DLCV HW3

NTU, FALL 2024

B10901151 林祐群
指導教授： 王鈺強
2024年11月19日

Problem 1: Zero-shot Image Captioning with LLaVA

1. Paper Reading(3%): Please read the paper "Visual Instruction Tuningand briefly describe the important components (modules or techniques) of LLaVA.

The LLaVA model in *Visual Instruction Tuning* combines vision and language understanding through three main components:

1. Visual Encoder: LLaVA uses a pre-trained CLIP visual encoder, which maps images into a feature space that aligns with text, allowing for multimodal understanding.

2. Projection Layer: A linear projection layer connects the CLIP-generated visual embeddings to the language model. This layer adapts visual features into a compatible format for language processing, enabling the language model to interpret visual information.

3. Language Model Integration: The transformed visual embeddings are passed to an LLM, which has been instruction-tuned to process these embeddings alongside natural language prompts. This setup allows the LLM to generate text responses based on both visual and textual cues.

4. Instruction Tuning: LLaVA is fine-tuned with a specialized dataset containing vision-language instruction pairs. These pairs include examples of questions about images, visual descriptions, and responses, which refine the model's ability to generate coherent answers to visual questions in an instruction-following format.

2. Prompt-text analysis (6%): Please come up with two settings (different instructions or generation config). Compare and discuss their performances.

1. Config one

- **Instruction:** "Describe this image using one or more simple sentences."
- **Generation Config:** max_new_tokens=30, do_sample=True, num_beams=5, temperature=0.5, top_p=0.9, top_k=50, min_length=10, num_return_sequences=3.
- **Performance:** CIDEr: 1.18, CLIP: 0.78

2. Config two

- **Instruction**: "Describe the primary subject and background in the image, including any key actions and spatial relationships."
- **Generation Config**: Identical to Config 1.
- **Performance:** CIDEr: 0.07, CLIP: 0.77
- **Analysis**: This instruction's complexity appears challenging, which dilutes the model's focus, lowering CIDEr scores. Despite the same generation setup with config one, the prompt's specificity leads to the lower CIDEr score, while the CLIP score remains high.

3. Config three

- **Instruction:** "Describe this image using one or more simple sentences."
- **Generation Config:** max_new_tokens=100, do_sample=False, num_beams=3, min_length=10, num_return_sequences=3.
- **Performance:** CIDEr: 1.12, CLIP: 0.79
- **Analysis**: Disabling sampling and using a larger token limit (100) leads to less variability but enables more exhaustive descriptions. The slightly lower CIDEr score than Config one indicates slight over-generation, where extra tokens may not directly improve relevance. The CLIP score benefits slightly, likely because the non-sampled outputs align more predictably with visual content.

4. Config four

- **Instruction**: "Describe the primary subject and background in the image, including any key actions and spatial relationships."
- **Generation Config:** Same as Config 3.
- **Performance:** CIDEr: ~0, CLIP: 0.80
- **Analysis**: This detailed instruction with an extended token limit (100) likely results in redundant or unfocused content, significantly dropping CIDEr scores. The model may struggle to produce a tightly relevant response with such an open-ended instruction. However, the CLIP score remains strong, suggesting the generated descriptions maintain visual correspondence, even if they are less relevant.

5. Summary of Hyper parameters Effects:

- **Sampling (`do_sample=True`) with shorter tokens** favors concise, coherent descriptions, which enhances CIDEr.
- **Beam count** balances relevance and diversity; higher beams improve quality at the cost of generation speed.
- **Temperature, `top_p`, and `top_k`** control diversity, but their effectiveness decreases with long instructions due to increased output variability.
- **Extended token limits** (like `max_new_tokens=100`) can dilute content relevance, particularly with complex instructions, affecting CIDEr.

Problem 2: PEFT on Vision and Language Model for Image Captioning

1. Report your best setting and its corresponding CIDEr & CLIPScore on the validation data. Briefly introduce your method. (TA will reproduce this result) (5%)

CIDEr Score: 0.8136990797662019

CLIP Score: 0.6901151275634766

Setting:

I. LoRA Parameters:

1. Rank: 16
2. Alpha: 32

II. Beam Search Parameters:

1. Beam Width: 6
2. Max Length: 25
3. Temperature: 0.8
4. Length Penalty: 0.9
5. Diversity Penalty: 0.5

Method:

The method in the problem combines a Vision Transformer (ViT) encoder and a LoRA-enhanced GPT-based decoder for image captioning:

1. **Vision Encoder:** A pre-trained ViT ('vit_base_patch16_224') extracts feature representations of the input image. Only the final transformer block and feature resizing layer are fine-tuned.
2. **LoRA Decoder:** The decoder utilizes LoRA (Low-Rank Adaptation) to inject trainable low-rank matrices into linear layers. The decoder generates captions conditioned on visual features and textual context.

3. **Beam Search Decoding:** Beam search with temperature scaling, length penalty, and diversity adjustment generates captions by balancing exploration and exploitation during decoding.
4. **Optimization:** LoRA's fine-tuning strategy focuses updates on low-rank parameter spaces. A combination of AdamW optimizer, learning rate scheduling, and gradient clipping ensures stable training.
5. **Evaluation:** CIDEr evaluates semantic similarity with ground-truth captions, while CLIPScore assesses alignment between visual and textual modalities.

2. Report 2 different attempts of LoRA setting (e.g. initialization, alpha, rank…) and their corresponding CIDEr & CLIPScore. (5%, each setting for 2.5%)

Setting 1:

Alpha = 16, Rank = 64 (3.77M trainable parameters)

Initialization:

nn.init.normal_(self.visual_token_embedding, mean=0.0, std=0.02)

Training epochs: 400

CIDEr Score = 0.8458549252404385

CLIP Score = 0.6854154968261719

Setting 2:

Alpha = 8, Rank = 16 (2.18M trainable parameters)

Initialization:

nn.init.normal_(self.visual_token_embedding, mean=0.0, std=0.02)

Training epochs: 250

CIDEr Score = 0.8196278972036857

CLIP Score = 0.6890658771304531

Problem 3: Visualization of Attention in Image Captioning

1. Given five test images ([p3_data/images/]), and please visualize the **predicted caption** and the corresponding series of **attention maps** in your report with the following template: (20%, each image for 2%, you need to visualize 5 images for both problem 1 & 2)

Problem 1:

In the implementation of the self-attention map of LLaVA, I discovered that the difference between attention map of different text token is relatively minor compared to globally significant attention. While analyzing attention maps in detail, it's still able to find out the different and the correlation between text tokens and attention. For example, difference among attention maps of caption for bike.jpg is more visually discoverable.

I. Caption:

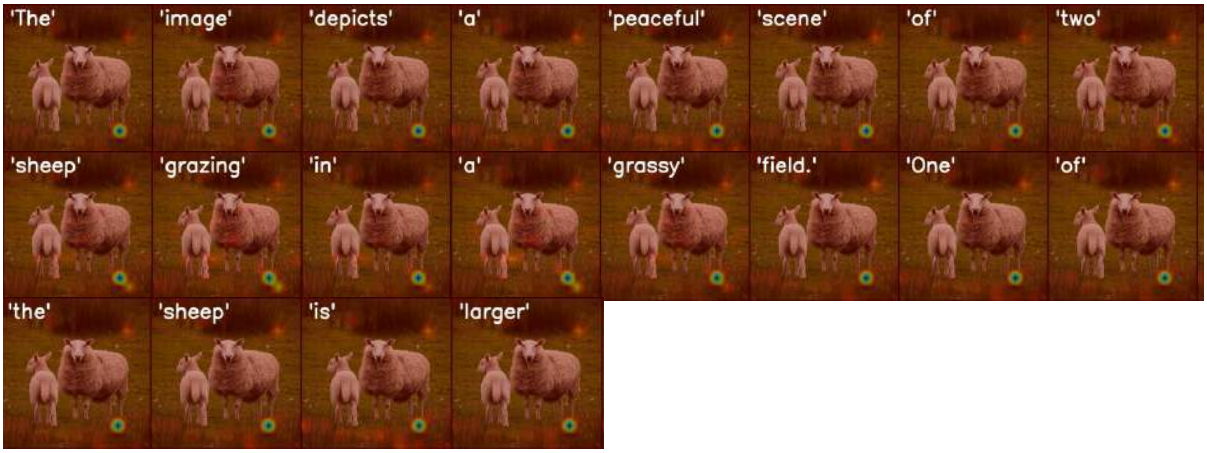
The image depicts a woman riding a bicycle down a city street. She is holding an umbrella to protect herself

II. Caption:

The image features a young girl sitting in a chair, holding a slice of pizza in her hand. She appears to be enjoying her meal

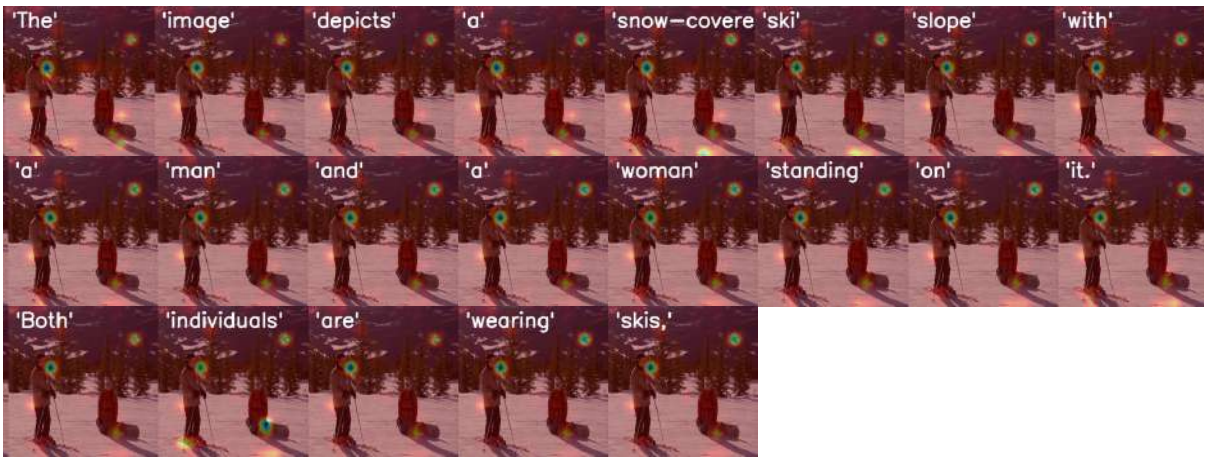
III. Caption:

The image depicts a peaceful scene of two sheep grazing in a grassy field. One of the sheep is larger



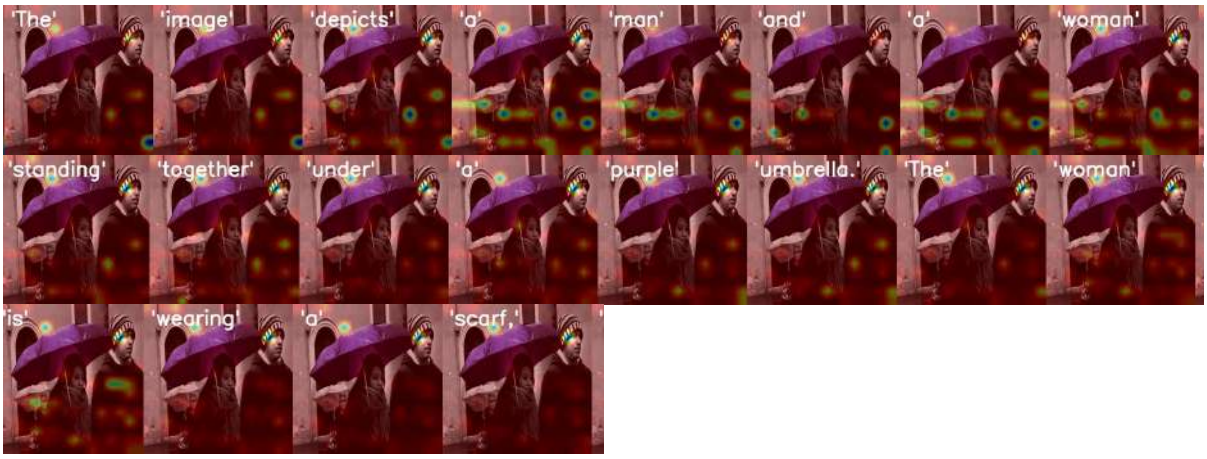
IV. Caption:

The image depicts a snow-covered ski slope with a man and a woman standing on it. Both individuals are wearing skis



V. Caption:

The image depicts a man and a woman standing together under a purple umbrella. The woman is wearing a scarf



Problem 2:

<Start> A couple walking a

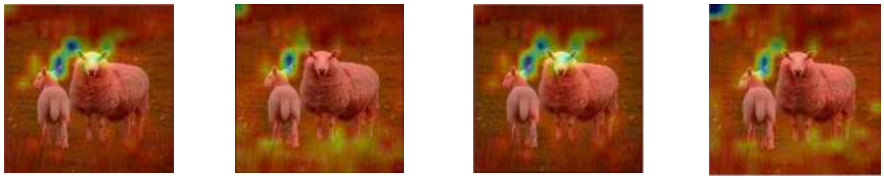
cart on the street <|endoftext|>

<Start> A smiling layer with

two pizzas <|endoftext|>

<Start> A couple of sheep

stand on grass <|endoftext|>



<Start> Three skiers pose on

a mountain <|endoftext|>



<Start> A woman walks with





a pink umbrella <|endoftext|>




2. According to **CLIPScore**, you need to:

    1. visualize top-1 and last-1 image-caption pairs
    2. report its corresponding CLIPScore

in the validation dataset of problem 2. (3%)

Top-1 image-caption pairs:

"caption":

"A red fire hydrant in front of a house."

“CLIP Score": 0.9669598191976547

Last-1 image-caption pairs:

"caption":

"A large white and blue bus on a street."

“CLIP Score": 0.3058889508247376

3. Analyze the predicted captions and the attention maps for each word according to the previous question. Is the caption reasonable? Does the attended region reflect the corresponding word in the caption? (3%)

With the following visualization:

Top-1 image-caption-attention pairs:

<Start> A red fire hydrant in front

of a house <|endoftext|>

For top-1 case, the caption is concise and reasonable. It captures the visually-significant concept with straight forward description. Analyzing the corresponding attention map, it displays that the attention mechanism does not precisely indicate the choice of words. This, in my opinion, is because of the way we train the model with frozen vision encoder. However, it still shows some guidance in the attention map of the word “red”, “house”, and “front” for instance. The focus of attention mechanism and the words being chosen by beam_search are correlated in spatial either locally or globally.

Last-1 image-caption-attention pairs:

<Start> A large white and blue bus

on a street. <|endoftext|>

As for last-1 case, the caption hugely deviates from the visual concept depicted in the image. I believe this is resulted from the frozen vision encoder and the possibly overfitting of decoder with LoRA fine-tuning. We can discover that the window frames side by side is similar to the windows of a bus to some extent, though the predicted color blue is quite wrong. We can notice from the attention map and corresponding text token that the attention is more fixed compared with the top-1 case ; while both cases' attention map are not really centralize on each specific visual concept, the performance of CLIP scores are a lot different. Since I utilize base vision transformer, the vision concept captured might not be detailed enough for the model to literally learn a great embedding for prediction.

References

1. Visual Instruction Tuning:

https://arxiv.org/abs/2304.08485

2. LLaVA Hugging Face:

https://huggingface.co/llava-hf/llava-1.5-7b-hf

3. CIDEr Score:

https://www.cv-foundation.org/openaccess/content_cvpr_2015/papers/Vedantam_CIDEr_Consensus-Based_Image_2015_CVPR_paper.pdf

4. LoRA utilization:

https://github.com/microsoft/LoRA

5. Image caption with attention map:

https://www.tensorflow.org/text/tutorials/image_captioning

6. LLaVA attention map forum:

https://github.com/haotian-liu/LLaVA/issues/1080

7. LoRA parameters finding:

https://www.determined.ai/blog/lora-parameters

8. Image caption with post-processing:

https://ieeexplore.ieee.org/document/10138597

9. ChatGPT
10. Claude
11. Tabnine AI