RESEARCH INTERNSHIP

Remote Sensing Image Captioning with Vision Language Models

By Dhairya Arora
IIT (BHU), Varanasi

Under guidance of:
Dr. Yogendra Rao Musunuri
Prof. Oh-Seol Kwon
Changwon National University (CWNU)

ABSTRACT

This research internship focused on advancing the field of remote sensing image captioning using multimodal vision-language models. The primary objective was to evaluate and compare state-of-the-art captioning architectures—including PureT, BLIP-2, and BITA—on the RSICD dataset, a benchmark for remote sensing caption generation.

The tasks involved fine-tuning models using image-caption pairs, implementing custom data preprocessing pipelines, and assessing performance using standard captioning metrics such as BLEU, METEOR, ROUGE-L, and CIDEr. Efficiency evaluations—including inference time, model size, and FLOPs—were also conducted to assess the trade-offs between accuracy and resource consumption. Further work included model compression through PEFT-based fine-tuning and layer pruning to achieve better performance and results.

OBJECTIVES

- Evaluate the following Vision-Language models on the RSICD dataset:
  1. Pure T
  2. BLIP-2
  3. BITA

  and collect evaluation metrics such as BLEU, CIDEr, ROUGE, and METEOR.
- Compare model parameters, GFlops and inference time.
- Improve one of the models through fine-tuning or optimization, and compare the results with existing baselines.

DATASET OVERVIEW

The dataset used for evaluation is the RSICD Dataset. It consists of 10,921 images in total, with 8000 training images, 1000 validation images and 1921 test images. Each image has 5 human-annotated captions. The satellite images are sourced from diverse remote sensing scenes such as airports, forests and metropolitan cities.

EVALUATION METRICS

Metric	What it does
BLEU Score	Compares n-gram overlap between annotations and generated captions (precision-focused)
ROUGE-L	Takes longest common subsequences, useful for capturing structural similarity (recall-focused)
METEOR	Incorporates synonym and stem matching, with harmonic mean of precision and recall.
CIDEr	Uses TF-IDF weighting for consensus-based scoring.

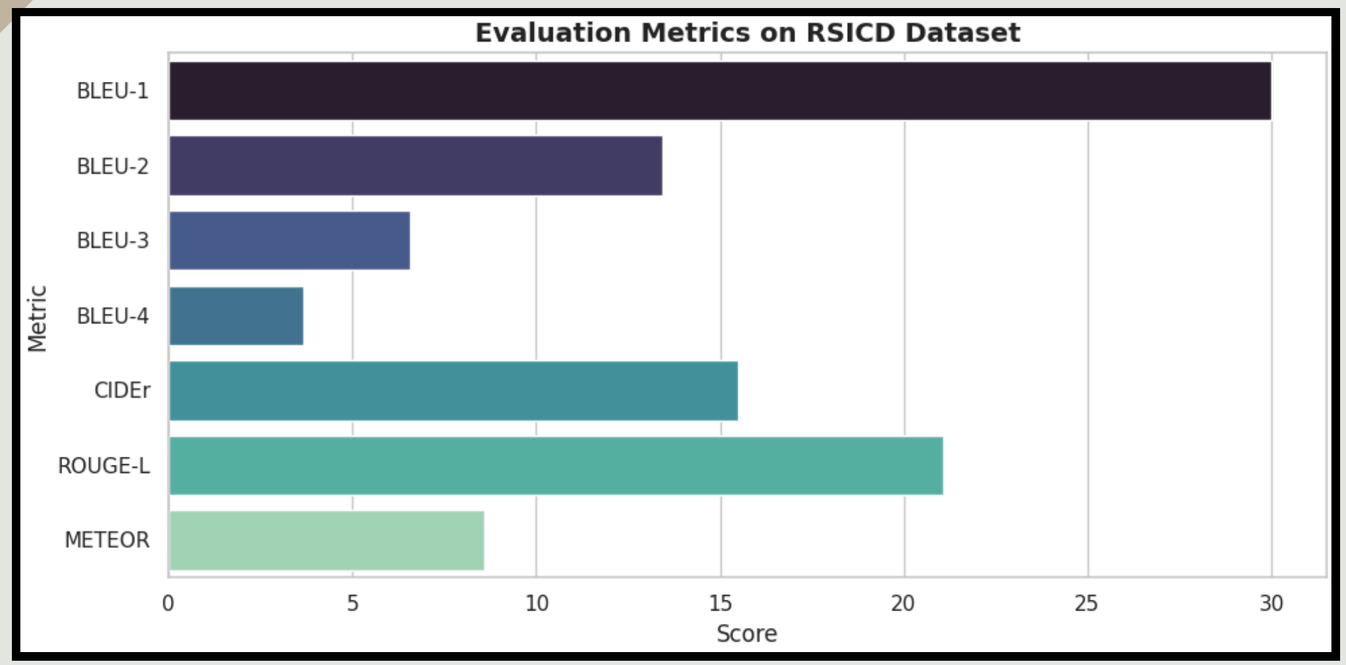
- Using different metrics captures different aspects of caption quality, giving a better idea of model performance.

PURE T

- The model is designed as a pure transformer-based architecture for image captioning, integrating the entire process into a single stage for end-to-end training, in contrast with prior works, which use a two-stage pipeline.

Component	Role
Swin Transformer	Extracts grid-level visual features from images
Refining Encoder	Refines visual features via self-attention
Decoder	Generates captions word by word
Global Feature Pool	Enhances multi-modal fusion in encoder/decoder

PURE T SIMULATION RESULTS



Model Parameters Overview

Metric	Value
Total Parameters	229.41M
Trainable Parameters	34.16M
Total GFLOPs	105.48
Avg Inference Time	416 ms

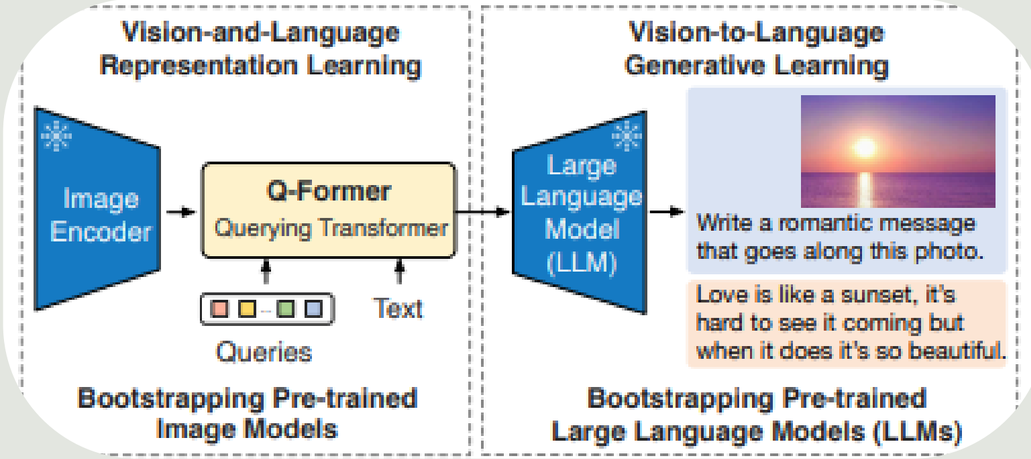
CAPTION GENERATION FOLLOWS BEAM SEARCH WITH NUM_BEAMS=5 AND GREEDY DECODE = TRUE

BLIP-2

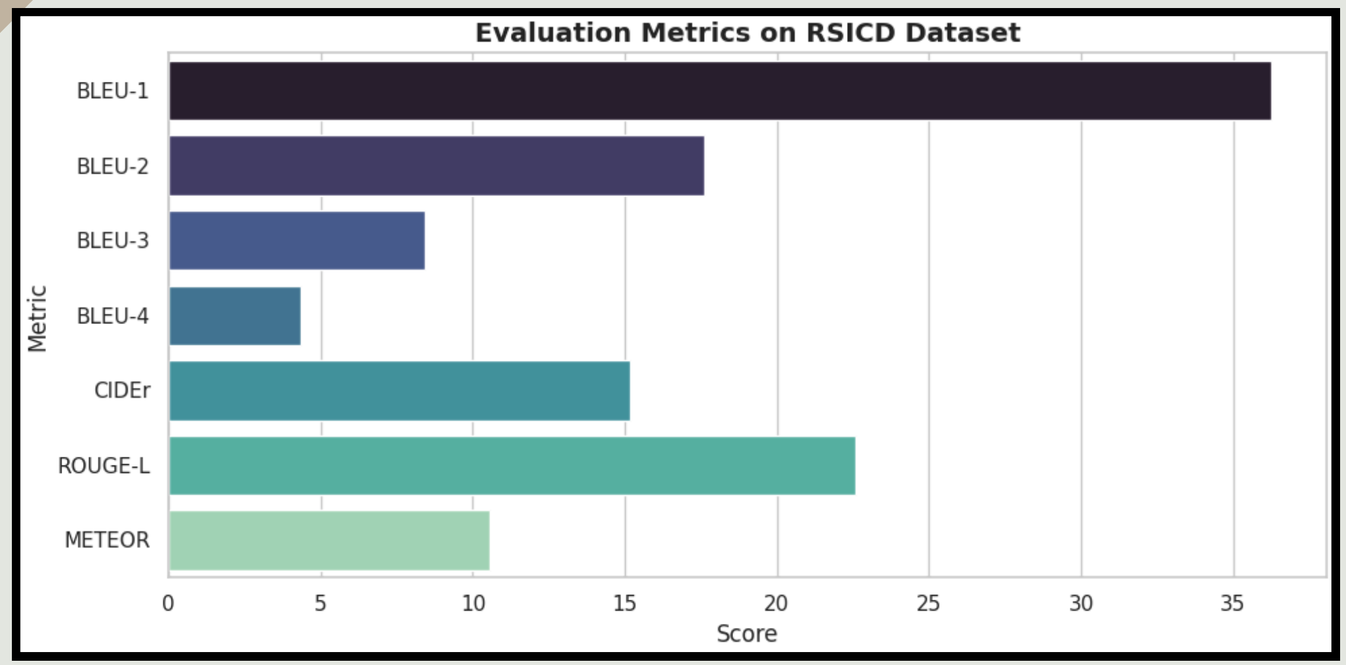
- BLIP-2 (Bootstrapped Language-Image Pre-training with Frozen Image Encoders and Large Language Models) is a state-of-the-art vision-language model designed for efficient multimodal learning. Technically, BLIP-2 innovates by leveraging two powerful, pre-trained but frozen components: a vision encoder (such as a CLIP-like image encoder) and a large language model. These are connected via a lightweight, trainable Querying Transformer (Q-Former), which bridges the modality gap between visual and textual representations

Our simulation uses

Salesforce/blip-2-opt-2.7b



BLIP-2 SIMULATION RESULTS



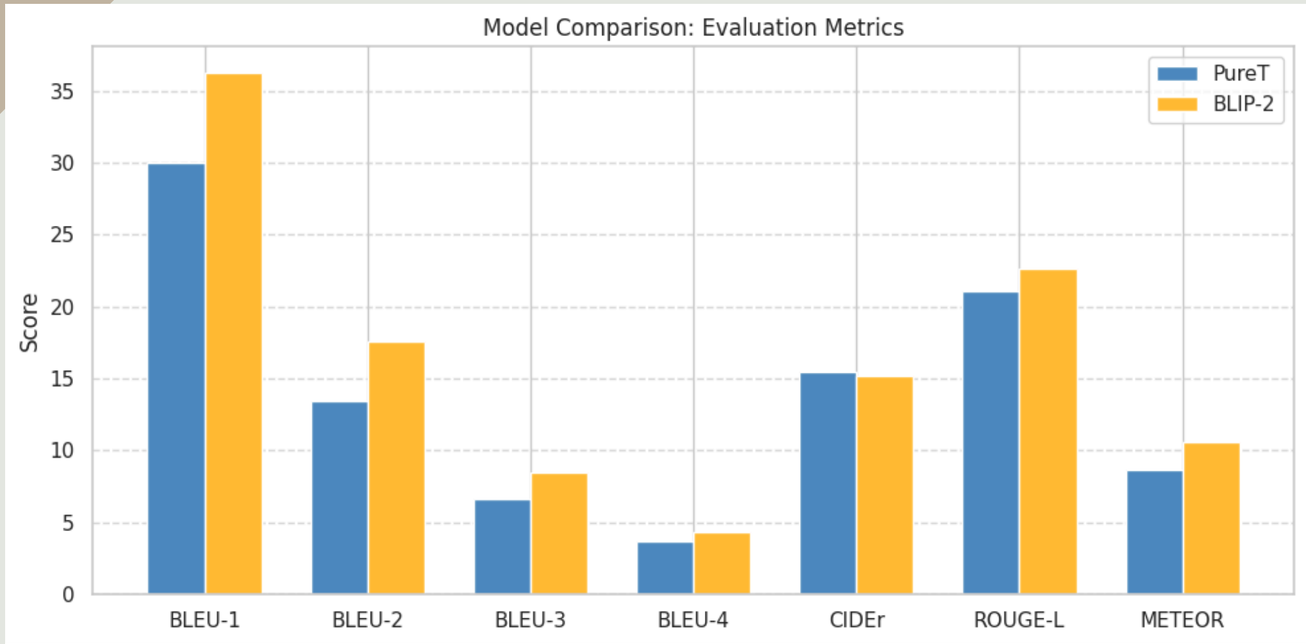
Model Parameters Overview

Metric	Value
Total Parameters	3.7B
Total GFLOPs	761.17
Avg Inference Time	844 ms

CAPTION GENERATION FOLLOWS BEAM SEARCH WITH

NUM_BEAMS=5 AND MAX_NEW_TOKENS=50

MODEL COMPARISON



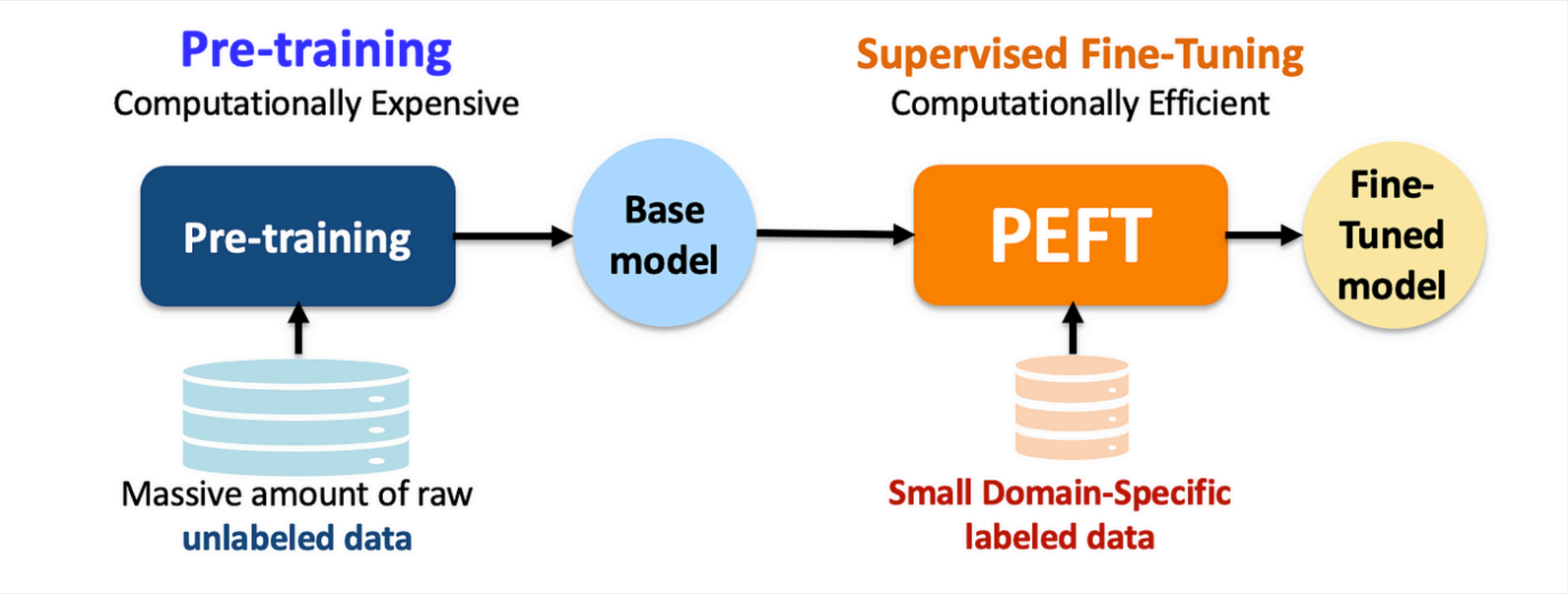
- **Captioning performance:** BLIP-2 consistently outperforms PureT across all language evaluation metrics. The most significant gains are observed in BLEU-1 (+6.26) and METEOR (+1.96), indicating improved fluency and relevance in generated captions.

- **Model Complexity:** Despite BLIP-2 offering superior performance, it comes with substantially higher computational cost. BLIP-2 has ~16x more parameters and requires ~7x more GFLOPs per inference compared to PureT. This also leads to a 2x increase in inference latency.

IMPROVEMENT STRATEGY

PEFT

- PEFT stands for Parameter-Efficient Fine-Tuning. PEFT fine-tunes a small subset of model parameters while keeping the rest frozen. PEFT is useful in resource-constrained environments as it reduces GPU memory usage and speeds up training, while enabling scalable adaptation of large models like BLIP-2 to specific datasets such as the RSICD.



METHODOLOGY-1

- The specific technique used for this task is **LoRA**, or Low Rank Adaptation. LoRA is a parameter-efficient fine-tuning method that injects small trainable low-rank matrices into existing weights, allowing large models to adapt with minimal parameter updates.

STEP-1

Augment the train data by replicating image for each of its 5 captions to generate 40k samples

STEP-2

Load LoRA Configuration for the BLIP-2 model

STEP-3

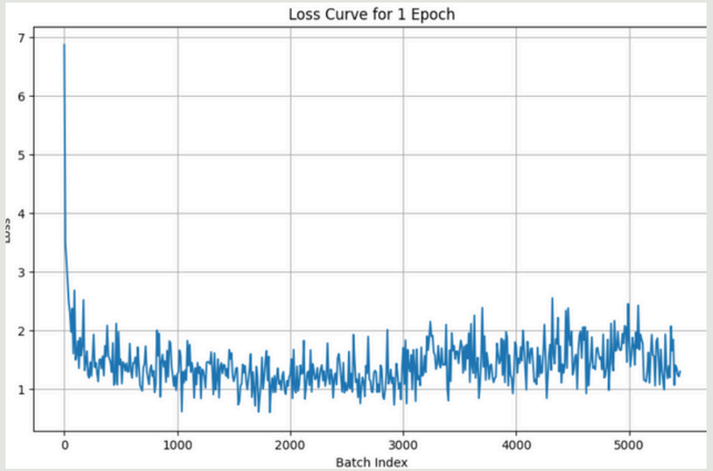
Train the model for 1 epoch on the dataset.

STEP-4

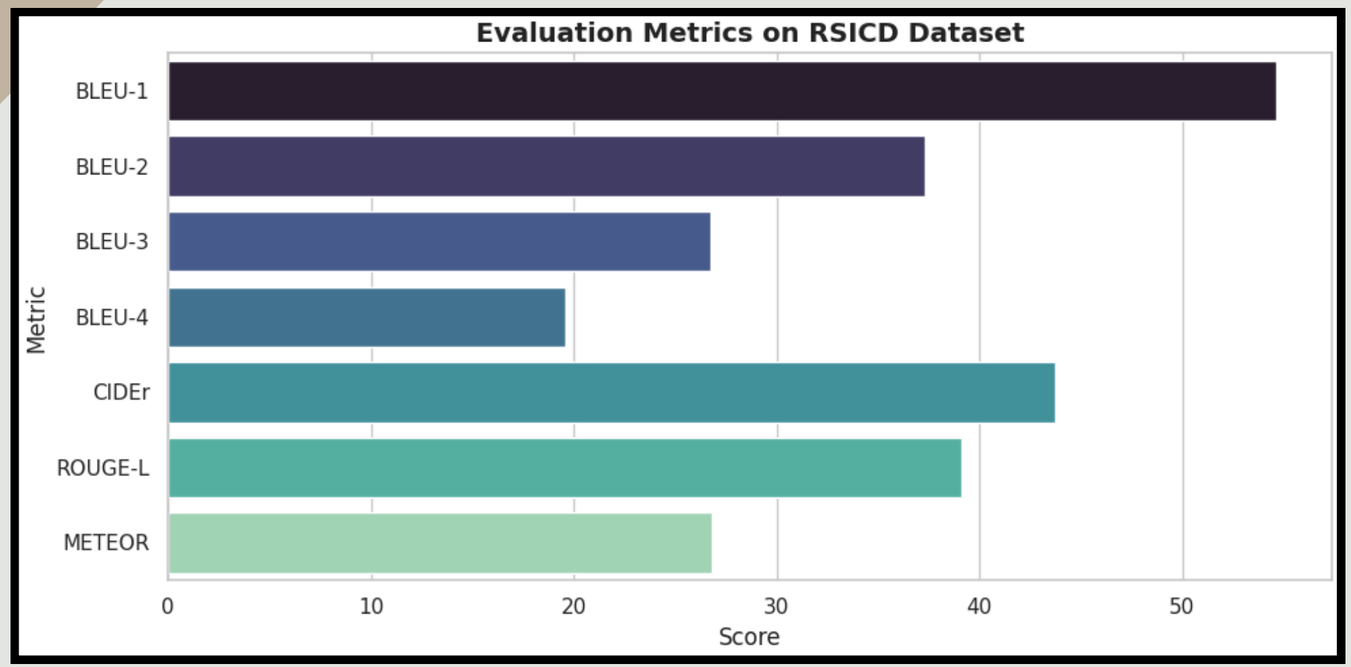
Evaluate on the test split of the data.

```
config = LoraConfig(
    r=16,
    lora_alpha=32,
    lora_dropout=0.05,
    bias="none",
    target_modules=["q_proj", "k_proj"]
)

model = get_peft_model(model, config)
```



IMPROVED RESULTS



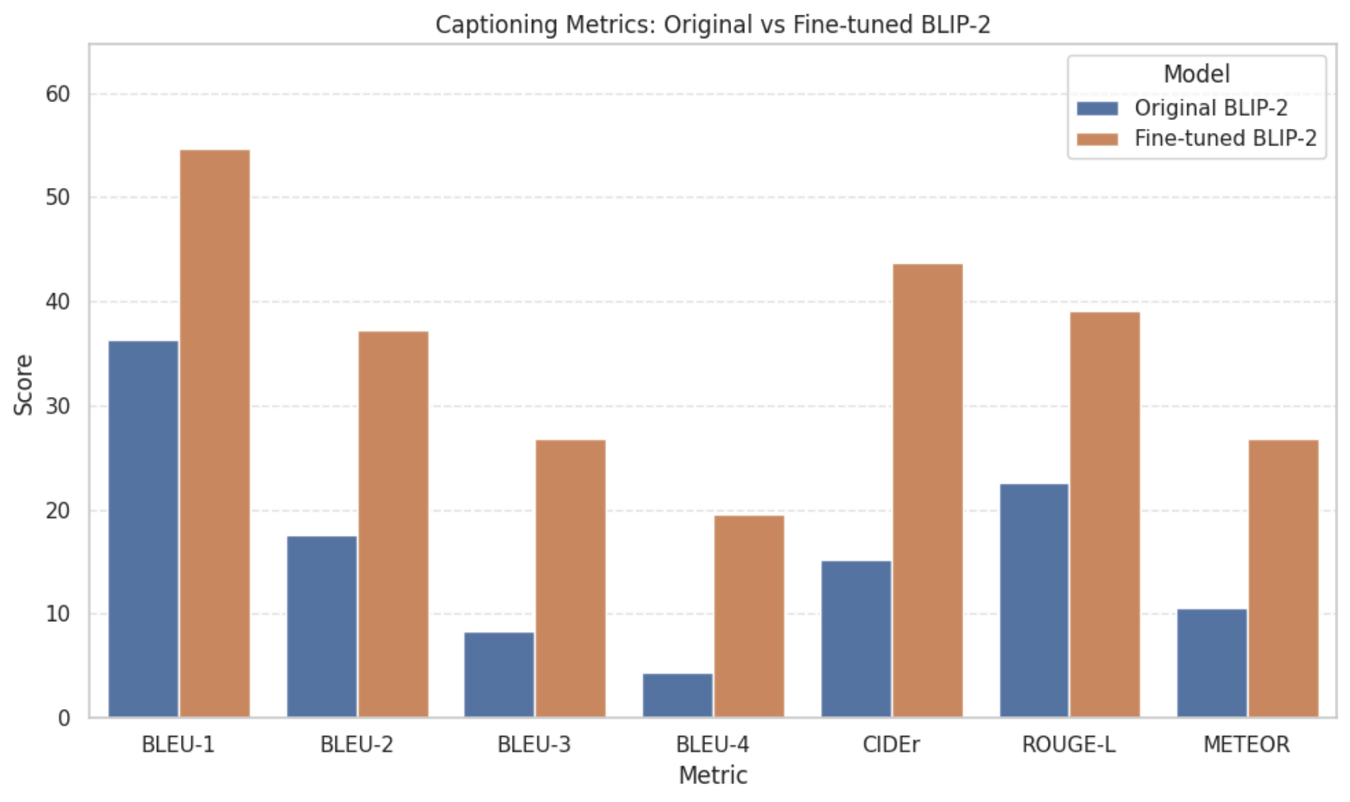
Model Parameters Overview

Metric	Value
Total Parameters	3.7B
PEFT Trainable Parameters	5.24M (0.14% of net parameters)
Total GFLOPs	792.91
Avg Inference Time	2.128 s

CAPTION GENERATION FOLLOWS BEAM SEARCH WITH NUM_BEAMS=5 AND MAX_LENGTH=50

FINAL MODEL SIZE IS 6.98 GB, HALF OF THE ORIGINAL BLIP-2 (13.95 GB) DUE TO HALF-PRECISION (FLOAT16)

IMPROVEMENTS AND TRADE-OFFS



Performance Comparison Table:

Metric	Original	Fine-tuned	% Improvement
BLEU-1	36.26	54.65	50.72%
BLEU-2	17.60	37.31	111.99%
BLEU-3	8.41	26.76	218.19%
BLEU-4	4.33	19.59	352.42%
CIDEr	15.19	43.71	187.76%
ROUGE-L	22.59	39.13	73.22%
METEOR	10.56	26.84	154.17%

WHILE THE IMPROVED MODEL HAS SIGNIFICANTLY BETTER METRICS, IT COMES AT THE COST OF HIGHER INFERENCE TIME

METHODOLOGY-2

I explored a paper titled "Parameter-Efficient Fine-Tuning of InstructBLIP for Visual Reasoning Tasks" which offers insight into different LoRA configurations used to fine-tune BLIP-2. Following are the 3 approaches stated:

- PEFT on LLM
- PEFT on Q-Former
- PEFT on both LLM and Q-Former

Although the paper uses InstructBLIP, I inspected the pruning methods should work on the BLIP-2 model too, due to similarities in architecture. Having explored approach 1 previously, which yielded improved results, I went on to explore approach 3.

PEFT on both LLM and Q-Former. Finally we apply LoRA to both the Q-Former and the LLM, using the same rank for all possible sublayers in both components. Our results show that this approach outperforms InstructBLIP for both base LLMs across both benchmarks, using fewer than 12% of trainable parameters (as depicted in Figure 2). A notable observation is that the performance gap is higher in ScienceQA than in IconQA. This discrepancy can be attributed to ScienceQA's richer

FINE-TUNING SPECIFICATIONS

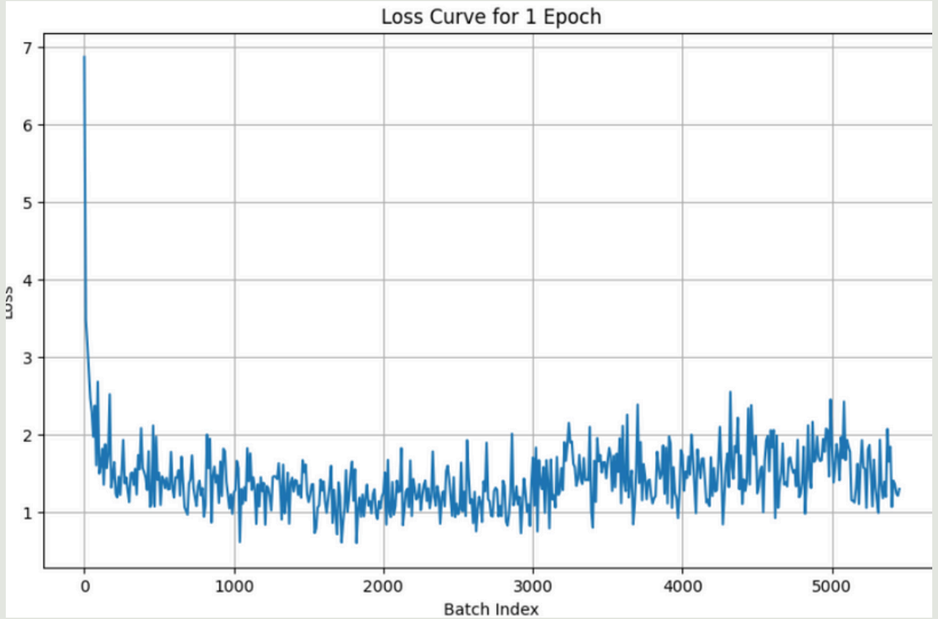
- Model used : ybelkada/blip2-opt-2.7b-fp16-sharded
- Processor used: Salesforce/blip2-opt-2.7b
- LoRA Config:

```
config = LoraConfig(
    r=16,
    lora_alpha=32,
    lora_dropout=0.05,
    bias="none",
    target_modules=["q_proj", "k_proj", "query", "key"]
)
```

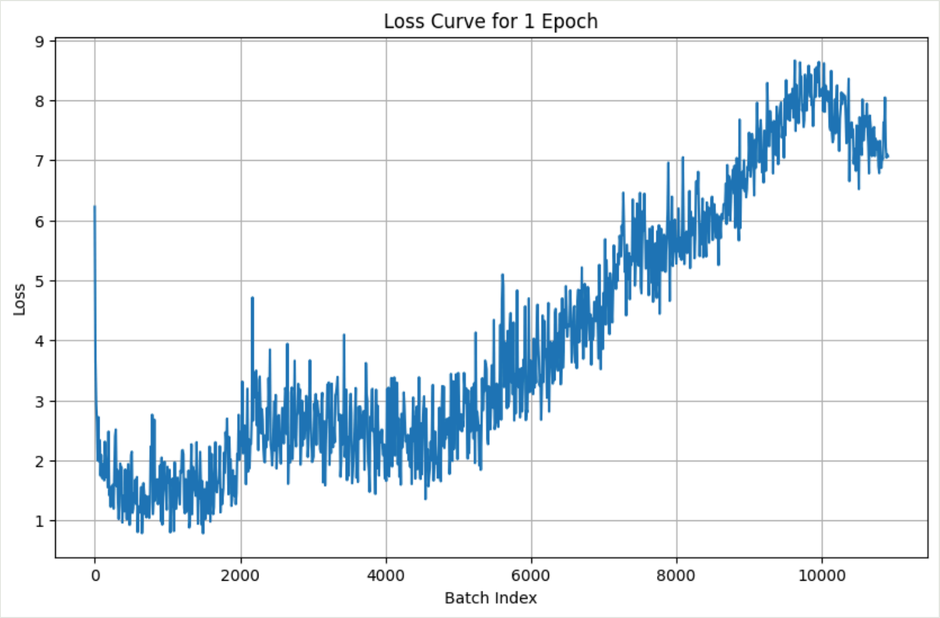
- Num_epochs = 1, with batch_size = 4

This results in around 6.1M trainable parameters, which is roughly 0.16% of the total parameters.

SIMULATION RESULTS



ORIGINAL PEFT FINE-TUNING



PEFT FINE-TUNING WITH Q-FORMER

- Although the original loss curve was noisy, the loss stabilized to between 1-2 at the end of the epoch
- But for the Q-former approach, the loss decreased only upto batch 2000, after which it steadily increases.

CONCLUSION FOR THIS METHODOLOGY

- Fine-tuning both the LLM and Q-Former at the same time did not prove to be a feasible approach on the RSICD dataset.

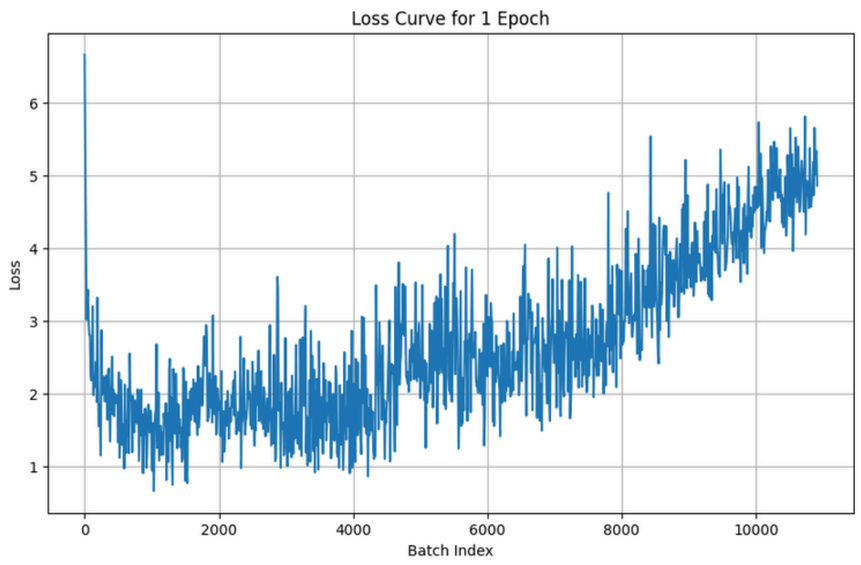
- The next approach is to try techniques such as pruning and distillation to a smaller model with similar architecture to reduce memory requirements and improve performance further.

METHODOLOGY-3
PRUNING SPECIFIC LAYERS

As part of a comprehensive study on optimizing the BLIP-2 architecture, I conducted a series of experiments focused on pruning different components of the model to reduce computational overhead while preserving performance. This involved selectively removing layers from both the Q-Former and the language model (LLM), analyzing the impact of various pruning strategies on downstream captioning performance, inference latency, and model size. Through iterative evaluation, I was able to identify configurations that significantly accelerated inference with minimal loss in output quality, thereby improving the model's suitability for resource-constrained deployment scenarios.

EXPERIMENT-1

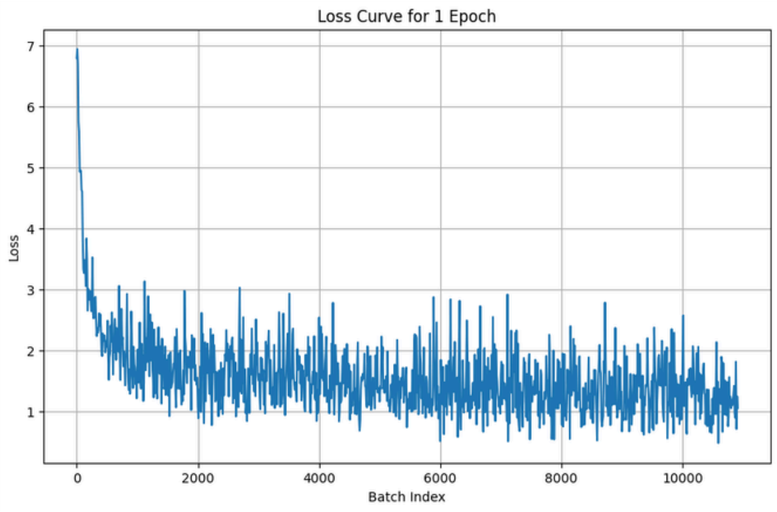
- Model used : ybelkada/blip2-opt-2.7b-fp16-sharded
- Module pruned : QFormer (retained 6 layers)
- LoRA tuning : applied only on the LLM



As inferred from previous experiments, this setup yields a noisy loss curve with the model overfitting.

EXPERIMENT-2

- Model used : ybelkada/blip2-opt-2.7b-fp16-sharded
- Module pruned : QFormer (retained 6 layers)
- LoRA tuning : applied on both LLM and QFormer

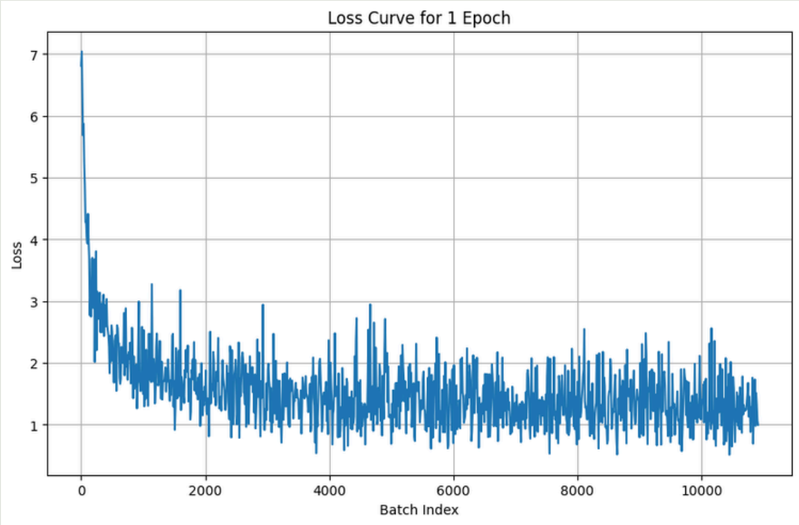


- Total parameters: 3,692,111,872
- Reduction in parameters: 1.40%
- Avg inference time: 508 ms
- GFlops per image: 750.58

- BLEU-1: 0.291
- BLEU-2: 0.17
- BLEU-3: 0.10
- BLEU-4: 0.06
- METEOR: 0.183
- ROUGE: 0.303
- CIDEr: 0.304

EXPERIMENT-3

- Model used : ybelkada/blip2-opt-2.7b-fp16-sharded
- Module pruned : QFormer (retained 8 layers)
- LoRA tuning : applied on both LLM and QFormer

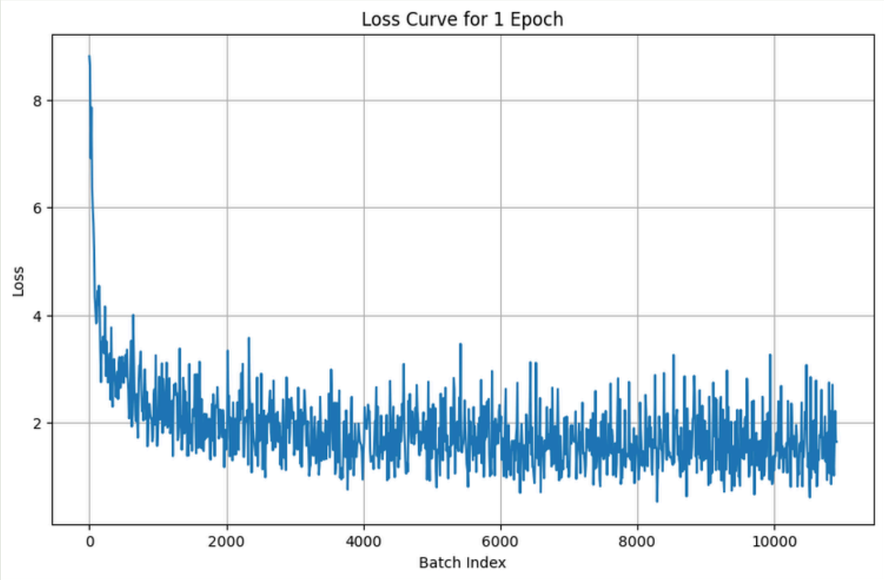


- Total parameters : 3,709,634,560
- Reduction in parameters: 0.93%
- Avg inference time: 456 ms
- GFlops per image: 750.58

- BLEU-1: 0.50
- BLEU-2: 0.312
- BLEU-3: 0.20
- BLEU-4: 0.129
- METEOR: 0.217
- ROUGE: 0.36
- CIDEr: 0.437

EXPERIMENT-4

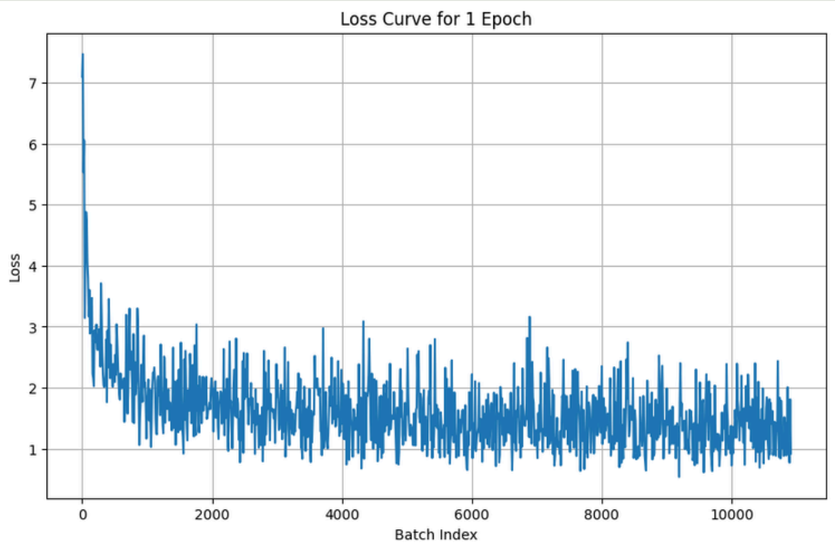
- Model used : ybelkada/blip2-opt-2.7b-fp16-sharded
- Module pruned : LLM(retained 24 layers)
- LoRA tuning : applied on LLM



- Total parameters : 3,115,268,096
- Reduction in parameters: 16.8%
- Evaluation was not conducted on this config since the sample outputs were garbage.

EXPERIMENT-5

- Model used : ybelkada/blip2-opt-2.7b-fp16-sharded
- Module pruned : LLM (retained 28 layers)
- LoRA tuning : applied on LLM

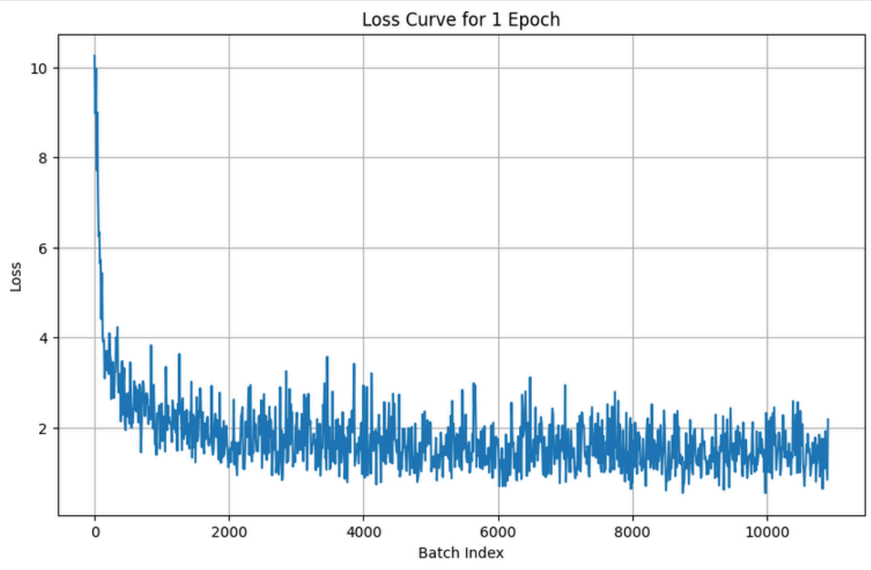


- Total parameters : 3,429,974,016
- Reduction in parameters: 8.40%
- Avg inference time: 2.15 s
- GFlops per image: 792.91

- BLEU-1: 0.47
- BLEU-2: 0.31
- BLEU-3: 0.22
- BLEU-4: 0.16
- METEOR: 0.215
- ROUGE: 0.351
- CIDEr: 0.350

EXPERIMENT-6

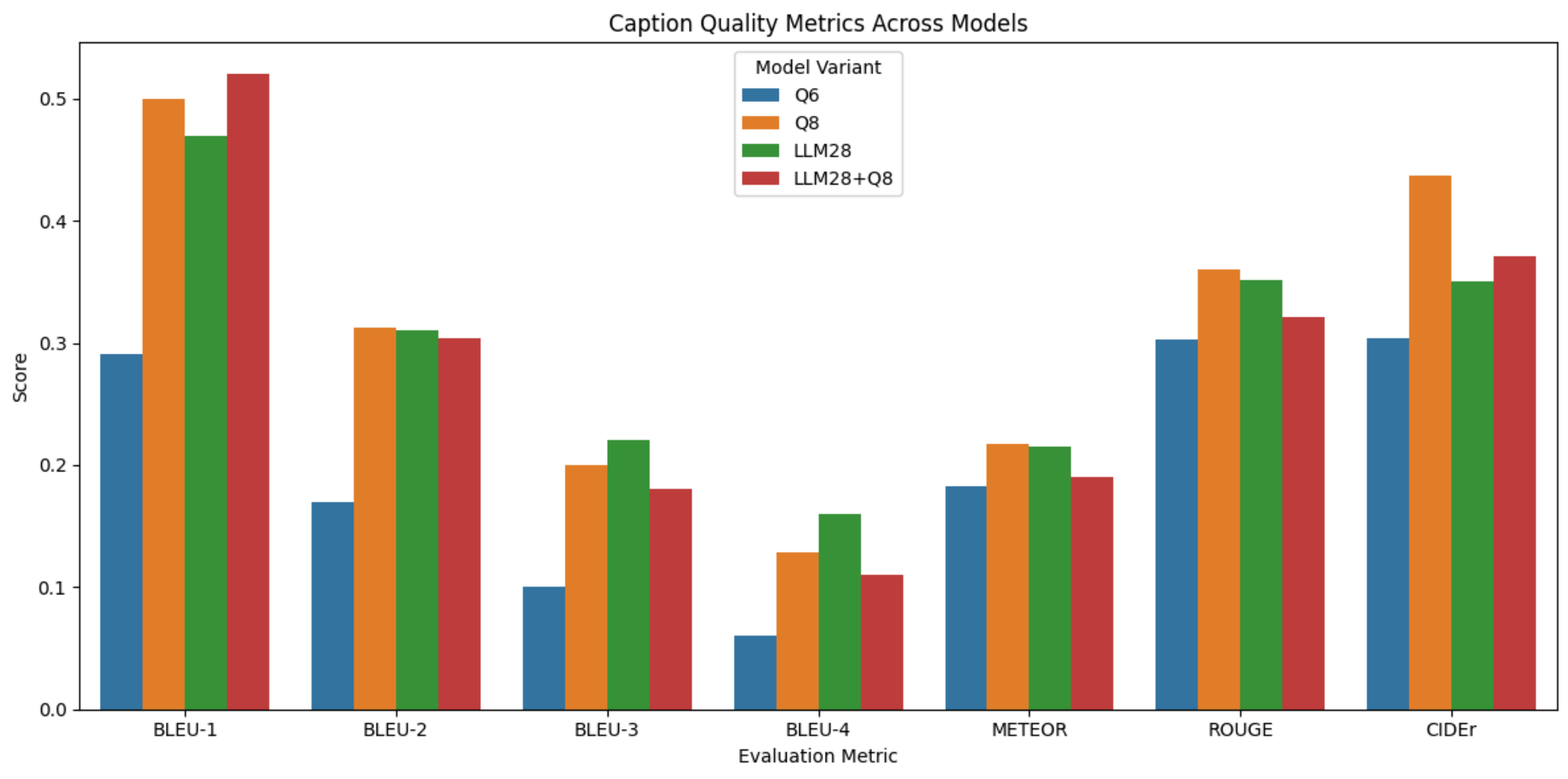
- Model used : ybelkada/blip2-opt-2.7b-fp16-sharded
- Module pruned : Both LLM (retained 28 layers) and QFormer (retained 8 layers)
- LoRA tuning : applied on both LLM and QFormer



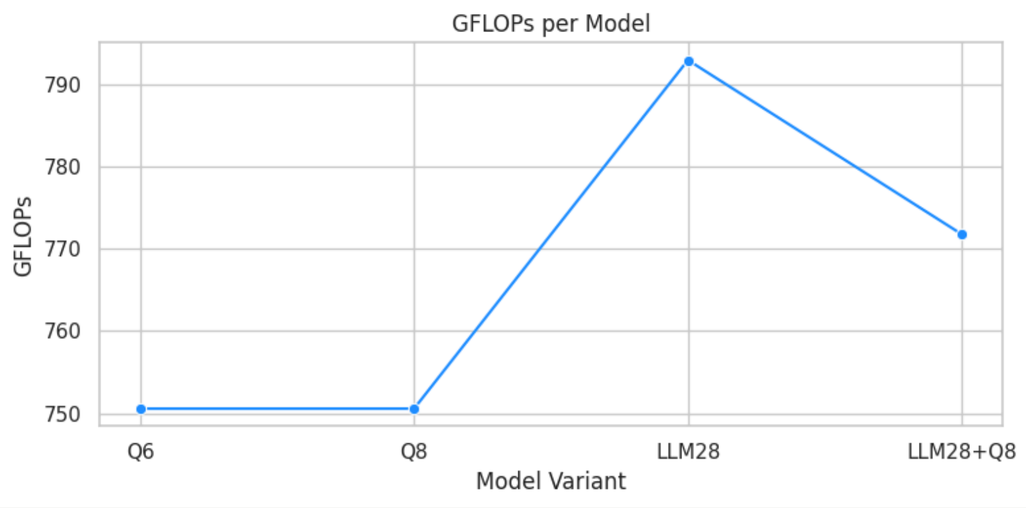
- Total parameters : 3,394,928,640
- Reduction in parameters: 9.34%
- Avg inference time: 668 ms
- GFlops per image: 771.75

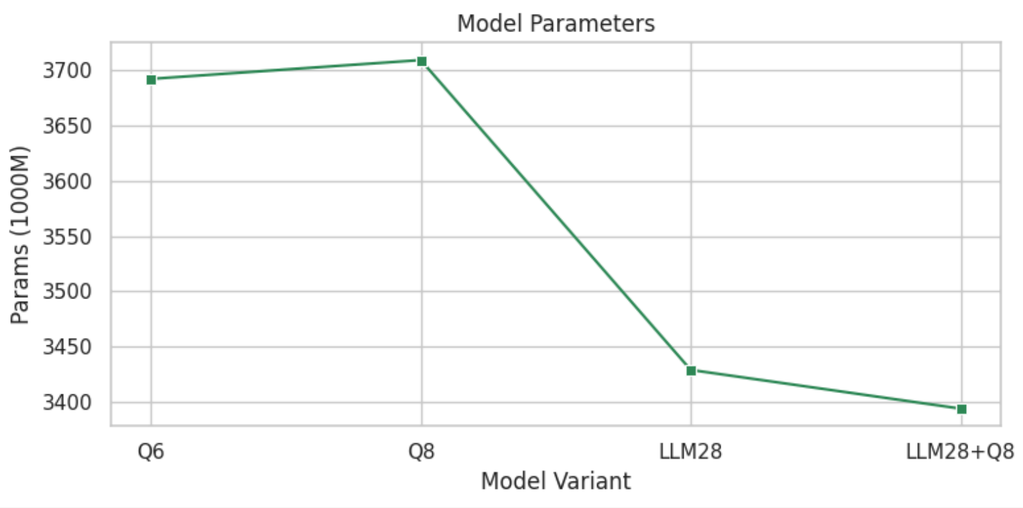
- BLEU-1: 0.52
- BLEU-2: 0.304
- BLEU-3: 0.181
- BLEU-4: 0.110
- METEOR: 0.19
- ROUGE: 0.321
- CIDEr: 0.371

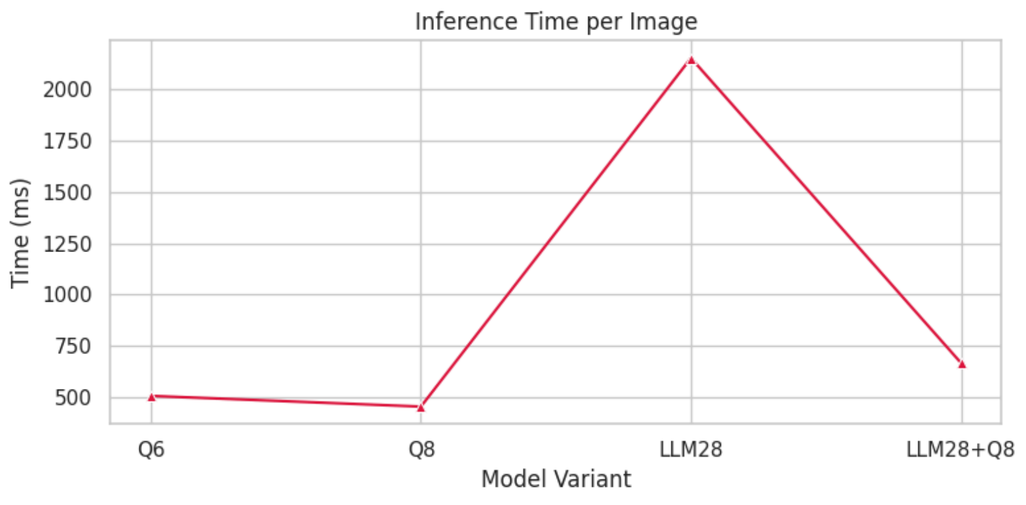
COMPARISON OF METRICS



COMPARISON OF MODEL CHARACTERISTICS







CONCLUSIONS FOR THIS METHODOLOGY

1. The QFormer has the highest contribution in model GFlops as well as inference time.
2. The LLM has the highest contribution in the model size / model parameters.
3. Q8 gives a reasonable trade-off in metrics for a significant reduction in inference time (2 secs to 0.4 secs)
4. LLM28 is the middle ground for high metrics with low parameters but almost same inference time (2 secs)
5. LLM28 + Q8 has proven to be the best combination, giving almost negligible reduction in accuracy for fast inference time (0.6 s) and lesser parameters and GFlops.

REFERENCES

- Bootstrapping Interactive Image-Text Alignment for Remote Sensing Image Captioning: https://arxiv.org/abs/2312.01191
- BLIP-2: Bootstrapping Language-Image Pre-training with Frozen Image Encoders and Large Language Models: https://arxiv.org/abs/2301.12597
- End-to-End Transformer Based Model for Image Captioning: https://arxiv.org/abs/2203.15350
- Parameter-Efficient Fine-tuning of InstructBLIP for Visual Reasoning Tasks: https://neurips2023-enlsp.github.io/papers/paper_88.pdf

Thank You