Image Caption Generation With CLIP+GPT-2 Model

Siyuan Jing and Haonan Wu
Boston University
siyuan16@bu.edu and whn17@bu.edu

May 3, 2025

Abstract

To be finished

1 Introduction

Image caption generation, the task of recognizing images to generate natural language descriptions, lies at the critical intersection of computer vision and natural language processing. It plays an important role in a variety of applications, including image retrieval, accessibility for the visually impaired, and automated image content processing.

Recent advances in vision-language models have enabled more accurate and fluent caption generation by leveraging large-scale pretraining on aligned image-text pairs . These models demonstrate remarkable capabilities in understanding visual content and translating it into coherent text descriptions.

Motivated by these developments, and after reviewing related literature [1],we aim to replicate a CLIP+GPT-2 based image captioning system. Through this project, we seek to explore and deepen our understanding of machine learning, as well as its broader applications in fields such as computer vision and natural language generation.

- Leverage the pretrained CLIP model to extract semantically rich image embeddings without the need for training a custom vision encoder.
- Utilize the generative capabilities of GPT-2 to produce fluent and coherent natural language captions.
- Bridge the gap between visual and textual modalities by introducing a projection layer that maps image embeddings into GPT-2's input space.
- Enable flexible and data-efficient image captioning, where the visual semantics guide the generation through prefix-based conditioning.
- Evaluate the quality of the generated captions using standard metrics such as BLEU and CIDEr, in order to quantitatively assess the model's accuracy and relevance.

2 Background and Related Work

2.1 CLIP(Contrastive Language-Image Pretraining)

CLIP from OpenAI is a visual-language model. Instead of relying on task-specific supervised learning, CLIP is trained on a dataset of 400 million image-text pairs collected from the internet using a contrastive loss function. CLIP consists of two separate encoders: a visual encoder (ResNet) for images, and a text encoder (Transformer) for captions. Its ability to generate rich, semantically meaningful image embeddings makes CLIP a powerful foundation for our systems and an ideal visual component in our CLIP+GPT-2 image captioning pipeline. For example, according to Mokady et al.[2], it mentioned that the visual encoding capability of CLIP can be used to embed and project the generated images into the input space of GPT-2 to generate prefixes, which helps the final caption generation of GPT-2. Inspired by this article, we decided to study the CLIP architecture and implement related deployments.

2.2 CLIP(Contrastive Language-Image Pretraining)

GPT-2 is a large-scale language model based on the Transformer decoder architecture proposed by Radford et al. [3]. According to the paper, Transformer completely replaces the traditional RNN or CNN structure with self-attention, which is more efficient and accurate when processing long sequence dependencies. Therefore, we consider implementing the transformer structure as our decoder of the whole pipeline. Unlike the traditional Transformer, which contains both encoder and decoder components, GPT-2 uses only a decoder. This design enables the model to predict the next token based solely on previously generated tokens, making the generated text semantically relevant and well suited for text generation tasks such as image captioning.

$$L = -\sum_{i \in I} \log \frac{\exp(\text{sim}(z_i, z_j)/\tau)}{\sum_{k \neq i} \exp(\text{sim}(z_i, z_k)/\tau)} \tag{1}$$

where $\text{sim}(z_i, z_j)$ is the cosine similarity between embeddings, and τ is the temperature hyperparameter.

2.3 Comparison to Classical Methods

Prior to contrastive learning, traditional unsupervised feature learning methods included:

- **Principal Component Analysis (PCA):** Projects data onto lower-dimensional eigenvectors with maximal variance.
- **Autoencoders:** Learn representations by reconstructing input data through an encoder-decoder framework.

We compare these classical approaches with SimCLR to understand its advantages and limitations.

3 Methodology

3.1 Implementation Details

Backbone Network: We use a ResNet-18 as the feature encoder. The final representation is fed into a two-layer MLP projection head.

Augmentation Pipeline: SimCLR relies heavily on augmentations, which include:

- Random crop and resize
- Color jittering
- Gaussian blur
- Horizontal flipping

Training Details: We train SimCLR on CIFAR-10 and STL-10 using:

- Batch size: {128, 256, 512}
- Temperature parameter: {0.07, 0.1, 0.5}
- Optimizer: Adam with a learning rate of 3×10^{-4}
- Number of epochs: 200

3.2 Evaluation Protocol

We evaluate the quality of learned representations by training a simple linear classifier on top of the frozen embeddings.

4 Experiments and Results

4.1 Dataset and Preprocessing

We conduct experiments on:

- **CIFAR-10:** A 10-class dataset with 60,000 images.
- **STL-10:** A larger dataset often used for unsupervised learning benchmarks.

4.2 Baseline Comparisons

We compare SimCLR embeddings with:

- PCA-based dimensionality reduction ($d = 128$)
- Autoencoders trained on the same dataset
- Supervised ResNet-18 trained on CIFAR-10

Method	CIFAR-10 Accuracy (%)	STL-10 Accuracy (%)
Supervised ResNet-18	92.5	85.4
PCA + kNN	45.6	38.2
Autoencoder + kNN	55.3	49.6
SimCLR (Ours)	80.2	76.4

Table 1: Comparison of representation learning methods. SimCLR significantly outperforms classical techniques.

4.3 Ablation Studies

Effect of Temperature τ: We analyze how different values of τ in the contrastive loss impact performance.

5 Discussion

5.1 Key Findings

- SimCLR significantly outperforms PCA and autoencoders in feature learning.
- The choice of augmentations greatly affects performance.
- Higher temperature values in contrastive loss lead to better separation of features.

5.2 Future Work

- Extend to other self-supervised methods (e.g., BYOL, MoCo).
- Apply to domain adaptation tasks.
- Explore contrastive learning for text or multimodal applications.

6 Conclusion

Our empirical study demonstrates the effectiveness of contrastive learning via SimCLR for representation learning. By systematically evaluating augmentation pipelines, batch sizes, and loss functions, we provide insights into optimizing contrastive learning for different datasets.

References

[1] Nukrai, D., Mokady, R., & Globerson, A. (2022). Text-Only Training for Image Captioning using Noise-Injected CLIP. https://doi.org/10.48448/n7sq-p557

[2] Mokady, R., Hertz, A., & Bermano, A. H. (2021). ClipCap: CLIP Prefix for Image Captioning. http://arxiv.org/abs/2111.09734

[3] Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). "Language Models are Unsupervised Multitask Learners." OpenAI Technical Report.

[4] Li, X., Yin, X., Li, C., Zhang, P., Hu, X., Zhang, L., Wang, L., Hu, H., Dong, L., Wei, F., Choi, Y., & Gao, J. (2020). "Oscar: Object-Semantics Aligned Pre-training for Vision-Language Tasks." ECCV 2020.

[5] Mokady, R., Hertz, A., & Bermano, A. H. (2021). "ClipCap: CLIP Prefix for Image Captioning." arXiv preprint arXiv:2111.09734.

[6] Papineni, K., Roukos, S., Ward, T., & Zhu, W. J. (2002). "BLEU: a method for automatic evaluation of machine translation." ACL 2002.

[7] Vedantam, R., Lawrence Zitnick, C., & Parikh, D. (2015). "CIDEr: Consensus-based image description evaluation." CVPR 2015.

[8] Ting Chen, Simon Kornblith, Mohammad Norouzi, Geoffrey Hinton. "A Simple Framework for Contrastive Learning of Visual Representations." ICML 2020.

[9] Geoffrey Hinton, Ruslan Salakhutdinov. "Reducing the Dimensionality of Data with Neural Networks." Science, 2006.