Team BTC(3): Final Report
CSE 481N

JoonHo (Brian) Lee, Thai Quoc Hoang, Connor McMonigle
{joonl4, qthai912, cmcmon}@uw.edu

Spring 2022

Abstract

Modern research in Image Captioning typically utilizes transformers to achieve high accuracy. However, these methods at a large scale require both substantial amounts of data and compute, which makes training often challenging. To address this issue, we propose to train a mapping network between a pretrained image encoder and text decoder for efficiency. Our approach, based on ClipCap, explores improved utilization of the pretrained models, yielding improved performance on the COCO Captions dataset while training only the mapping network. This report has been developed as part of a Capstone class (CSE481N, University of Washington), and our code is available on https://github.com/quocthai9120/UW-NLP-Capstone-SP22.

1 Introduction

The task of Image Captioning provides opportunities for research on multimodal (vision and language) learning, and to aid the visually impaired. With advancements in Deep Learning (Goodfellow et al., 2016), the latest methods incorporate various neural network architectures (e.g. Covolutional Neural Networks, Recurrent Neural Networks). In a prior method, Xu et al. (2015) utilizes a CNN encoder and an RNN decoder for their captioning model. More recently, Li et al. (2020); Liu et al. (2021); Wang et al. (2022); Zhou et al. (2020) each proposed transformer (Vaswani et al., 2017) based models to unify the encoder-decoder architecture, achieving new state-of-the-art performance.

However, training for these methods require large amounts of data and compute. Mokady et al. (2021) addressed the problem by utilizing a frozen image model and text decoder and training a mapping network that routes the features between the modules. Their framework ClipCap (Mokady et al., 2021) is lightweight in comparison to state-of-the-art approaches, albeit with some drop in performance. For our Capstone project, we explore augmenting the framework in two ways: 1) incorporating CLIP's text encoder in caption generation i.e. beam search, and 2) extracting spatial features from CLIP's image encoder by removing the final aggregation layer. Results for our method, ClipCap++, show that with minimal changes achieves competitive results on the COCO Captions dataset (Lin et al., 2014).

In this report, we first lay foundations on the ClipCap (Mokady et al., 2021) framework (Sec 2). We then introduce technical contributions to the framework on guided text decoding and spatial feature extraction (Sec 3). Our evaluations show the additions to the framework yield an improvement over the baseline on the COCO Captions dataset (Sec 4). Additionally, we conduct a set of ablation studies on individual components (Sec 4.4). For a broader context, we also provide an overview of related works (Sec 5).

2 Background

We consider ClipCap as our baseline model. The input image is first processed through the CLIP image encoder, extracting a global feature vector $F_t \in \mathbb{R}^D$, where $D = 512$ for ViT(Dosovitskiy et al., 2021) and

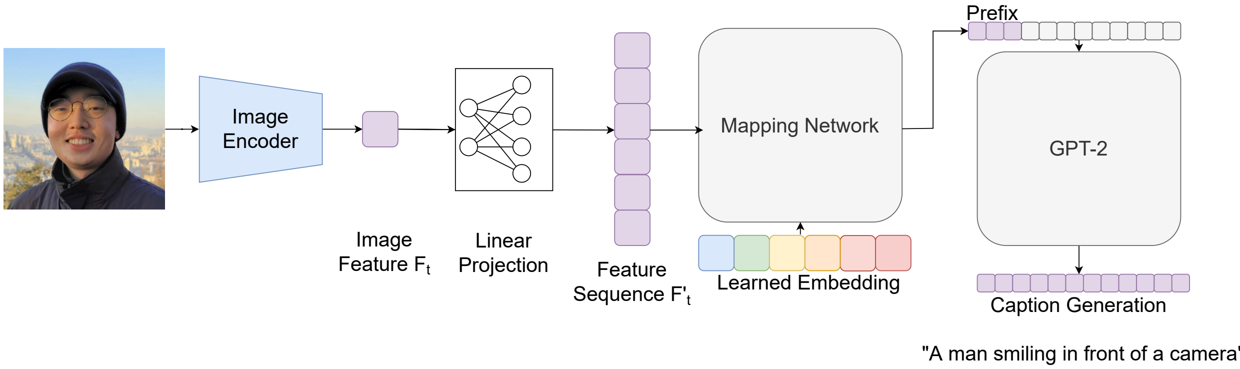

Figure 1: ClipCap framework, re-illustrated (cf. Mokady et al., 2021)

$D = 640$ for ResNet (He et al., 2016). The feature vector is processed through a linear projection layer and reshaped to get a sequence of features $F'_t \in \mathbb{R}^{N \times D}$, where N is the prefixed sequence length. The attention-based mapping network processes the sequence of features to generate the caption prefix, which is then fed to GPT-2 as an input prompt for caption generation. The pipeline is illustrated in Figure 1.

3 Method

3.1 CLIP-guided text decoding

ClipCap extracts image features using CLIP's image encoder, but does not utilize the text encoder in any way. Hence, we propose to incorporate the text encoder by augmenting beam search with the cosine similarity score between the image features and the text features from generated captions, as illustrated in Algorithm 1. Our approach is closely related to Twist Decoding (Kasai et al., 2022). Highlighted in blue, the additional scoring function exploits CLIP's original training objective to guide the search towards semantically corresponding captions. As we quantitatively analyze in Sec 4, the additional guidance leads to an improvement over the baseline framework, and does not require additional training as the frozen CLIP model is used. λ defines an additional hyperparameter used to scale the cosine similarity score. We found $\lambda = 0.2$ and $\lambda = 1.0$ to produce the best results for the baseline and spatial feature model respetively.

Algorithm 1 CLIP-guided Beam Search

Input length n, text encoder g,θ_g, captioning model f,θ_f, base hypothesis H_0, vocabulary $\mathcal{L}$, image x_I, λ

for $i \leftarrow 1, n$ **do**
  $H_i = \emptyset$
  for sequence h_{i-1} scored by H_{i-1} **do**
    for token $y \in \mathcal{L}$ **do**
      $h_i = y \circ h_{i-1}$
      Extract features $\mathcal{F}_g = g(x_I)$, $\mathcal{F}_f = f(h_i)$
      score(h_i) = $H_{i-1}(h_{i-1}) + p(y|h_{i-1}, \theta_f) +$ $\lambda \cdot$ cosine similarity$(\mathcal{F}_g, \mathcal{F}_f)$
      $H_i \leftarrow h_i$ with score(h_i)
    end for
  end for
  Let H_i be top k best scored elements.
end for
Return best scoring h_n from H_n.

3.2 Spatial Feature Extraction

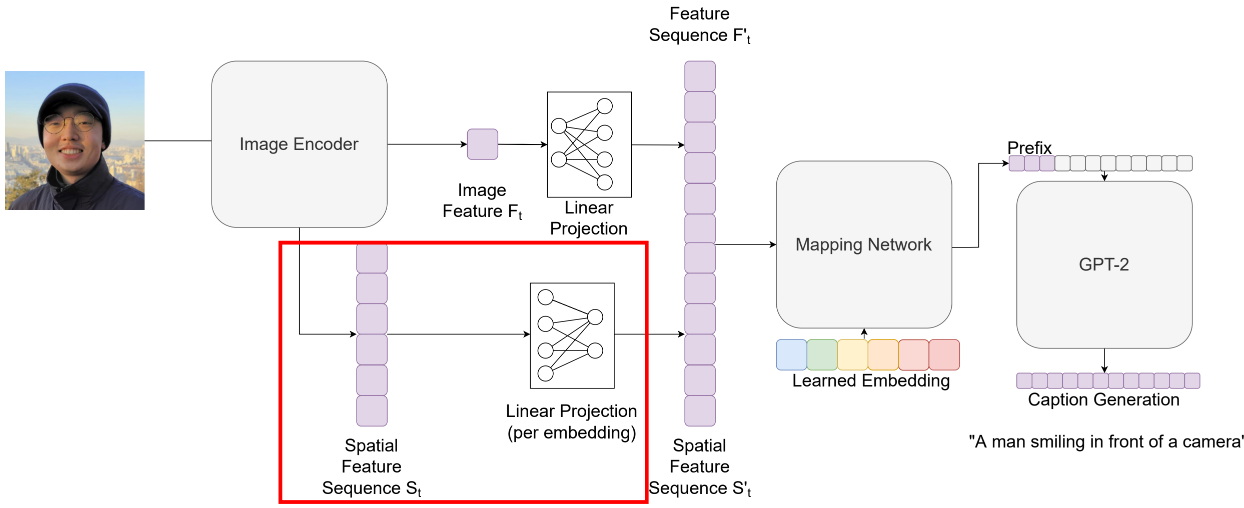

Figure 2: ClipCap incorporating additional spatial features from the ViT image encoder.

A potential limitation of the ClipCap (Mokady et al., 2021) architecture is the mapping network's reliance solely on the comparatively low dimensional, dense, image embedding vector yielded by the CLIP image encoder as input. The frozen CLIP image encoder yields dense 512 dimensional embedding vectors, $F_t \in \mathbb{R}^D, D = 512$ which are in turn linearly projected and processed by the mapping network into a 40 element sequence with an embedding dimension of 768, $F'_t \in \mathbb{R}^{N \times D}$, $N = 40$, $D = 768$, which serves as the prefix embedding for the pretrained language model used for captioning. Therefore, it is conceivable that caption-relevant image-specific features are lost in the CLIP image encoding due to its comparatively low dimensionality.

To combat this perceived limitation, we incorporate spatial feature embeddings from earlier layers in the CLIP image encoder as an additional input to the mapping network. The requisite modifications to the mapping network correspond to the red highlighted region in Figure 2. The individual elements of the CLIP ViT(Dosovitskiy et al., 2021) spatial feature embedding sequence directly correspond to regions of the input image. Therefore, it is hypothesized that the finer-detailed spatial feature embeddings extracted from the CLIP ViT image encoder eliminate the bottleneck introduced by the low dimensional CLIP embedding vector and preserve more caption-relevant spatial information from the input image.

To incorporate the aforementioned extracted spatial feature sequence as additional input to the ClipCap mapping network, the embedding vectors of the extracted spatial feature sequence, S_t, are first affine projected to obtain a sequence of embedding vectors of appropriate dimension, S'_t. This spatial feature embedding sequence is subsequently concatenated with the original affine linear projection of the CLIP image embedding along the sequence axis. As in the original ClipCap architecture, this sequence is next passed through an encoder-decoder transformer mapping network to obtain a prefix sequence. Accordingly, the last prefix-length elements of the resultant prefix sequence are used to condition the pretrained language model to yield image captions.

4 Empirical Section(s)

We evaluate our method on the COCO Captions dataset (Lin et al., 2014) using the Karpathy and Fei-Fei (2017) split for training and validation, with discussions in Sec 4.3. For consistency, we compare against the baseline model re-evaluated on our implementation. With regards to the individual components and the consistency of the baseline, we provide ablation studies in Sec 4.4.

4.1 Implementation details

Our implementation is based on existing implementations for ClipCap (Mokady, 2022), CLIP (jongwook, 2022), and GPT-2 (HuggingFace, 2022) respectively. Notably, for the spatial feature extraction model, the official CLIP implementation was modified such that the encodings learned by earlier layers could be extracted and forwarded to the mapping network. When training the spatial feature extraction model, we follow the same training scheme as the baseline, using AdamW (Loshchilov and Hutter, 2019) as the optimizer with a learning rate of 2e-5 and weight decay of 0.01. For guided decoding, we select $\lambda = 0.2$ for the base model and $\lambda = 1.0$ for the spatial feature extraction model.

4.2 Evaluation Metric

We evaluate each method with BLEU (Papineni et al., 2002), METEOR (Denkowski and Lavie, 2014), CIDEr (Vedantam et al., 2014), and SPICE (Anderson et al., 2016). Since the official implementation of ClipCap does not provide evaluation, we use our own implementation based on Zhou (2018).

4.3 Comparison with ClipCap

Metric	Baseline	Self-evaluated	ClipCap++
BLEU@4	33.53	33.4	**35.4**
METEOR	27.45	**27.5**	27.1
CIDEr	113.08	110.9	**112.2**
SPICE	21.05	20.4	20.4

Table 1: Comparison of our methods to the ClipCap baseline. **Baseline** refers to the originally reported results, while **Self-evaluated** reports results from running our evaluation code on a reproduced model. **ClipCap++** shows results from our proposed method, which improves over the baseline on BLEU and CIDEr.

We report our main experiment results in Table 1. Our method, ClipCap++, shows an improvement on BLEU and CIDEr over the baseline. Considering the improvements and the relatively minor drop in METEOR and no change in SPICE, we find our approach to yield competitive results over the base ClipCap Framework. Furthermore, as we report in Table 2, our approach is able to generate convincing captions that closely match the ground truth for unseen examples.

	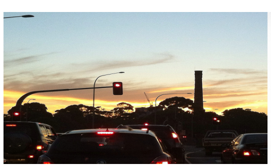			
Ground Truth	A group of cars stopped at a stop light.	A woman kneeling down to pet a large brown dog.	A kitchen filled with lots of pots, pans and dishes.	A motorcycle parked in a parking lot next to a car.
ClipCap++	Cars are stopped at a red light at an intersection.	A woman sitting next to a brown dog.	A kitchen with lots of pots and pans.	A motorcycle parked next to a parked car.

Table 2: Some results from the COCO validation set (Karpathy and Fei-Fei (2017) split).

4.4 Ablation Studies

We answer the following three questions to the best of our abilities:

1. How much does each component contribute to the method?
2. How does λ affect performance?
3. How valid is the comparison with the reproduced baseline?

How much does each component contribute to the method?

Metric	Baseline	Self-evaluated	Guided Decoding	Spatial Features	ClipCap++
BLEU@4	33.53	33.4	34.4	33.4	**35.4**
METEOR	27.45	**27.5**	27.4	27.3	27.1
CIDEr	113.08	110.9	111.9	110.5	**112.2**
SPICE	21.05	20.4	20.2	**20.6**	20.4

Table 3: evaluation of the individual components.

We make comparisons on the baseline with guided decoding, and spatial feature extraction with and without decoding, reported in Table 3. Results show the two modules combined are necessary for the best performance. We observe that while Spatial Features alone does not perform better than the baseline, when coupled with guided decoding produces much more competitive results. Specifically, the combined approach shows +2 improvement on BLEU and +1.7 improvement on CIDEr compared to +1 on BLEU and +1 on CIDEr for the baseline with decoding.

How does λ affect performance?

Metric \ λ	0 (no guidance)	0.15	0.2	0.3	0.5	1.0
BLEU@4	33.4	34.1	34.4	34.7	**34.8**	**34.8**
METEOR	**27.5**	27.4	27.4	27.3	27.1	26.6
CIDEr	110.9	111.7	111.9	**112.0**	111.6	110.8

Table 4: Comparison over different values of λ for base model.

Metric \ λ	0 (no guidance)	0.1	0.2	0.3	0.5	1.0	1.5
BLEU@4	33.4	33.4	33.7	34.7	34.8	**35.4**	35.1
METEOR	27.3	27.1	**27.4**	27.3	27.1	27.1	26.9
CIDEr	110.5	110.3	111.9	112.0	111.6	**112.2**	111.7

Table 5: Comparison over different values of λ for spatial feature extraction model.

We report our results over different values of λ in Tables 4 and 5. Spatial Feature Extraction model shows a larger improvement in BLEU and CIDEr with guided decoding, while resulting in a minor drop in METEOR. Given CLIP is trained to extract similar features between captions and images, we believe that the detailed features from spatial feature extraction provide more consistent information for generating captions that match the images. Additionally, we observe a plateau and even a drop in performance as λ is increased to much higher values. The above results have been used to select the best values $\lambda = 0.2$ and $\lambda = 1.0$ for baseline and Spatial Feature Extraction model respectively.

How valid is the comparison with the reproduced baseline?

Metric	40/40/40/4	80/40/20/4	80/80/20/4
BLEU@4	31.0	32.2	32.8
METEOR	27.1	27.3	27.2
CIDEr	105.7	107.9	108.1
SPICE	20.4	20.1	20.2

Table 6: Comparison over different configuration of (Prefix Length / CLIP Length / Batch Size / Training Epochs) for reproducing ClipCap.

The reproduced results under various settings are shown in Table 6. To our surprise, the reported results for ClipCap has been difficult to achieve, while we reach matching scores on a model distributed by the authors. Note that due to limited computing resources, we set the batch size for the experiment of Prefix Length 80 to be 20. Training the baseline may be done more consistently by increasing the batch size, which may lead to results closer to the originally reported scores.

5 Related Work

Large language models pre-trained on web-scale datasets, such as BERT and GPT (Cohen and Gokaslan, 2020; Devlin et al., 2018), have revolutionized natural language processing. Through fine-tuning large attention-based transformer Vaswani et al. (2017) language models, researchers have been able to reach new levels of performance on a variety of language tasks. Inspired by the advances in natural language processing enabled by the self-supervised pre-training of large models, computer vision researchers have developed a number of self-supervised image representation learning strategies for vision models such as BYOL, Barlow twins, SEER (Goyal et al., 2021; Richemond et al., 2020; Zbontar et al., 2021). Closely related to these approaches, strategies involving the contrastive learning of both text and image encoders using a large, web-scale, dataset of captioned images have become ubiquitous, building upon the techniques first introduced with CLIP (Radford et al., 2021).

Following the development of such large pretrained models, researchers have focused on techniques involving fusing multiple large frozen pretrained models to realize new capabilities on various, mostly visual language modeling related, cross-domain tasks. ClipCap (Mokady et al., 2021), upon which this paper builds, introduces such an approach. ClipCap realizes an image captioning model by mapping CLIP image encodings to a prefix embedding used to condition the text generation of a GPT language model. More recently, models such as Flamingo (Alayrac et al., 2022), which fuses an image encoder trained with a CLIP objective and a large pre-trained language model by way of an intermediate Perceiver (Jaegle et al., 2021) network, have utilized multiple pre-trained models to realize a single model capable of a variety of visual language modeling tasks including image captioning, text-conditional image generation and visual question answering (Alayrac et al., 2022; Ramesh et al., 2021).

6 Conclusion

From this report, we proposed an efficient Image Captioning model that improves over ClipCap (Mokady et al., 2021) with a guided beam search using CLIP's text encoder and with a spatial feature extraction for reducing latent feature bottleneck. We could use our method to demonstrate competitive results on the COCO Captions dataset, while our ablation studies show that is crucial for the two modules to be used jointly.

Acknowledgments

We thank Mokady et al. (2021) for distributing their implementation of ClipCap. Additionally, we would like to thank Ofir Press, Yizhong Wang, and Professor Noah Smith for their guidance on this project.

References

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob Menick, Sebastian Borgeaud, Andrew Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikolaj Binkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, and Karen Simonyan. Flamingo: a visual language model for few-shot learning, 2022. URL https://arxiv.org/abs/2204.14198.

Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. Spice: Semantic propositional image caption evaluation, 2016. URL https://arxiv.org/abs/1607.08822.

Vanya Cohen and Aaron Gokaslan. Opengpt-2: Open language models and implications of generated text. *XRDS*, 27(1):26–30, sep 2020. ISSN 1528-4972. doi: 10.1145/3416063. URL https://doi.org/10.1145/3416063.

Michael Denkowski and Alon Lavie. Meteor universal: Language specific translation evaluation for any target language. In *Proceedings of the EACL 2014 Workshop on Statistical Machine Translation*, 2014.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2018. URL https://arxiv.org/abs/1810.04805.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=YicbFdNTTy.

Ian J. Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, Cambridge, MA, USA, 2016. http://www.deeplearningbook.org.

Priya Goyal, Mathilde Caron, Benjamin Lefaudeux, Min Xu, Pengchao Wang, Vivek Pai, Mannat Singh, Vitaliy Liptchinsky, Ishan Misra, Armand Joulin, and Piotr Bojanowski. Self-supervised pretraining of visual features in the wild, 2021. URL https://arxiv.org/abs/2103.01988.

Kaiming He, X. Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

HuggingFace. Hugging face transformers. https://github.com/huggingface/transformers, 2022.

Andrew Jaegle, Felix Gimeno, Andrew Brock, Andrew Zisserman, Oriol Vinyals, and Joao Carreira. Perceiver: General perception with iterative attention, 2021. URL https://arxiv.org/abs/2103.03206.

jongwook. Clip. https://github.com/openai/CLIP, 2022.

Andrej Karpathy and Li Fei-Fei. Deep visual-semantic alignments for generating image descriptions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39:664–676, 2017.

Jungo Kasai, Keisuke Sakaguchi, Ronan Le Bras, Hao Peng, Ximing Lu, Dragomir Radev, Yejin Choi, and Noah A. Smith. Twist decoding: Diverse generators guide each other, 2022. URL https://arxiv.org/abs/2205.09273.

Xiujun Li, Xi Yin, Chunyuan Li, Xiaowei Hu, Pengchuan Zhang, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, Yejin Choi, and Jianfeng Gao. Oscar: Object-semantics aligned pre-training for vision-language tasks. In *ECCV*, 2020.

Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft coco: Common objects in context, 2014. URL https://arxiv.org/abs/1405.0312.

Wei Liu, Sihan Chen, Longteng Guo, Xinxin Zhu, and Jing Liu. Cptr: Full transformer network for image captioning. *ArXiv*, abs/2101.10804, 2021.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Bkg6RiCqY7.

Ron Mokady. Clipcap: Clip prefix for image captioning. https://github.com/rmokady/CLIP_prefix_caption, 2022.

Ron Mokady, Amir Hertz, and Amit H Bermano. Clipcap: Clip prefix for image captioning. *arXiv preprint arXiv:2111.09734*, 2021.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA, July 2002. Association for Computational Linguistics. doi: 10.3115/1073083.1073135. URL https://aclanthology.org/P02-1040.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021. URL https://arxiv.org/abs/2103.00020.

Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation, 2021. URL https://arxiv.org/abs/2102.12092.

Pierre H. Richemond, Jean-Bastien Grill, Florent Altché, Corentin Tallec, Florian Strub, Andrew Brock, Samuel Smith, Soham De, Razvan Pascanu, Bilal Piot, and Michal Valko. Byol works even without batch statistics, 2020. URL https://arxiv.org/abs/2010.10241.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

Ramakrishna Vedantam, C. Lawrence Zitnick, and Devi Parikh. Cider: Consensus-based image description evaluation, 2014. URL https://arxiv.org/abs/1411.5726.

Peng Wang, An Yang, Rui Men, Junyang Lin, Shuai Bai, Zhikang Li, Jianxin Ma, Chang Zhou, Jingren Zhou, and Hongxia Yang. Unifying architectures, tasks, and modalities through a simple sequence-to-sequence learning framework. *arXiv preprint arXiv:2202.03052*, 2022.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In Francis Bach and David Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 2048–2057, Lille, France, 07–09 Jul 2015. PMLR. URL https://proceedings.mlr.press/v37/xuc15.html.

Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stephane Deny. Barlow twins: Self-supervised learning via redundancy reduction. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 12310–12320. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/zbontar21a.html.

Luowei Zhou. Microsoft coco caption evaluation. https://github.com/LuoweiZhou/coco-caption/tree/de6f385503ac9a4305a1dcdc39c02312f9fa13fc, 2018.

Luowei Zhou, Hamid Palangi, Lei Zhang, Houdong Hu, Jason Corso, and Jianfeng Gao. Unified vision-language pre-training for image captioning and vqa. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(07):13041–13049, Apr. 2020. doi: 10.1609/aaai.v34i07.7005. URL https://ojs.aaai.org/index.php/AAAI/article/view/7005.