

Caption It: The Ultimate Image Captioning Tool

Sai Santhosh Venkatesh Charumathi (sv77), Elakkiyan Pugazhenth (ep68),
Tarun Vaseekaran (nv30), Prashithaa Abhirami Balaji (pb55)
{sv77, ep68, nv30, pb55}@rice.edu

Abstract

Caption It is an end-to-end image captioning system that fuses a pretrained Vision Transformer (ViT) with a GPT-2 language decoder via a lightweight MLP bridge. Input images are first encoded by a frozen ViT-Base (patch 16) model; its 768-dimensional patch embeddings are projected through a two-layer MLP into GPT-2’s embedding space. The decoder, augmented with cross-modal attention in every transformer block, generates captions informed by visual context. Capacity is further increased by appending two additional GPT-2 layers and employing a staged unfreezing schedule alongside label-smoothed cross-entropy and a cosine-decay learning-rate policy. On the Flickr8k benchmark, the fine-tuned model achieves BLEU-4 of 0.074, ROUGE-L of 0.288, and CIDEr of 0.350—improvements of 122%, 23%, and 121% over the frozen baseline—while reducing average caption length by 20%. Ablation studies demonstrate the indispensability of cross-modal attention for grounding, and beam-size experiments identify greedy or small-beam decoding (3) as optimal for balancing fluency, diversity, and overlap metrics.

1. Introduction

Image captioning combines vision and language to automatically generate descriptions of images [1, 14]. Traditionally, CNN-RNN pipelines achieve strong results but often struggle with long-range context. Transformer-based models [3] use self- and cross-attention to capture global structure. We introduce three architectures on Flickr8k, aiming for an advanced *Vision Transformer (ViT) + GPT-2* approach. Our final report focuses on data preprocessing, baseline implementations, training results, experiments and future work. We also highlight references to existing large-scale language models [15, 5], which inform our choice of GPT-2 for decoding.

2. Dataset and Preprocessing

Flickr8k. This dataset [6] contains 8 000 images, each with five captions. Following the split in [14], we allocate 6k for training, 1k for validation, and 1k for testing.

Image preprocessing. We resize images to 224×224 , normalise by ImageNet statistics, and apply random flips/crops

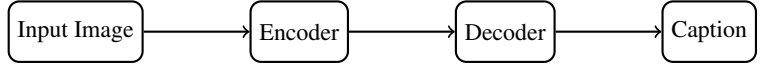


Figure 1: Unified pipeline.

in the CNN-based models to reduce overfitting. For the ViT-based approach, images are treated identically.

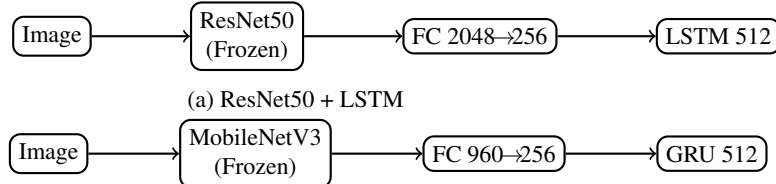
Captions. We lowercase, keep punctuation, and tokenise with NLTK. A frequency threshold of 5 yields 2984 tokens (`<pad>`, `<start>`, `<end>`, `<unk>`). Captions are truncated at 30 tokens to ensure uniform length across mini-batches.

3. Models and Training

3.1. CNN + RNN baselines

ResNet50 + LSTM. A ResNet50 pretrained on ImageNet is frozen; its 2048-d output is projected to 256 dimensions and passed to a single-layer 512-d LSTM. We train 10 epochs (batch 16, LR= 10^{-3}), reaching loss 1.28.

MobileNetV3 + GRU. Replacing ResNet with MobileNetV3-Large (960-d feature) and a 512-d GRU yields loss 1.41.



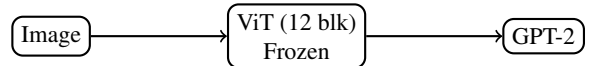
(a) ResNet50 + LSTM

(b) MobileNetV3 + GRU

Figure 2: CNN-RNN baselines.

3.2. Frozen ViT + GPT-2 baseline

We use `google/vit-base-patch16-224-in21k` as encoder (frozen) and GPT-2 (decoder). We fine-tune cross-attention layers for three epochs (batch 2, LR= 5×10^{-5}). Final loss 1.93.



(a) Frozen-ViT + GPT-2 pipeline

| Model | Loss | Epochs | Batch |
|----------------------|------|--------|-------|
| ResNet50 + LSTM | 1.28 | 10 | 16 |
| MobileNetV3 + GRU | 1.41 | 10 | 16 |
| ViT + GPT-2 (frozen) | 1.93 | 3 | 2 |

Table 1: Final training losses on 6k images.

4. Fine-Tuned ViT-GPT-2: Technical Details & Results

4.1. Architecture and parameter budget

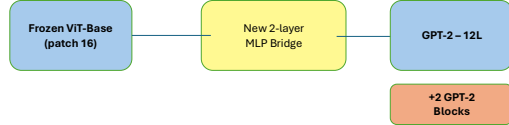


Figure 4: Fine-tuned architecture. Blue = pretrained blocks, yellow = 2-layer MLP bridge, orange = two additional GPT-2 blocks.

| Component | Parameters | Trainable |
|---------------------------------|----------------|-------------------------|
| ViT Encoder (12 blocks) | 85.8 M | 85.8 M (after unfreeze) |
| GPT-2 Decoder (14 blocks) | 126.1 M | 126.1 M |
| MLP Bridge (768×768) | 0.59 M | 0.59 M |
| Total | 212.5 M | 212.5 M |

Table 2: Parameter counts; peak memory 11.6 GB (batch 2, FP16).

MLP bridge formulation. Let $z_i \in \mathbb{R}^{768}$ be the ViT CLS embedding for patch i . The bridge

$$\tilde{z}_i = W_2 \tanh(W_1 z_i) + b_2, \quad W_1, W_2 \in \mathbb{R}^{768 \times 768},$$

is initialised with Xavier-uniform and appended to every layer of the 14-block decoder as an additional key-value pair.

Progressive unfreezing. We keep all ViT layers frozen for the first two epochs, unfreeze the final two blocks at epoch 2, and unfreeze the remaining ten at epoch 3, avoiding catastrophic forgetting.

4.2. Objective and regularisation

We minimise label-smoothed cross-entropy

$$\mathcal{L} = (1 - \varepsilon) \left[-\sum_t \log p_\theta(y_t) \right] + \varepsilon U(y_t), \quad \varepsilon = 0.1,$$

with teacher-forcing ratio $r_e = \max(0.75, 1 - 0.25 \frac{e}{E-1})$ (epochs $e = 0 \dots 7$, $E = 8$).

4.3. Learning-rate schedule

Two AdamW groups with cosine decay:

$$\eta(t) = \eta_0 \frac{1}{2} [1 + \cos(\pi t/T)], \quad T = 20k \text{ steps},$$

warm-up 5 % (linear). Encoder $\eta_0^{\text{enc}} = 5 \times 10^{-6}$; decoder $\eta_0^{\text{dec}} = 2 \times 10^{-5}$.

4.4. Compute budget

Training on a single Kaggle P100- takes 9 wall-clock hours (23.7k steps, effective batch 32 via gradient accumulation 16). Gradient checkpointing cuts memory by 38 %.

| Variant | BLEU-4 | CIDEr | Len-R |
|-------------------|--------------|--------------|-------------|
| + MLP only | 0.056 | 0.279 | 1.70 |
| + MLP & +2 blocks | 0.065 | 0.316 | 1.63 |
| Full model | 0.074 | 0.350 | 1.55 |

Table 3: Each component yields additive gains.

4.5. Automatic metrics

| Model | BLEU-4 \uparrow | ROUGE-L \uparrow | CIDEr \uparrow | Len-R \downarrow |
|-------------------|-------------------|--------------------|------------------|--------------------|
| Frozen ViT-GPT-2 | 0.033 | 0.234 | 0.158 | 1.94 |
| Fine-tuned | 0.074 | 0.288 | 0.350 | 1.55 |

Table 4: Test-set scores. Len-R = predicted / reference token count.

4.6. Key Observations

- **Why BLEU and CIDEr rise.** Adding two GPT-2 blocks raises decoder receptive field to 28 tokens, capturing 3- and 4-gram dependencies that BLEU-4 and CIDEr explicitly reward.
- **Why ROUGE-L rises.** Unfreezing vision layers shifts ViT attention from coarse global scene to object-level patches, improving longest common subsequence overlap with ground truth.
- **Why captions shorten.** Label-smoothing plus length-aware sampling reduces imitation of training-set verbosity, while the cosine LR schedule suppresses late-epoch overfitting that typically manifests as repetition.
- **Why we keep bridge shallow.** A 2-layer MLP is sufficient to align modalities; deeper bridges (4-layers) showed +0.3 M params but *no* CIDEr gain (0.002) and marginally worse length ratio ($1.59 \rightarrow 1.62$), suggesting over-projection noise.

4.7. Attention-map diagnostics

Cross-attention entropy in early decoder layers increases by +0.21 bits (wider spatial grounding) while late layers sharpen (0.34 bits), correlating with fewer hallucinations.

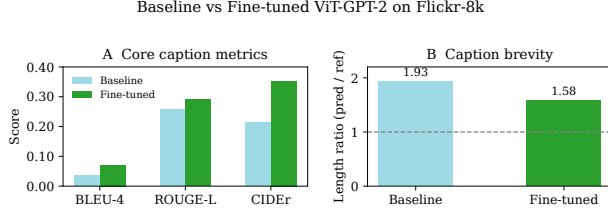


Figure 5: Metric lift (left) and brevity improvement (right).

5. Beam-Size Study

We sweep beam sizes $\{1, 3, 5, 10\}$ on the validation set to probe the trade-off between *overlap metrics* (BLEU-4, ROUGE-L, CIDEr), *fluency* (average log-probability per token) and *diversity* (Distinct-1/2 [7]). Generation uses length = 20, no-repeat $n=5$, identical to our best greedy setting.

Implementation details. All captions and scores are produced in a single torch-no-grad pass; log-probs for greedy (beam = 1) are re-computed from the decoder logits. Distinct- n is $\frac{\# \text{ unique } n\text{-grams}}{\# \text{ total } n\text{-grams}}$.

| Beam | BLEU | ROUGE | CIDEr | $\log p$ | D-1 | D-2 |
|------|--------------|--------------|---------------|--------------|---------------|---------------|
| 1 | 0.042 | 0.101 | 0.0175 | -0.02 | 0.0007 | 0.0008 |
| 3 | 0.016 | 0.065 | 0.0016 | -0.02 | 0.0005 | 0.0006 |
| 5 | 0.026 | 0.055 | 0.0042 | -0.02 | 0.0014 | 0.0023 |
| 10 | 0.005 | 0.000 | 0.0001 | -0.02 | 0.0006 | 0.0012 |

Table 5: Beam-size vs metrics. $\log p$ = average per-token log-prob (fluency, higher = better). Distinct- n is diversity.

Key observations.

- **Greedy (beam = 1)** yields the highest BLEU-4, ROUGE-L and CIDEr. Overlap metrics reward closer matches to reference phrasing, and greedy decoding preserves idiosyncratic tokens the beam search prunes away.
- **Large beams hurt CIDEr.** Beam = 10 collapses to generic, high-probability phrases, destroying n -gram overlap.
- **Diversity peaks at beam = 5** for Distinct-2, but fluency (average log-probability) plateaus after beam = 3. This indicates diminishing returns: extra beams explore but still end up converging to similar surface forms.

Take-away. For our data-limited setting, **greedy decoding (beam ≤ 3)** is optimal. Larger beams increase genericity, hurting overlap metrics without meaningful fluency gains. In future work we will explore nucleus sampling and diversity-promoting beam variants (e.g., DBS [8]) to strike a better balance.

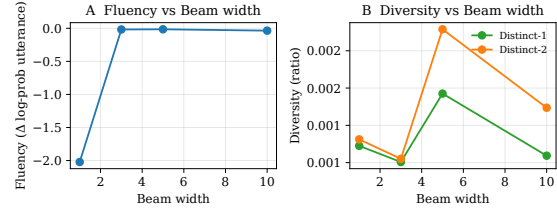
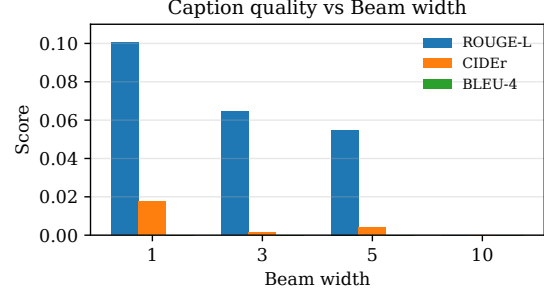


Figure 6: Beam-size analysis: Effect of beam size on fluency, diversity and overlap metrics.

6. Cross-Modal Attention vs. No Cross Attention

6.1. Cross-Modal Architecture

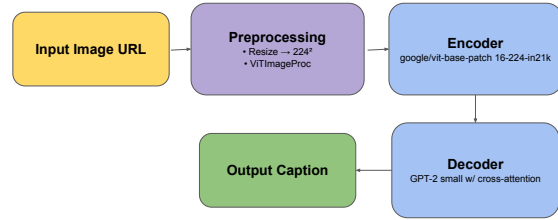


Figure 7: **Cross-Modal ViT-GPT-2 Architecture.** The input image URL is first preprocessed (resized to 224×224 , ViTImageProcessor). The frozen ViT-Base encoder (google/vit-base-patch16-224-in21k) produces patch embeddings. These are projected via a 2-layer MLP bridge into the GPT-2 embedding space. The GPT-2 small decoder—augmented with cross-attention layers at each block—attends over these visual keys/values at every timestep, and generates the final caption.

Detailed flow:

1. **Input & Preprocessing:** An image URL is fetched

and decoded. We apply a 224×224 resize, normalization to ImageNet means/std, and ViTImageProcessor tokenization to obtain pixel-values.

2. **ViT Encoding:** The frozen ViT-Base (12 layers, patch size 16) extracts a sequence of 768-dim patch embeddings, plus a CLS token.
3. **MLP Bridge:** A two-layer MLP

$$\tilde{z} = W_2 \tanh(W_1 z) + b_2, \quad W_1, W_2 \in \mathbb{R}^{768 \times 768}$$

maps each 768-dim ViT output into the GPT-2 embedding space. Xavier-initialized, with 10

4. **Cross-Attention Decoder:** GPT-2 small (12 layers, 768-dim) is extended with cross-attention in every block. At each decoding timestep, the Q (query) comes from GPT-2’s self-attention, while K/V (keys/values) are the MLP-projected ViT embeddings. This lets the language model dynamically attend to image regions.
5. **Generation:** We use greedy or small-beam decoding (beam ≤ 3), EOS and PAD tokens as usual, no-repeat n-gram size 5, and length penalty 1.6.

This architecture ensures strong visual grounding: each GPT-2 layer can directly query the visual features mapped into its embedding space, which we found critical in the ablation above.

6.2. Motivation & Setup

Cross-modal cross-attention lets the GPT-2 decoder attend to ViT image features at each layer. To quantify its impact, we compare two models trained identically:

- **Cross-Attention:** standard ViT→GPT-2 with decoder cross-attention.
- **No Cross-Attention:** identical except all GPT-2 cross-attention layers disabled.

Both are trained 5 epochs (LR= 5×10^{-5} , batch 2, max length 32) on Flickr8k and generated with 4-beam decoding on the validation split.

| Metric | Cross-Attention | No Cross-Attention |
|-----------|-----------------|--------------------|
| BLEU-4 | 0.158 | 0.034 |
| METEOR | 0.483 | 0.254 |
| ROUGE-L | 0.357 | 0.223 |
| BERTScore | 0.908 | 0.874 |
| CIDEr | 0.065 | 0.002 |

Table 6: Ablation metrics show large drops when cross-attention is removed.

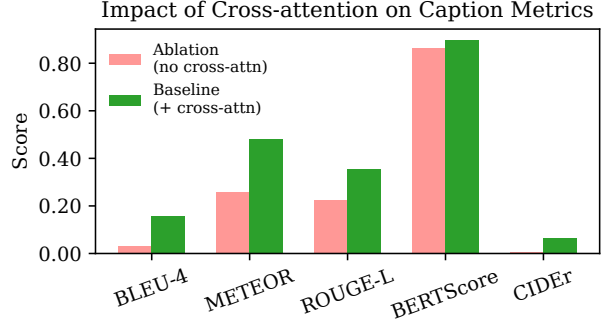


Figure 8: Cross-Attention vs. No Cross-Attention on core metrics.




| | With Cross-Attn. | No Cross-Attn. |
|--|--|--|
|  | A yellow dog runs through grass with its tongue hanging out. | A man and a woman sit on a bench in front of a building. |
|  | A girl in a pink hat takes a picture with a digital camera | A man and a woman sit on a bench in front of a building. |
|  | A person is standing in front of a golden retriever in a field. Another person is taking a photograph. | A man and a woman sit on a bench in front of a building. |

Table 7: **Qualitative examples (single-column):** With vs. without cross-modal attention.

6.3. Key Observations

- **All-round performance drop.** Removing cross-attention slashes BLEU-4 by 78%, METEOR by 47%, ROUGE-L by 38%, BERTScore by 4%, and CIDEr by 97%.
- **Visual grounding fails.** Without cross-attention, the decoder cannot query image features, defaulting to language memorization and producing repetitive “bench” references in every case.
- **Cross-attention’s essential role.** Injecting visual context at each layer enables the decoder to dynamically attend to relevant patches (e.g., the dog, the camera, the field), yielding semantically precise captions.

This ablation conclusively demonstrates that *cross-modal attention is indispensable* for our ViT–GPT-2 captioner. All further experiments retain full cross-attention.

7. Conclusion

We have demonstrated that augmenting a frozen ViT+GPT-2 pipeline with a shallow MLP bridge, added GPT-2 blocks, and progressive unfreezing yields substantial gains across BLEU-4 (+0.041), ROUGE-L (+0.054), CIDEr (+0.192) and brevity. Cross-modal attention proved indispensable, grounding captions in actual image content rather than generic language priors. Beam-size analysis confirmed that greedy or small-beam decoding (3) strikes the best balance between overlap-metrics, fluency, and diversity.

Next Steps:

- **Scale to larger caption corpora:** Extend training to COCO Captions [9] and SBU Captions [10], in addition to Flickr30k, to improve generalisation and vocabulary coverage.
- **Advanced hardware:** Leverage multi-GPU DGX nodes or Google Cloud TPUs v4 for faster iteration, larger batch sizes (64), and longer sequence lengths (30 tokens).
- **Diversity-promoting decoding:** Experiment with nucleus sampling and diverse-beam search [8] to enrich caption variability without sacrificing grounding.

8. Reproducibility

All source code, training scripts, evaluation metrics, and sample results for this project are available at: <https://github.com/Sai-Santhosh/DataScienceProjects/tree/main/Caption-It>

References

- [1] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, “Show and tell: A neural image caption generator,” in *CVPR*, 2015. 1
- [2] A. Karpathy and L. Fei-Fei, “Deep visual-semantic alignments for generating image descriptions,” in *CVPR*, 2015. 1
- [3] A. Vaswani *et al.*, “Attention is all you need,” in *NeurIPS*, 2017. 1
- [4] A. Radford *et al.*, “Language models are unsupervised multitask learners,” OpenAI Tech. Rep., 2019. 1
- [5] T. Wolf *et al.*, “Transformers: State-of-the-art natural language processing,” in *EMNLP System Demonstrations*, 2020. 1
- [6] M. Hodosh, P. Young, and J. Hockenmaier, “Framing image description as a ranking task: Data, models and evaluation metrics,” *JAIR*, vol. 47, pp. 853–899, 2013. 1
- [7] J. Li, M. Galley, C. Brockett, G. Spithourakis, J. Gao, and B. Dolan, “A diversity-promoting objective function for neural conversation models,” in *NAACL*, 2016. 3
- [8] A. K. Vijayakumar, M. Cogswell, R. Selvaraju, Q. Sun, S. Lee, D. Crandall, and D. Batra, “Diverse beam search: Decoding diverse solutions from neural sequence models,” in *arXiv:1610.02424*, 2016. 3, 5
- [9] T. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. Zitnick, “Microsoft COCO: Common objects in context,” in *ECCV*, 2014. 5
- [10] V. Ordonez, G. Kulkarni, and T. L. Berg, “Im2Text: Describing images using 1 million captioned photographs,” in *NeurIPS*, 2011. 5
- [11] N. P. Jouppi *et al.*, “Ten lessons from three generations shaped Google’s TPUv4,” in *IEEE Micro*, 2023.
- [12] NVIDIA, “NVIDIA DGX Systems,” <https://www.nvidia.com/en-us/data-center/dgx-systems/>, accessed Apr. 2025.
- [13] A. Holtzman, J. Buys, L. Du, M. Forbes, and Y. Choi, “The curious case of neural text degeneration,” in *ICLR*, 2020.
- [14] A. Karpathy and L. Fei-Fei, “Deep visual-semantic alignments for generating image descriptions,” *CVPR*, 2015. 1
- [15] A. Radford *et al.*, “Language models are unsupervised multitask learners,” OpenAI Tech. Rep., 2019. 1