Automatic Image Captioning with Model Benchmarking and Robustness Analysis

Team Neural Navigators

Sandeep Vetcha 22CS10079   Sathvik Pratapagiri 22CS10053   Lubesh Sharma 22CS30065

Part A – Custom Encoder-Decoder Model

Methodology

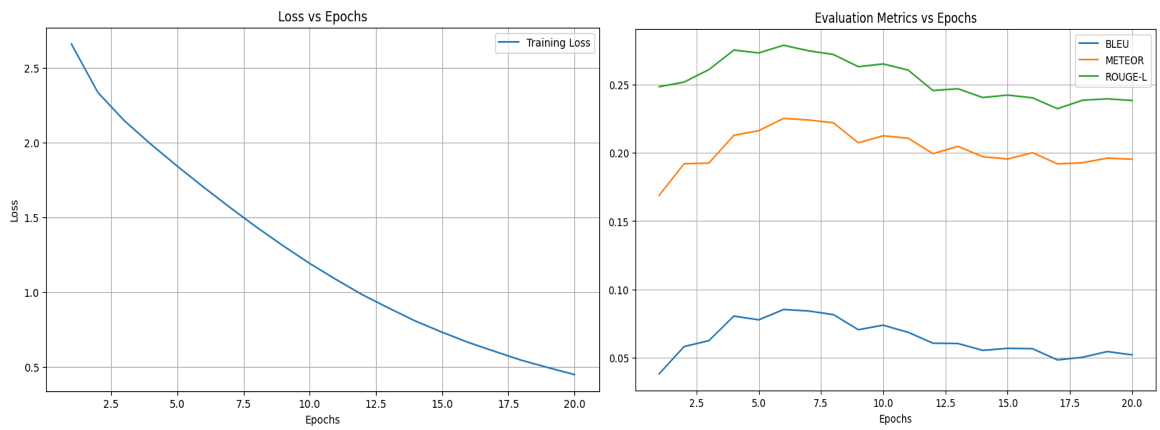
We implemented a custom image captioning model using a Transformer-based encoder-decoder architecture

- Encoder: Pretrained ViT-Small-Patch16-224
- Decoder: Transformer Decoder (based on GPT-style transformer blocks)
- Positional encodings were added to the decoder inputs.
- Teacher forcing was applied during training.
- Training:
    - Optimizer: AdamW
    - Loss: CrossEntropyLoss
    - Batch size: Tuned to fit within 15 GB GPU limit (Colab T4)
    - Epochs: 20
    - Dataset: Provided image-caption dataset

Zero-Shot Baseline: SmolVLM was used without any fine-tuning, using attn_implementation='eager' to bypass compatibility issues with flash attention.

Models	BLEU Score	METEOR Score	ROUGE-L Score
SMOLVLM	0.0066	0.1299	0.0790
Custom	0.0527	0.1947	0.2385

GRAPHS



From graphs we can conclude that suitable number of epochs to run are 7 for best model

---

Part B – Occlusion Robustness Analysis

Methodology

Step 1: Patch-wise Occlusion

- Each image is divided into 16×16 patches, matching the patch size of the ViT encoder.
- A percentage of patches (10%, 50%, or 80%) is selected randomly and their pixel values are replaced with black (0, 0, 0).
- This simulates varying levels of information loss in a structured way

```
occluded_image = occlude_image(original_image, mask_percentage=50)
```

Step 2: Caption Generation

- For each occlusion level:
  - Both SmolVLM (zero-shot) and Custom model (fine-tuned) generate captions.
  - The captions are compared against the ground truth using:
    - BLEU
    - ROUGE-L
    - METEOR

Step 3: Performance Degradation Analysis

For each occlusion level, compute:

Metric Degradation = Metric(after occlusion) - Metric(before occlusion)

- A more robust model would show less degradation in scores.

Step 4: Data Logging for Part C

- Save the original caption, generated caption, and perturbation level (10/50/80) into a .csv file for use in training the BERT classifier.

Architecture Diagram (Descriptive Text)

You can draw the below as a flowchart or describe in the report like this:

1. Input Image → Patch Splitter → Random Patch Masking (blackout)
2. → Occluded Image → Captioning Model (SmolVLM / Custom)
3. → Generated Caption
4. → Metric Evaluator (BLEU, ROUGE-L, METEOR)

Repeat for 10%, 50%, 80% occlusion levels.

SmolVLM Performance:

Occlusion %	**BLEU**		**ROUGE-L**		**METEOR**	
	SMOLVLM	**CUSTOM**	**SMOLVLM**	**CUSTOM**	**SMOLVLM**	**CUSTOM**
10%	**0.0101**	**0.0202**	**0.1318**	**0.2124**	**0.0861**	**0.1371**
50%	**0.0033**	**0.0174**	**0.1031**	**0.2018**	**0.0573**	**0.1283**
80%	**0.0006**	**0.0141**	**0.0744**	**0.1928**	**0.0379**	**0.1203**

Part C – Caption Source Classification using BERT

Methodology

Step 1: Dataset Construction

- From Part B, for each test image and occlusion level, we collected:
    - The ground truth caption
    - The model-generated caption (from both SmolVLM and Custom)
    - The occlusion level (10, 50, or 80)
- We structured the input as:
  Input Text: <original_caption> <SEP> <generated_caption> <SEP><occlusion_level>
  Label: SmolVLM or Custom
- This dataset was saved as a .csv file and used to train the classifier.

Step 2: Model Architecture

- We used bert-base-uncased from HuggingFace as the text encoder.
- The encoder's final CLS token output is passed through a small feedforward head:
  BERT (CLS) Output → Dropout → Linear Layer (768 → 128) → ReLU → Linear Layer (128 → 2)
- Final output: Logits for binary classification ([SmolVLM, Custom])

Step 3: Training Setup

- Loss Function: CrossEntropyLoss
- Optimizer: AdamW
- Learning Rate: Tuned (default: 2e-5)
- Batch Size: 16
- Epochs: Trained until validation F1 stopped improving
- Early Stopping: Based on F1 score on validation set

Step 4: Data Splitting

- Dataset split by images (no overlap):
    - 70% Train
    - 10% Validation
    - 20% Test

Evaluation Results (Test Set)

Metric	Value
Accuracy	97.50%
Precision	97.62%
Recall	97.50%
F1 Score	97.50%

Performance by Occlusion Level

Occlusion %	Accuracy
10%	97.33%
50%	97.59%
80%	97.59%