Can we train vision and language zero-shot classification models without syntax?

Ajinkya Tejankar [1*] **Maziar Sanjabi** [2] **Bichen Wu** [2] **Madian Khabsa** [2]
Saining Xie [2] **Hamed Pirsiavash** [1] **Hamed Firooz** [2]
[1] UC Davis [2] Meta AI

Abstract

Natural language supervision in the form of image captions was recently shown to be an effective way of training zero-shot image classification models. In this work, we focus on teasing out what parts of the language supervision are essential for training zero-shot models. Through extensive and careful experiments, we show that replacing intact captions with Bag-of-Words (BoW) does not significantly degrade the zero-shot performance. Surprisingly, we can even slightly improve the performance on some datasets by balancing the frequency of words in BoW.

1 Introduction

Recently, CLIP [1] showed impressive results on zero-shot learning, demonstrating the ability to generalize to unseen image classification tasks. CLIP achieves this by pre-training of vision and text encoders with cross-modal contrastive learning on a large-scale dataset scraped from the internet. Unlike supervised learning datasets that require human annotations, image-text data are easier to obtain [1] and contain richer information about images than categorical labels. As shown in [2], this rich information can provide better supervision and lead to improved generalization as compared to categorical labels. Further, cross-modal contrastive learning connects images with language which makes them useful out-of-the box for zero-shot learning, unlike unimodal contrastive learning.

However, it is not clear whether all the rich information present in textual supervision is necessary for training the model. Alternatively, is it possible to train a good model without all the rich information in textual supervision? Compare the two captions *"a man standing in front of a red fire truck"* and *"man red standing fire"*. The first one is a semantically rich signal that denotes multiple objects, their attributes, and different relationships among them. While, the second one is a sparse signal, similar to categorical labels, that simply enumerates few objects, attributes, and relationships. Is the additional information provided in the structure of the first sentence a requirement for CLIP-like models?

To answer the above question, we conduct experiments by deforming the text captions to change their language properties, and observe the impact on the trained model's zero-shot performance. For example, we use the word shuffling operations to break a sentence to a Bag-of-Words (BoW), we drop random words, etc. We observe that applying such transformations and converting the captions to BoW does not significantly degrade the model's zero-shot performance. Surprisingly, we can even marginally improve the model's accuracy on some datasets with a simple strategy of balancing the frequency of words. This further indicates that the model can improve performance without relying on the dense signal of intact captions. Overall, our results show that there is room for designing better vision and language models that can exploit the rich information in intact captions such that zero-shot performance is also improved.

*Work done while interning at Meta AI. Corresponding author (atejankar@ucdavis.edu).

3rd Self-Supervised Learning Theory and Practice Workshop at NeurIPS 2022.

2 Related Works

Our work is mainly related to vision and language pretraining, zero-shot learning, and text as supervision works which are described below. However, our key idea of text deformation is also studied in [3] and [4] for natural language understanding tasks.

Vision and Language Pre-training (VLP): Here, the goal is to learn generalizable, deeply fused, vision and language (V&L) features during pre-training and then fine-tune them for complex downstream V&L tasks [5, 6, 7, 8, 9, 10, 11]. Unlike works in this area, we do not do complex modality fusion, which requires parameter sharing between both modalities, and focus on zero-shot learning for evaluation. Our work is also related to [12] which creates a new dataset and task to understand complex compositional reasoning of VLP models.

Zero-Shot Learning (ZSL): Traditionally, zero-shot learning implies a model that can classify classes without seeing any examples (zero shot) during training [13]. However, CLIP [1] defines it as generalization to novel image recognition datasets where it shows impressive performance by learning representations from scratch with a large-scale (400M) dataset of image and caption pairs. Works building upon CLIP have scaled-up the pre-traininig dataset to noisier 1B data points [14], improved loss function [15], added unimodal SSL losses [16]. Our training and evaluation is similar to these works [1, 16, 14], but our goal is to shed more light on different aspects of text supervision.

Text as Supervision: Our work explores natural language sentences as a source of supervision. The work in [17] uses the objective of predicting the BoW tags associated with an image to train visual features. This work is extended in [18] to predict n-grams instead of individual words. VirTex [2] uses generative modeling of text as a data-efficient way of training visual features. We show that natural language sentences can be turned into BoW without significantly hurting performance.

3 Method

The primary question explored in this work is: what parts of language supervision are necessary for training vision models for zero-shot (ZS) classification? We first introduce the cross-modal contrastive pre-training framework used in all of our experiments. Then, we introduce various techniques of deforming the text, thereby converting the captions to BoW.

Cross-modal Contrastive Pre-training: Given a pair of image and caption, denoted as (x_i, y_i), we use image encoder f and text encoder g to extract a pair of feature embeddings as (u_i, v_i), where $u_i = f(x_i), v_i = g(y_i)$. The encoders f and g are initialized with self-supervised learning (SSL) models trained in respective modalities. Similar to [1] and [19], we use a cross-modal InfoNCE [20] loss. In order to increase the number of negative samples without increasing the batch size and training memory, we follow [21] and maintain a memory bank of negative samples. Embeddings in the memory bank are computed by an exponential moving average (EMA) copy of the models, which we denote as f' and g', and their outputs as u'_i and v'_i. Now we can write our training losses as:

$$L_i^{img} = -\log \frac{\exp(d(u_i, v'_i)/\tau)}{\sum_{j=1}^{K} \exp(d(u_i, v'_j)/\tau)} \quad L_i^{txt} = -\log \frac{\exp(d(v_i, u'_i)/\tau)}{\sum_{j=1}^{K} \exp(d(v_i, u'_j)/\tau)} \tag{1}$$

Where, K is the size of the memory bank, τ is the temperature parameter, and $d(\cdot, \cdot)$ is the cosine similarity function. The loss per sample is the equally weighted average of above two losses and over all samples in the batch.

3.1 From Intact Captions to Bag-of-Words

In order to understand the importance of natural language structure in captions, we design a set of operations to change/deform them. Each operation removes some information from the original caption. If the lost information was important for zero-shot learning, then the model would degrade. We use following operations in our experiments.

`Shuffle`: Shuffles the order of words in a given caption, thereby breaking its syntactic structure.

`RmStopNalpha`: Removes stop words and non-alphabetical words from a given caption. The goal is to remove words that are important for syntactical structure of natural language captions.

Figure 1: **Intact vs. BoW captions.** Illustration of a few intact captions and their BoW counterparts. The words are first shuffled to remove syntax by `Shuffle`. Stop words ("all", "were", "the", etc.) and non-alphabetical tokens (":" and ".") are removed with `RmStopNalpha`. Next, words like "cars", "table", "rocks" are removed by `RmTopFreq(1000)`. Note that the model still sees and learns about them but from the small set of captions that are kept intact. Further, the blue word "irresistibly" is removed by `LimitToBaseVocab`, and purple words like "utter" are removed by `Keep(n)`.

`LimitToBaseVocab`: Limits the words in a given caption to be from the vocabulary of a base set of captions. The goal is to restrict the diversity of words in captions. Hence, we randomly select a small subset of all captions, 10% in our experiments, to be the "base" captions. Then, we deform the 90% split such that any word not from the vocabulary of the base 10% split is removed.

`RmTopFreq(t)`: Removes top-`t` most frequent words. We calculate the frequency of each word as the number of captions from the base set in which it appears. Similar to `LimitToBaseVocab`, this deformation is only applied on the 90% set of captions. The goal is to balance the frequency of words by suppressing those that occur frequently.

`Keep(n)`: Keeps the first `n` words of a given caption. The goal is to simply remove information without discrimination. Note that except `Keep(n)`, the order in which other operations are applied does not matter. Hence, it is always applied in the end after all other operations.

These operations can be combined in different ways to increase or decrease the strength of deformation. The default deformation for constructing BoW captions is the following cascade of operations: `Shuffle + RmStopNalpha + LimitToBaseVocab + RmTopFreq(1000) + Keep(4)`. The resulting dataset is referred to as the "default BoW". See Figure 1 for a comparison of intact and BoW image caption pairs. Note that in some cases, captions may become empty if all their words are removed during deformation. In this case, the empty caption and the corresponding image is removed from the dataset which reduces the BoW dataset size. For instance the "default BoW", dataset is about 20% smaller than the original intact captions dataset.

4 Experiments

We use Conceptual Captions (CC) [22] with $\sim$ 2.9M image-caption pairs for training our models. We use following encoders: ResNet-50 trained on unlabeled ImageNet [23] with BYOL [24] as the image encoder and transformer models trained with DeCLUTR (sci-base) [25], BERT (base-uncased) [26], RoBERTa (base) [27], and DistilBERT (base-uncased) [28] as text encoders. We train for 20 epochs by default and the results are averaged over 4 runs with different seeds. For zero-shot evaluation, we follow the procedure in CLIP [1] but only use a single prompt, `'a photo of a {class name}'`, instead of prompt ensembling. We use following datasets for zero-shot evaluation: ImageNet [23], Food101 [29], SUN397 [30], CIFAR100 [31], Flowers102 [32], Pets [33], Caltech-101 [34] and DTD [35]. Further details and ablations can be found in the Appendix.

Intact Captions vs. Bag-of-Words: We progressively add stronger deformations and explore their effect on zero-shot ImageNet evaluation in Table 1. We find that shuffling the word order, removing stop/ non-alphabetical words, and only keeping 4 original words for 100% of the captions does not significantly hurt the model performance. Further, we show that performance can even be improved by keeping 10% captions intact and removing the top 1000 most frequent words from the rest 90% in

Table 1: **Effect of deformations:** Each subsequent row adds a new operation on top of the previous row with lesser indentation. "mostly BoW" refers to the setting where 90% of text is converted to BoW while in "only BoW" all 100% are BoW. Note that `Keep(n)` is always applied in the end after all other deformations. Also, note that the application of a deformation may lead to an empty caption in which case the image-caption pair is removed from the dataset. For instance, deformation in the last row results in $\approx 20\%$ reduction in dataset size. Despite this reduction, most deformations do not significantly hurt the performance. In fact, the last row even shows slight improvement.

Deformations	Avg. Cap Len	Zero-Shot on ImageNet	
		DeCLUTR	DistilBERT
Intact captions	10.3	27.9	29.5
only BoW			
`Shuffle`	10.3	25.9	26.9
`+ RmStopNalpha`	5.8	**27.8**	**28.5**
`+ Keep(1)`	1.0	16.7	18.3
`+ Keep(2)`	2.0	24.2	26.1
`+ Keep(4)`	3.8	27.3	27.9
mostly BoW			
`Shuffle`	10.3	25.7	28.2
`+ RmStopNalpha`	6.3	28.2	28.9
`+ Keep(4)`	4.5	27.8	29.1
`+ LimitToBaseVocab`	4.5	28.0	29.0
`+ RmTopFreq(1000)`	3.2	**29.1**	**30.1**

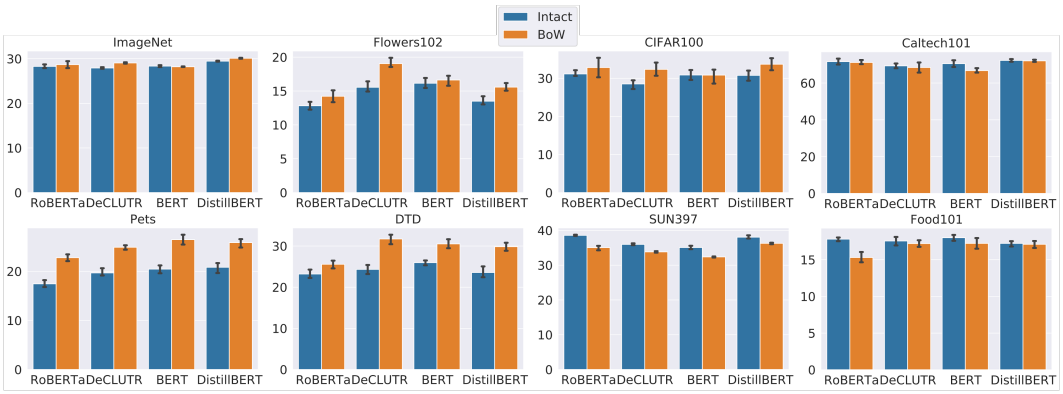

Figure 2: **Intact vs. BoW.** We compare intact and default BoW caption based models on zero-shot evaluation across 4 architectures and 8 datasets. BoW achieves comparable or higher accuracy on most settings resulting in a 7.1% relative improvement in accuracy averaged across all settings.

addition to other deformations. Next, we evaluate the model trained with "default BoW" captions on all 8 datasets in Figure 2. These datasets include fine-grained classification tasks like, food classification [29], flowers classification [32], etc. We observe that for some datasets BoW models are consistently better than their intact counterparts, while on others we observe small degradation. Overall, we find that the average relative change in accuracy is +7.1%. This shows that improvements outweigh the degradations when switching to BoW. See Appendix for more details and ablations.

5 Conclusion

We investigate what components of the natural language supervision in image-caption pairs are truly exploited by vision and language models for zero-shot evaluation. We design various operators to deform captions and convert them into Bag-of-Words. We find that such deformations do not hurt the model performance, and can even slightly improve it in some cases.

References

[1] 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. 1, 2, 3, 8, 9

[2] Karan Desai and Justin Johnson. Virtex: Learning visual representations from textual annotations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021. 1, 2

[3] Koustuv Sinha, Robin Jia, Dieuwke Hupkes, Joelle Pineau, Adina Williams, and Douwe Kiela. Masked language modeling and the distributional hypothesis: Order word matters pre-training for little. *arXiv preprint arXiv:2104.06644*, 2021. 2, 8

[4] Koustuv Sinha, Prasanna Parthasarathi, Joelle Pineau, and Adina Williams. UnNatural Language Inference. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 2021. 2

[5] Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. Visualbert: A simple and performant baseline for vision and language. *arXiv preprint arXiv:1908.03557*, 2019. 2

[6] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *Advances in Neural Information Processing Systems*, 2019. 2

[7] Junnan Li, Ramprasaath R. Selvaraju, Akhilesh Deepak Gotmare, Shafiq Joty, Caiming Xiong, and Steven Hoi. Align before fuse: Vision and language representation learning with momentum distillation. In *Advances in Neural Information Processing Systems*, 2021. 2

[8] Liunian Harold Li, Haoxuan You, Zhecan Wang, Alireza Zareian, Shih-Fu Chang, and Kai-Wei Chang. Unsupervised vision-and-language pre-training without parallel images and captions. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*, 2021. 2, 8

[9] Stella Frank, Emanuele Bugliarello, and Desmond Elliott. Vision-and-language or vision-for-language? On cross-modal influence in multimodal transformers. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2021. 2

[10] Jiasen Lu, Vedanuj Goswami, Marcus Rohrbach, Devi Parikh, and Stefan Lee. 12-in-1: Multi-task vision and language representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2

[11] Emanuele Bugliarello, Ryan Cotterell, Naoaki Okazaki, and Desmond Elliott. Multimodal pretraining unmasked: A meta-analysis and a unified framework of vision-and-language berts. *Transactions of the Association for Computational Linguistics*, 2021. 2

[12] Tristan Thrush, Ryan Jiang, Max Bartolo, Amanpreet Singh, Adina Williams, Douwe Kiela, and Candace Ross. Winoground: Probing vision and language models for visio-linguistic compositionality. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 2

[13] Yongqin Xian, Christoph H. Lampert, Bernt Schiele, and Zeynep Akata. Zero-shot learning—a comprehensive evaluation of the good, the bad and the ugly. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019. 2

[14] 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 *Proceedings of the 38th International Conference on Machine Learning (ICML)*, 2021. 2

[15] Ruizhe Cheng, Bichen Wu, Peizhao Zhang, Peter Vajda, and Joseph E Gonzalez. Data-efficient language-supervised zero-shot learning with self-distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021. 2

[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. In *International Conference on Learning Representations*, 2022. 2, 8, 9

[17] Armand Joulin, Laurens Van Der Maaten, Allan Jabri, and Nicolas Vasilache. Learning visual features from large weakly supervised data. In *European Conference on Computer Vision*, 2016. 2

[18] Ang Li, Allan Jabri, Armand Joulin, and Laurens van der Maaten. Learning visual n-grams from web data. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4183–4192, 2017. 2

[19] 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. 2

[20] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. 2

[21] 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 (CVPR)*, 2020. 2

[22] 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, 2018. 3, 12

[23] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 2015. 3, 12

[24] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, Bilal Piot, koray kavukcuoglu, Remi Munos, and Michal Valko. Bootstrap your own latent - a new approach to self-supervised learning. In *Advances in Neural Information Processing Systems*, 2020. 3, 9, 11

[25] John Giorgi, Osvald Nitski, Bo Wang, and Gary Bader. DeCLUTR: Deep contrastive learning for unsupervised textual representations. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 2021. 3

[26] 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)*, 2019. 3

[27] 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. 3

[28] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, 2019. 3

[29] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101 – mining discriminative components with random forests. In *European Conference on Computer Vision*, 2014. 3, 4

[30] Jianxiong Xiao, James Hays, Krista A. Ehinger, Aude Oliva, and Antonio Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *Computer Vision and Pattern Recognition*, 2010. 3

[31] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 3

[32] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *Indian Conference on Computer Vision, Graphics and Image Processing*, 2008. 3, 4

[33] O. M. Parkhi, A. Vedaldi, A. Zisserman, and C. V. Jawahar. Cats and dogs. In *Computer Vision and Pattern Recognition*, 2012. 3

[34] Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. *Computer Vision and Pattern Recognition Workshop*, 2004. 3

[35] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *Computer Vision and Pattern Recognition*, 2014. 3

[36] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, 2014. 8

[37] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *Proceedings of the 37th International Conference on Machine Learning*, 2020. 9

[38] Soroush Abbasi Koohpayegani, Ajinkya Tejankar, and Hamed Pirsiavash. Mean shift for self-supervised learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. 9

[39] Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V Le. Self-training with noisy student improves imagenet classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020. 11

Table 2: **BoW vs Intact captions on retrieval evaluation**. The task of caption retrieval is a harder task compared to zero-shot classification. As a result, the BoW model results in a significantly worse performance compared to its intact captions counterpart in the zero-shot setting. However, when the models are fine-tuned the gap between them is greatly reduced. These findings are similar to those of [3, 8] which show that input deformations during pre-training do not hurt the models when they are fine-tuned for downstream tasks. For Image to text retrieval with R@1, we see that the gap between Ours (BoW) and Ours (Intact) is about 6 points but when both of them are fine-tuned the gap reduces to about 1 point.

	MS-COCO					
	Image to Text			Text to Image		
	R@1	R@5	R@10	R@1	R@5	R@10
Zero-Shot						
Visual N-Grams	8.7	23.1	33.3	5.0	14.5	21.9
Ours (BoW)	17.7	39.5	52.0	13.7	33.2	45.5
Ours (Intact)	23.0	48.2	62.2	19.0	42.0	53.8
CLIP	58.4	81.5	88.1	37.8	62.4	72.2
Finetune						
Ours (BoW)	54.4	81.6	89.1	37.7	67.4	78.0
Ours (Intact)	55.8	82.5	89.8	38.9	68.3	78.8

A Appendix

Implementation Details: We use PyTorch for implementation with SGD optimizer (weight_decay = 0, momentum = 0.9), batch size = 512, and a cosine learning rate decay with initial LR = 0.003. The model is trained on 8 A100 GPUs which requires 18hrs for 20 epochs of full 2.9M CC dataset. The temperature is set to 0.02, and the default memory bank size is 8192 ($\sim$ 8k). We use a linear layer on top of the vision encoder to match the dimension of its embeddings with the text embeddings. Since the dimension of text encoder embeddings is typically 768, the linear layer is 2048×768 since we use ResNet-50 as the vision encoder. For the creating a single sentence level embedding from a list of tokens, we use average pooling. We also perform an ablation over using `[CLS]` pooling but we did not find any noticeable difference in performance. The comparison between average and `[CLS]` pooling methods can be found in Table 7.

BoW vs intact captions on retrieval evaluation: Despite being image-to-text retrieval at the core, zero-shot evaluation only tests the ability of a model to distinguish between a set of categories or nouns. This is easier compared to the task of cross-modal (image-to-text and text-to-image) retrieval on a dataset like MS-COCO [36] where the descriptions can be much more complex. Thus, it is reasonable to expect a model trained on BoW text descriptions to perform worse than its intact counterpart in the zero-shot retrieval setting. We indeed see this behaviour in Table 2 but, we also see that when both models are fine-tuned the gap between them shrinks greatly. This finding is consistent with U-VisualBERT [8] where a vision-and-language model without any paired data has comparable performance to the one that uses paired data for fine-tuning evaluation. Further, [3] shows that transformer based language models are robust to word order shuffle during pre-training for fine-tuning evaluation.

Intact captions baselines: Here, we show that our training setup is competitive with other SOTA methods given similar number of aligned images. We compare our model trained for 40 epochs with intact captions and DistilBERT architecture. The comparison in Table 3 shows that our model trained on only $\sim$ 3M aligned data points is comparable to the CLIP [1] model trained on 15M aligned data points. Moreover, our results are higher compared to those of DeCLIP [16] and CLIP (implementation from [16]) given same amount of aligned data. However, DeCLIP and CLIP models are initialized from scratch while ours are initialized from SSL models. We believe this difference is the reason for the better performance of our models.

Ablations: We explore freezing one of the encoders in Table 4. We find that even a frozen text encoder works better with BoW captions than intact ones despite being pre-trained on intact natural language sentences. We study the effect of removing different number of most frequent words from

Table 3: **Intact captions baselines**. We compare the results of our implementation with other works to show that our implementation is competitive. ^{b} refers to results of CLIP [1] method from [16].

Method	Dataset	IN1k-ZS
CLIP [1]	YFCC-15M	31.3
CLIP^b	CC-3M	20.6
DeCLIP^a [16]	CC-3M	27.2
Ours	CC-3M	30.3

Table 4: **Ablation of freezing different encoders.** In case of frozen text encoder, intact caption results are similar to BoW. This shows that despite being initialized from an already trained language model, presence or absence of syntax does not matter. Further, we find that freezing the image encoder hurts the model significantly less as compared to freezing the text model. This indicates that training text encoder is more important than the image encoder.

Input	IN-1k-ZS		
	Freeze		Train
	Text	Image	Both
Intact captions	15.1	26.7	29.5
BoW captions	16.9	26.4	30.1

the base vocabulary in Table 5. We also explore the effect of training hyperparameters like epochs, memory bank size, and text encoder architectures/initializations on an intact captions model in Table 6. We can see that memory bank size = 8k and epochs = 20 work well for all architectures, and longer training only helps marginally. Thus, we chose these as the default parameters for all experiments. We chose DistilBert and DeCLUTR for other experiments since they are respectively the best and the worst. In Table 7, we study the effect of `[CLS]` pooling vs avearge pooling for caption embeddings. While average pooling is the default in our experiments, we find that switching to `[CLS]` does not change the message of our work. Even with `[CLS]` pooling, BoW models are still better than their intact counterparts by similar margins.

Transfer linear evaluation: Similar to [1, 24, 37], we also evaluate the transferability of our trained models by training linear layers on top of frozen vision encoders. The training and evaluation procedure is exactly the same as MSF [38], and the results are reported in Table 8.

Detailed Zero-Shot: We provide detailed zero-shot with mean and standard deviations in Table 9.

Table 5: **Ablation for removing different number of most frequent words.** We explore different paramters for `RmTopFreq(t)` operation in the default deformation. We can see that removing too many frequent words reduces the dataset size which degrades the results. The average caption length increases for `t > 2000` since the overall dataset size decreases but the 10% captions are always intact.

	Avg. Cap Len	Δ Dset Size	IN1k-ZS De CLUTR	IN1k-ZS Distil BERT
Intact captions	10.3	0%	27.7	29.6
`RmTopFreq(0)`	10.3	0%	27.8	29.1
`RmTopFreq(500)`	3.4	-10.3%	29.3	30.3
`RmTopFreq(1000)`	3.2	-19.6%	29.1	30.2
`RmTopFreq(2000)`	3.1	-34.1%	28.3	29.1
`RmTopFreq(4000)`	3.4	-53.6%	26.8	27.5
`RmTopFreq(8000)`	4.7	-73.5%	21.4	23.3

Table 6: **Ablation of training hyperparameters.** All models are trained with intact captions. We can see that 8k memory bank size and 20 epochs work well for all inits with DistilBERT being the best, and longer training only leads to marginal gains. Thus, we select 8k memory bank size and 20 epochs as the default setting for most experiments.

Memory	Epochs	IN1k-ZS	
		DeCLUTR	DistilBERT
8k	10	25.9	28.0
8k	20	27.7	29.6
8k	30	28.2	29.8
8k	40	28.5	30.3
16k	10	25.8	27.4
16k	20	28.5	29.4
		RoBERTa	BERT
8k	10	25.3	26.5
8k	20	28.2	28.5
16k	10	25.2	25.3
16k	20	27.3	28.0

Table 7: **Ablation of average vs.** `[CLS]` **pooling:** We explore the effect of using `[CLS]` pooling strategy to go from a list of tokens to a sentence level embedding. Averarge pooling is the default strategy, but we show that switching to `[CLS]` does not significantly change the results. The BoW models are still better than their intact counterparts by similar margins.

		IN-1k-ZS	
		Avg-pooling	`[CLS]`
RoBERTa	Intact	28.3	28.3
	BoW	28.7	28.9
	Δ	+0.4	+0.6
BERT	Intact	28.4	28.1
	BoW	28.2	28.4
	Δ	-0.2	+0.3
DeCLUTR	Intact	27.9	27.9
	BoW	29.1	28.9
	Δ	+1.2	+1.0
DistilBERT	Intact	29.5	29.3
	BoW	30.1	29.8
	Δ	+0.6	+0.5

Table 8: **Transfer linear evaluation:** We evaluate the models by training linear probes over frozen vision encoders. We see little difference between using BoW and intact caption models. We also see little effect of the text encoder's architecture over the performance of the vision encoder. Note that unlike the zero-shot setting, only the vision encoder is used during evaluation. Finally, we see that both BoW and intact caption models are better than the BYOL [24] initialization. This suggests that natural language supervision improves the quality of the visual features.

		Food101	CIFAR100	Sun397	DTD	Pets	Caltech101	Flowers	Mean
BYOL [24]	-	75.3	78.4	62.2	75.5	90.4	94.2	96.1	81.7
RoBERTa	Intact	77.7 (±0.2)	78.6 (±0.4)	66.4 (±0.1)	76.8 (±0.2)	90.6 (±0.1)	94.9 (±0.1)	96.5 (±0.1)	83.1
	BoW	77.5 (±0.0)	78.2 (±0.1)	65.6 (±0.1)	76.0 (±0.3)	91.0 (±0.3)	94.7 (±0.2)	96.3 (±0.1)	82.8
BERT	Intact	77.7 (±0.3)	79.1 (±0.3)	66.1 (±0.1)	76.7 (±0.2)	90.6 (±0.2)	94.8 (±0.0)	96.7 (±0.1)	83.1
	BoW	77.8 (±0.2)	78.6 (±0.4)	65.5 (±0.0)	76.3 (±0.2)	91.1 (±0.1)	94.8 (±0.1)	96.4 (±0.1)	82.9
DeCLUTR	Intact	77.9 (±0.1)	78.9 (±0.3)	66.3 (±0.1)	77.0 (±0.2)	90.7 (±0.1)	94.8 (±0.1)	96.4 (±0.1)	83.1
	BoW	77.7 (±0.1)	78.5 (±0.2)	65.7 (±0.1)	76.2 (±0.3)	90.9 (±0.1)	94.6 (±0.2)	96.4 (±0.0)	82.8
DistilBERT	Intact	77.8 (±0.2)	78.8 (±0.3)	66.5 (±0.1)	76.3 (±0.3)	90.3 (±0.8)	94.8 (±0.1)	96.5 (±0.2)	83.0
	BoW	77.6 (±0.2)	78.2 (±0.3)	65.7 (±0.1)	75.8 (±0.3)	90.9 (±0.3)	94.7 (±0.2)	96.5 (±0.1)	82.8

Table 9: **Detailed zero-shot results:** We report detailed zero-shot results here. These results were used to generate the Figure 3 of the main text. We can see that the results are consistently improved for all architectures on Pets by minimum $\sim$ 5 points and on DTD by minimum $\sim$ 2 points. On average, BoW is better than using intact captions by 0.6 points for RoBERTa, 0.4 points for BERT, 2.2 points for DeCLUTR, and 1.9 points for DistilBERT. This shows that the model's improvements with BoW are relatively higher compared to using intact captions.

		ImageNet	Food101	CIFAR100	Sun397	DTD	Pets	Caltech101	Flowers	Mean
RoBERTa	Intact	28.3 (±0.4)	17.8 (±0.3)	31.1 (±0.9)	38.6 (±0.2)	23.2 (±1.1)	17.5 (±0.7)	71.7 (±1.7)	12.8 (±0.6)	30.1
	BoW	28.7 (±0.8)	15.3 (±0.7)	32.8 (±2.7)	35.0 (±0.7)	25.6 (±1.0)	22.8 (±0.8)	71.2 (±1.2)	14.2 (±0.9)	30.7
	Δ	+0.4	-2.5	+1.7	-3.5	+2.4	+5.3	-0.5	+1.4	+0.6
BERT	Intact	28.4 (±0.2)	18.0 (±0.4)	30.9 (±1.4)	35.1 (±0.5)	26.0 (±0.6)	20.4 (±0.8)	70.5 (±1.9)	16.1 (±0.8)	30.7
	BoW	28.2 (±0.1)	17.2 (±0.8)	30.8 (±2.0)	32.4 (±0.2)	30.5 (±1.1)	26.5 (±1.0)	66.8 (±1.2)	16.6 (±0.7)	31.1
	Δ	-0.2	-0.8	-0.0	-2.7	+4.5	+6.0	-3.8	+0.5	+0.4
DeCLUTR	Intact	27.9 (±0.2)	17.6 (±0.6)	28.5 (±1.1)	36.0 (±0.3)	24.3 (±1.2)	19.7 (±0.8)	69.3 (±1.3)	15.5 (±0.8)	29.9
	BoW	29.1 (±0.2)	17.2 (±0.5)	32.3 (±1.7)	33.8 (±0.2)	31.7 (±1.1)	24.9 (±0.4)	68.4 (±3.0)	19.1 (±0.7)	32.1
	Δ	+1.2	-0.4	+3.8	-2.1	+7.4	+5.2	-0.9	+3.5	+2.2
DistilBERT	Intact	29.5 (±0.1)	17.2 (±0.4)	30.8 (±1.4)	38.0 (±0.4)	23.6 (±1.3)	20.8 (±1.0)	72.3 (±0.7)	13.5 (±0.6)	30.7
	BoW	30.1 (±0.1)	17.1 (±0.5)	33.7 (±1.7)	36.3 (±0.2)	29.9 (±1.0)	25.9 (±1.0)	72.0 (±0.6)	15.6 (±0.6)	32.6
	Δ	+0.6	-0.1	+2.9	-1.8	+6.3	+5.0	-0.2	+2.1	+1.9

A.1 Utilizing Unaligned Images

The insight that intact captions can be effectively replaced with BoW opens the opportunity to use unaligned images (images without associated captions) by pseudo-labeling them with BoW in a process similar to self-training [39]. The goal is to leverage unaligned data which can be easier to acquire and maintain in comparison to aligned data. We call this setting semi-aligned learning since only a subset of data is aligned.

Method: We first train a model with fully aligned set S_a of image-caption pairs, and then use it to generate pseudo-labels for unaligned images. Given an unaligned image x_i, we augment it c times with random cropping and resizing to get c augmented images $\{x_i^b\}_{b=1:c}$. Next, we get features for each x_i^b by passing it through the vision encoder $u_i^b = f(x_i^b)$. Then, for each u_i^b, we retrieve its top-k nearest neighbors from a set of captions, and put them in a set $NN_i = \{y_i^j\}_{j=1:kc}$. The set of captions used for retrieval are intact and come from the set S_a which was also used for training the fully-aligned model. Next, we use different strategies to rank the words in the NN_i and then pick first p words. The goal of these word ranking strategies is to find the set of words that best describe the image. The entire process is illustrated in Figure 3 and the resulting pseudo-labels for unaligned images are visualized in Figures 4.

Word Ranking Strategies: `MaxCount`: rank each word according to the count of times it is present at least once in a caption $y_i^j \in NN_i$. `RmTopMaxCount`: each word is ranked according to the `MaxCount` strategy, but similar to the text deformation discussed previously, 1000 most frequent words in the S_a captions are removed. The goal is to focus more on the infrequent words. `RmTopRand`: rank

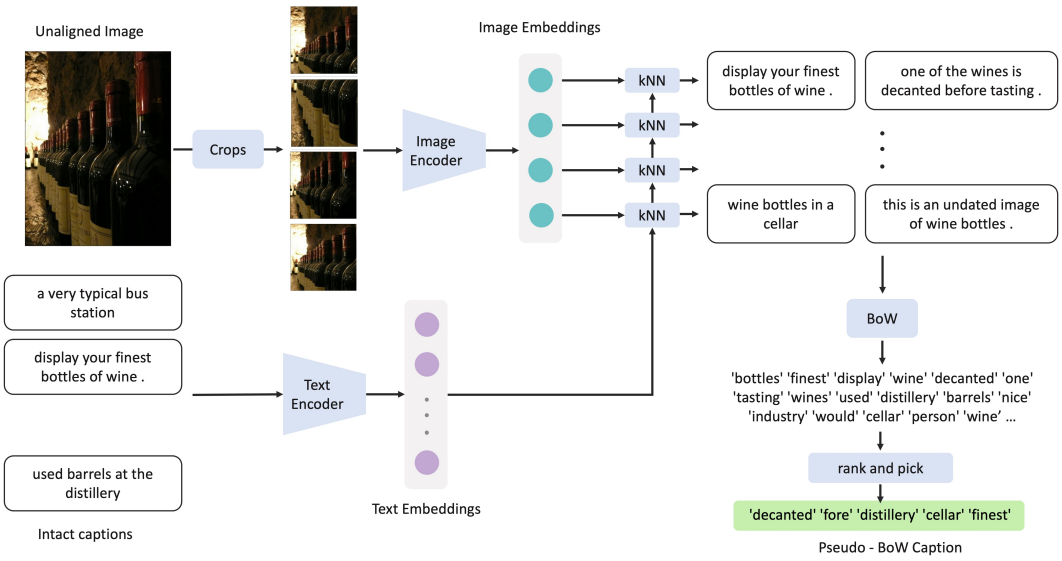

Figure 3: **Generating pseudo BoW captions:** We illustrate the process of constructing pseudo BoW captions for unaligned images in this figure. A fully aligned dataset, CC [22], to train the vision and text encoders and to obtain the intact captions used for retrieval. We find nearest neighbors in the text domain for different crops of a given unaligned image. The retrieved captions of all crops are then aggregated into a BoW. The words are then ranked according to one of the strategies mentioned in Section 2.3 of main text, and a few top words (4 in this figure) are chosen as the pseudo BoW caption. We used the `WeightedCount` strategy in this figure.

Table 10: **Semi-aligned**. We explore using additional images without captions (unaligned). We experiment with various word ranking strategies for creating BoW pseudo caption and find that `WeightedCount` works the best and is on-par with the CLIP model that is trained with 15M aligned data. A model from the first row is used to for pseudo-labeling. We also compare with a longer trained baseline for a fair comparison.

	Aligned	Un-aligned	Epochs	IN1k ZS
Aligned-only (ours)	2.3M	-	20	30.1
Aligned-only (ours)	2.3M	-	40	30.7
CLIP	15M	-	32	31.3
`RmTopRand`	2.9M	3M	10	30.5
`RmTopMaxCount`	2.9M	3M	10	31.1
`MaxCount`	2.9M	3M	10	31.4
`WeightedCount`	2.9M	3M	10	31.5

the words randomly in `RmTopMaxCount` instead of ranking them based on count. `WeightedCount`: re-rank the words in `MaxCount` strategy with the inverse of their counts in the fully aligned caption set S_a. Specifically, rank each word $w_p \in NN_i$ by the ratio $count(w_p, NN_i)/count(w_p, S_a)$. Instead of removing the most frequent words, this strategy simply ranks them lower.

Experiments: In addition to the fully aligned CC dataset [22] of size $\sim$ 3M, we use 3M randomly sampled images from ImageNet-21k [23] during this experiment. We use the DistilBERT model with 30.2% on the IN-1k-ZS benchmark for pseudo-labeling. We selected this model randomly from one of the 4 models trained with above configuration. It is a BoW model trained with default hyperparameters. The pseudo-labels are only calculated once at the beginning of the training after which they remain frozen. The results are reported in the Table 10. The total number of iterations used by a semi-aligned model are larger since it relies on an already trained, aligned-only model. Thus, we compare semi-aligned models with a longer trained baseline.

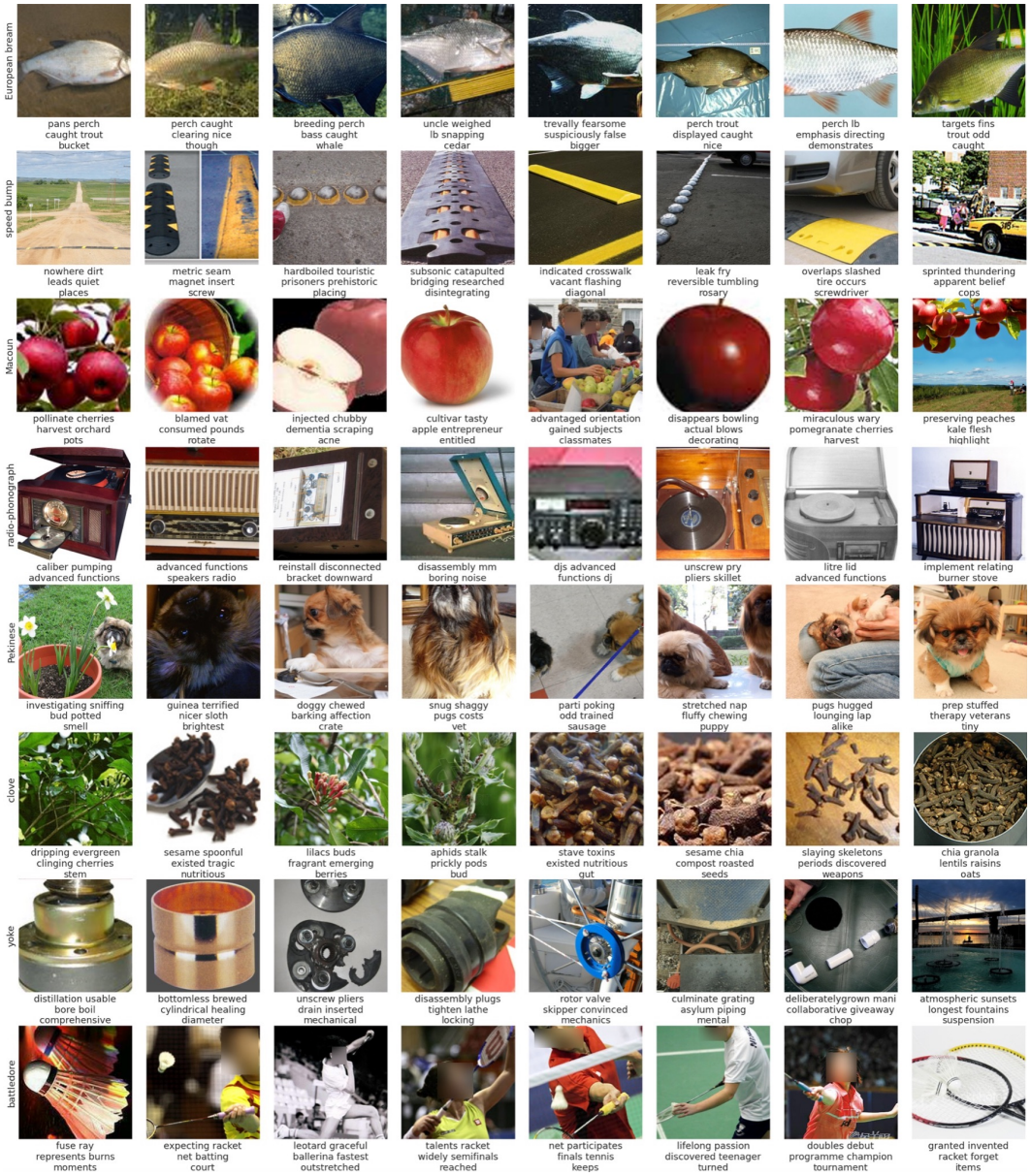

Figure 4: **Examples of pseudo BoW captions:** We show some examples of the pseudo BoW captions generated by the `WeightedCount` strategy. Each row contains images from a single category. The actual category name is listed on the left, at the start of each row, and the pseudo captions are below each image.