no guarantee that all images will be available at the time of download. Because of this, we have to utilize not only Conceptual Captions 3M, but also the 12M variant to collect enough data. A favorable side effect this has is that our approach becomes more scalable due to the uncurated nature of CC12M [19], [20], which we will elaborate on in the next section. The index of CC3M is available on the official Conceptual Captions website (training split)[6], and the index of CC12M can be found in the corresponding GitHub repository[7].

Another popular choice for image-text pairs is the Visual Genome dataset [18], containing high quality images and detailed annotations. However, we refrain from using this dataset, as we experienced unstable training during preliminary tests, a circumstance we will address again in the experimental part of this work.

In total, we collect more than **650 GB** of data.

Dataset	Avg. Caption Length	$\frac{\#\text{ Captions}}{\#\text{ Images}}$	# Images	# Image-Text Pairs
COCO [17]	11.0	5.0	82,783	566,747
CC3M (subset) [19]	12.0	1.0	1,516,133	1,516,133
CC12M (subset) [20]	10.3	1.0	1,181,988	1,181,988
Total	-	-	2,780,904	3,264,868

Table 2: Multimodal Dataset used for aligning image and text.

On Curated Datasets

The goal of this work is to develop a multimodal model that is cheap to train and does not rely on labeled data in the end-to-end process. That means not only should our multimodal model not require labeled data for training, but also any pretrained models and components used in the process.

Whether image-text datasets, and, in fact, any multimodal dataset consisting of pairs of data, can be seen as curated or even labeled data is a matter of perspective. The difference between curated and labeled data lies in the purpose and level of human involvement: curated data focuses on the careful selection, organization, and cleaning of data to ensure quality and relevance, while labeled data involves explicit tagging or annotation of each example to provide a ground truth for training supervised models (which implies that the data is curated as well).

While image-text datasets are not labeled in the traditional sense, as in having a label for an image or text, the pairs themselves can be seen as labels. Single images or texts can be considered as in-the-wild data, i.e. data that appears naturally in the real world, like in books, articles, or on the internet, image-text pairs however require image and text to be paired together. This can be seen as less

[5] https://cocodataset.org/#download

[6] https://ai.google.com/research/ConceptualCaptions/download

[7] https://github.com/google-research-datasets/conceptual-12m

natural, as it requires a human to create the caption for an image, or vice versa, which is a form of labeling. The COCO dataset, for example, can be seen as labeled, as for each image a human created a caption with the specific intention of training Machine Learning models [17]. Consequently, whether multimodal learning can be seen as self-supervised learning, as it is often referred to in the literature [2]–[4], is debatable. With this in mind, creating a multimodal model that is scalable in the sense that it does not rely on labeled data, which is one of the most challenging aspects of AI research, is, if multimodal data is seen as labeled data, not possible.

However, there are multimodal data sources that are at the very least uncurated. One example is the alt-text of images on the internet. Even though the alt-text is created by humans, it is not created with the intention of creating data for Machine Learning, but rather to provide a description of the image for visually impaired people. Consequently, the data was generated naturally as a byproduct of a different task, and we therefore refer to any uncurated dataset as unlabeled data in this work.

This is exactly why we select both CC3M and CC12M, as they, especially CC12M, consists of in-the-wild image-text pairs from the internet [19], [20]. This way of collecting data and training models is therefore significantly more scalable than using curated datasets specifically created for Machine Learning, and ensures that our approach to multimodal models can be applied to a wide range of tasks and domains without any explicit human intervention. A comparison between curated and labeled samples, and in-the-wild samples can be seen in Figure 1 below.

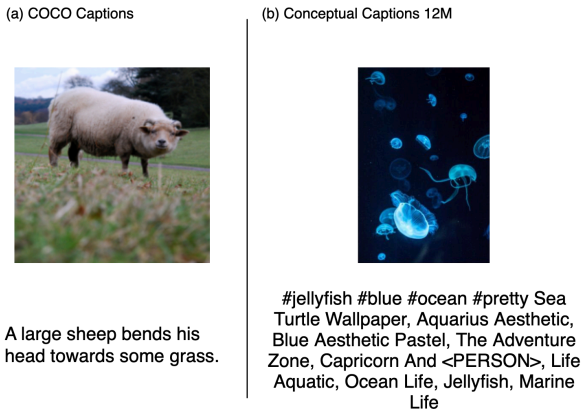

Figure 1: Side-by-side comparison of examples seen in COCO (a) and CC12M (b). While COCO features high quality images and detailed annotations, CC12M consists of in-the-wild image-text pairs from the internet. The latter enables scalability, as more data can be collected without the need for human annotation. The caveat is that the quality of the data is not guaranteed, and image-text pairs might be less correlated. Images and text in the figure have been taken from the COCO train set [17] and CC12M [20], respectively.

Data Persistence

How to organize, store, and batch the data during training is an important aspect, as storing this much data is not trivial, and we need to ensure an efficient data pipeline to avoid bottlenecks during training.

For all datasets, except OpenWebText, we organize the data in a format inspired by the authors of BEiT-3. In the file system, each dataset is stored in a separate folder. Each split of a dataset is stored in a jsonl file (inside the respective dataset folder), containing a list of python-parseable dictionaries, with each dictionary representing one (train/val/test) example. Each dictionary contains a sample id, which is especially important for e.g. image-text retrieval, as the sample id is used to check if an image-text pair is a positive (they have the same id) or negative pair. Further, each dictionary contains a key for the image path in the file system, or a key for the text. If the dataset is

multimodal, each sample, and therefore each dictionary, contains both an image path key and a text key. The text, already splitted into a list of tokens, is directly stored in the dictionary, as it is small in size.

All datasets containing images have a separate folder for each split, which contains the raw images in png format.

During training, we load the whole jsonl file into memory, which is manageable, as they rarely exceed 1 GB in size. For each batch of size B, B elements are drawn from the list of the jsonl file, where each element is the dictionary representing one example.

If the dataset contains images, the images are loaded from the file system using the image path in the dictionary, and then resized to a fixed size of 224x224 pixels, which is the size for images we use throughout this work. When appropriate, data augmentation is applied. Since all images are of the same size, they can be stacked into a tensor of shape $B \times 3 \times 224 \times 224$.

If the dataset contains text, which is a list of tokens already, each token of the text is converted to its corresponding token id, and the resulting list of token ids is prepended with the id of the [`T_CLS`] token, and appended with the id of the [`T_SEP`] token. The text is then padded to a fixed number of tokens, using the id of the [`T_PAD`] token. To which fixed length the text is padded depends on the model and approach, and will be specified in the respective section. After padding, all texts are stacked into a tensor of shape $B \times T$, where T is the fixed number of tokens.

Since sample ids are simple integers, they can be stacked into a tensor of shape $B \times 1$ without any further processing.

After that, the batch is ready to be fed into the model.

For OpenWebText we take a different approach. The data is stored in a single binary file for each split, which contains the tokenized text data. We choose this strategy for OpenWebText, as the dataset consists of raw unstructured text, from which training examples can easily be created during training by loading the whole dataset into memory, and slicing the text, into consecutive slices of fixed length. Padding is not necessary, as the slices are already of the same length. Each token is then converted to its corresponding token id, and the resulting list of token ids is prepended with the id of the [`T_CLS`] token, and appended with the id of the [`T_SEP`] token. Consequently, for a maximum sequence length of T tokens, the whole dataset, which is one long list of tokens, is iterated over during training in steps of $B * (T - 2)$, where B is the batch size. We do not take $B * T$, as for each example the [`T_CLS`] and [`T_SEP`] token are added, which increases the length of each example by 2 tokens. After a slice has been taken from the dataset, was converted to token ids, and special tokens were added, it is stacked into a tensor of shape $B \times T$, and then fed into the model.

The different data formats used in this work are compared in Figure 2 below.

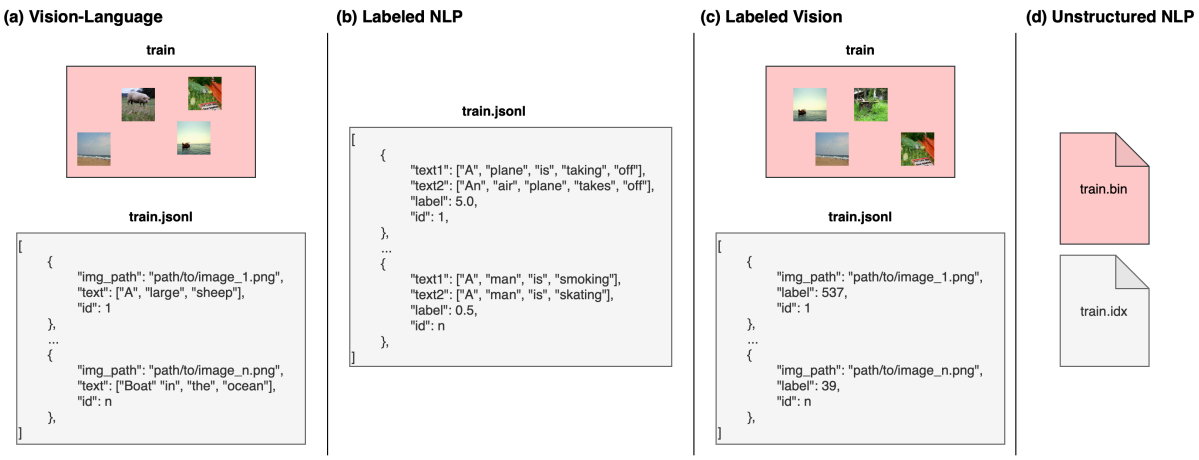

Figure 2: Comparison of different dataset organization and storage formats used in this work. Labeled NLP (d) datasets are not binarized, as they have a structure, i.e. there are fixed examples with labels.

Dataset	Example	Label
CoLA	Our friends won't buy this analysis, let alone the next one we propose.	1
SST-2	hide new secretions from the parental units	0
MRPC	Amrozi accused his brother, whom he called "the witness", of deliberately distorting his evidence. [SEP] Referring to him as only "the witness", Amrozi accused his brother of deliberately distorting his evidence.	1
STS-B	A plane is taking off. [SEP] An air plane is taking off.	5.0
QQP	How is the life of a math student? Could you describe your own experiences? [SEP] Which level of prepration is enough for the exam jlpt5?	0
MNLI	Conceptually cream skimming has two basic dimensions - product and geography. [SEP] Product and geography are what make cream skimming work.	1
QNLI	When did the third Digimon series begin? [SEP] Unlike the two seasons before it and most of the seasons that followed, Digimon Tamers takes a darker and more realistic approach to its story featuring Digimon who do not reincarnate after their deaths and more complex character development in the original Japanese.	1
RTE	No Weapons of Mass Destruction Found in Iraq Yet. [SEP] Weapons of Mass Destruction Found in Iraq.	1

Table 3: Training examples of the GLUE benchmark tasks (one example per task).

Bibliography

[1] O. Russakovsky *et al.*, "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision (IJCV)*, vol. 115, no. 3, pp. 211–252, 2015, doi: 10.1007/s11263-015-0816-y.

[2] H. Bao *et al.*, "VLMo: Unified Vision-Language Pre-Training with Mixture-of-Modality-Experts," in *Advances in Neural Information Processing Systems*, 2022. [Online]. Available: https://openreview.net/forum?id=bydKs84JEyw

[3] W. Wang *et al.*, "Image as a Foreign Language: BEIT Pretraining for Vision and Vision-Language Tasks," in *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023, pp. 19175–19186. doi: 10.1109/CVPR52729.2023.01838.

[4] A. Singh *et al.*, "FLAVA: A foundational language and vision alignment model," *CoRR*, 2021, [Online]. Available: https://arxiv.org/abs/2112.04482

[5] A. Gokaslan and V. Cohen, "OpenWebText Corpus." 2019.

[6] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman, "GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding," 2019.

[7] R. Socher *et al.*, "Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank," in *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, Seattle, Washington, USA: Association for Computational Linguistics, Oct. 2013, pp. 1631–1642. [Online]. Available: https://www.aclweb.org/anthology/D13-1170

[8] A. Warstadt, A. Singh, and S. R. Bowman, "Neural Network Acceptability Judgments," *arXiv preprint arXiv:1805.12471*, 2018.

[9] P. May, "Machine translated multilingual STS benchmark dataset.," 2021. [Online]. Available: https://github.com/PhilipMay/stsb-multi-mt

[10] W. B. Dolan and C. Brockett, "Automatically constructing a corpus of sentential paraphrases," in *Proceedings of the International Workshop on Paraphrasing*, 2005.

[11] P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang, "SQuAD: 100,000+ Questions for Machine Comprehension of Text," in *Proceedings of EMNLP*, Austin, Texas: Association for Computational Linguistics, 2016, pp. 2383–2392.

[12] I. Dagan, O. Glickman, and B. Magnini, "The PASCAL recognising textual entailment challenge," *Machine learning challenges. evaluating predictive uncertainty, visual object classification, and recognising tectual entailment*. Springer, pp. 177–190, 2006.

[13] R. Bar Haim *et al.*, "The second PASCAL recognising textual entailment challenge," 2006.

[14] D. Giampiccolo, B. Magnini, I. Dagan, and B. Dolan, "The third PASCAL recognizing textual entailment challenge," in *Proceedings of the ACL-PASCAL workshop on textual entailment and paraphrasing*, 2007, pp. 1–9.

[15] L. Bentivogli, I. Dagan, H. T. Dang, D. Giampiccolo, and B. Magnini, "The Fifth PASCAL Recognizing Textual Entailment Challenge," 2009.

[16] A. Williams, N. Nangia, and S. R. Bowman, "A Broad-Coverage Challenge Corpus for Sentence Understanding through Inference," in *Proceedings of NAACL-HLT*, 2018.

[17] T.-Y. Lin *et al.*, "Microsoft COCO: Common Objects in Context," in *Computer Vision - ECCV 2014 - 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V*, D. J. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds., in Lecture Notes in Computer Science, vol. 8693. Springer, 2014, pp. 740–755.

[18] R. Krishna *et al.*, "Visual Genome: Connecting Language and Vision Using Crowdsourced Dense Image Annotations," *Int. J. Comput. Vision*, vol. 123, no. 1, pp. 32–73, May 2017.

[19] P. Sharma, N. Ding, S. Goodman, and R. 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, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, I. Gurevych and Y. Miyao, Eds., 2018, pp. 2556–2565.

[20] S. Changpinyo, P. Sharma, N. Ding, and R. Soricut, “Conceptual 12M: Pushing Web-Scale Image-Text Pre-Training To Recognize Long-Tail Visual Concepts,” in *CVPR*, 2021.

[21] V. Ordonez, G. Kulkarni, and T. L. Berg, “Im2text: Describing images using 1 million captioned photographs,” in *Advances in Neural Information Processing Systems 24: 25th Annual Conference on Neural Information Processing Systems 2011. Proceedings of a meeting held 12-14 December 2011, Granada, Spain*, J. Shawe-Taylor, R. S. Zemel, P. L. Bartlett, F. C. N. Pereira, and K. Q. Weinberger, Eds., 2011, pp. 1143–1151.