Application of Generative Visual Question Answering in Power Transmission System Inspection

Duy Khoi Tran

June 6, 2024

1 Introduction

This project is for the completion of the Special curriculum FYS-8805 Generative AI. The topic of this project is the implementation of vision-language models to perform visual question answering on two tasks: Insulator disc counting and pole section matching.

A model capable of disc counting is a beneficial component in an automatic visual inspection system of transmission and distribution systems. This model can give us insight into what type of insulator is being examined and can assist us in pinpointing defects.

In order to have a fine-grained annotation of poles and masts, instead of a bounding box for an entire pole, we obtained multiple bounding boxes ground-truths for every pole section (See Figure. 2). These sections are divided from a utility pole via components such as cross arms. However, we did not specifically link the sections belonging to the same pole together and had problems connecting them as a downstream task, especially when there are multiple poles in the same image.

Although the two tasks above can be solved via discriminative directions (classification models, clustering, etc...). We would like to explore whether the recent advancements in Generative AI can help solve these efficiently. This is because it is shown that many Generative AI models can be fine-tuned to operate well in different domains using limited data [HSW+21].

Thus, in this project, we attempted to fine-tune a couple of vision-language models to perform vision question answering on these two tasks. The resulting models can be used in the demos on github.

2 Methodology

2.1 Models

Due to the limitations in hardware, we had to choose smaller vision-language models compared to the remaining state-of-the-art options. The two models we chose are Idefics2-8B and MiniCPM-V-2.

Idefics2-8B [Fac23], an advanced 8 billion parameter vision-language model by Hugging Face, significantly enhances multimodal tasks like visual question answering and OCR. With open licensing, it builds on Idefics1, introducing improvements such as a reduced dependency on specific input resolution and a simplified architecture. Trained on diverse datasets including Wikipedia and LAION-COCO, it competes with larger models and is easily fine-tunable via the Hugging Face Hub. Key advancements include enhanced OCR and handling of large-resolution images, making Idefics2 a versatile tool for various multimodal applications.

MiniCPM-V 2.0 [Ope23b] is a state-of-the-art multimodal language model. With just over 2B parameters, it excels in OCR and comprehensive understanding tasks. Developed by OpenBMB, it features efficient end-side processing, making it suitable for deployment in various applications requiring strong visual and textual comprehension. The model's design focuses on balancing performance and computational efficiency, offering robust capabilities in processing and understanding complex multimodal inputs. Its architecture and training methodology enable high accuracy in recognizing and interpreting visual and textual data, positioning MiniCPM-V 2.0 as a powerful tool in the field of multimodal AI.

These two recent models boast superior performance over models of similar size and even larger models of up to 30B parameters at several vision question answering tasks [Fac23, Ope23b, Ope23a]. Thus, they are promising candidates for our attempt to fine-tune them for insulator disc counting and pole section matching. In this project, we use MiniCPM-V-2 only for insulator disc counting because, from our qualitative evaluation, we could not produce a MiniCPM-V-2 model that performs well for pole section matching.

2.2 Low-Rank Adaptation

To efficiently fine-tune these models, we use the popular Low-Rank Adaptation (LoRA) methods [**?**]. It is an efficient method for fine-tuning large language models. Instead of updating all parameters during adaptation, LoRA introduces trainable low-rank matrices into the Transformer architecture, significantly reducing the number of trainable parameters and memory requirements. This method maintains the original model's quality while minimizing computational overhead and inference latency. LoRA is versatile, performing well across various models such as RoBERTa, DeBERTa, GPT-2, and GPT-3, and facilitates easier deployment and task-switching by only modifying the low-rank matrices.

3 Implementation

3.1 Data

Figure 1: Examples of data for training and test insulator disc counting

We procure two VQA datasets for insulator disc counting and pole section matching. Each dataset has 250 data points that are split into train and validation sets in the ratio of 80 : 20. Each data point is in the form of {*image*, *query*, *answer*} where *image* is an image cropped using the bounding box around an insulator or an image featuring an electrical tower or pole prominently. *query* and *answer* are the query and answer strings that are encoded by the language module of either Idefics2-8B or MiniCPM-V-2.

More specifically, for insulator disc counting, the query is set specifically to *"The picture prominently displays an insulator. Provide a number only. How many discs are there in the insulator?"* and the answer is set to a single integer of the number of discs on the insulator. In the attempts to improve the performance, we also tried to diversify the queries by using ChatGPT-4o [**?**] to rephrase the initial query or to translate it to different languages multiple times.

For pole section matching, the bounding boxes of the pole sections are drawn onto the image in different colors, the query is set to *In this image we have colored bounding boxes. Group the bounding boxes that cover the same part of the powerline towers.*, and the answer has the form *Group 1 contains the bounding boxes colored in* {*color0_0*}, {*color0_1*}, ..., *and* {*color0_n*}. *Group 2 contains the bounding boxes colored in* {*color1_0*}, {*color1_1*}, ... *Group M...* We settled with this type of formulation after trying several other options such as giving the actual number coordinates of the bounding boxes of in the query and ask which pole sections belong to each other. The other option was to iteratively go through each bounding box, all of which are drawn onto the image, and ask which of the remaining bounding boxes are connected to the one in question. We observe that our proposed formulation gives a more consistent result.

Figure 2: Examples of data for training and test pole section matching

3.2 Training configurations

The training code for MiniCPM-V-2 is based on instructions provided from github. The training code for Idefics2 is created with the help of this example code.

The MiniCPM-V-2 models are trained with standard LoRA. Through some hyperparameter search, we found the following settings to produce the best performing models: LoRA rank at 16, LoRA alpha at 32, and gradient accumulation steps at 32 with batch size 4 (due to hardware limitation).

To be able to train Idefics-8B using our hardware (a RTX 3080 24GB) we have to use QLoRA [DPHZ23], which is an extension of LoRA that reduces memory requirements and increases computational efficiency. It achieves that by quantizing weights stored in memory to the 4-bit NormalFloat format as well as the quantization constant (double quantization). Furthermore, computations within the models are half-precision (16-bit float).

4 Results

As a result, through some experimentation, we produce a fine-tuned MiniCPM-V-2 and a fine-tuned Idefics2-8B model for insulator counting as well as a fine-tuned Idefics2-8B.

According to Table 1, after using the best training settings (hyperparameters and dataset variations), MiniCPM-V-2 was able to achieve considerable improvement over the default variants. When training with one query for every data point, the performance does not improve at all. By translating that one query into multiple languages and randomly sampling queries for each data point we see a jump to 36.4%. It should be noted that increasing LoRA rank to 32 and above, which further increases the number of trainable parameters, does not provide much help to the performance.

Model	Accuracy	Mean differences
Without fine-tuning	21%	3.38
One query	20%	3.93
Multiple languages	36.4%	2.45
LoRA rank 32	23.2%	3.48
Best	46%	2.19

Table 1: Performance of MiniCPM-V-2 across multiple training settings for insulator disc counting.

	Accuracy	Mean differences
w/o QLoRA adapter	16%	4.36
w/ QLoRA adater	72%	1.64

Table 2: Performance of Idefics2-8B with and without finetuning for insulator disc counting.

	Grouping accuracy
w/o QLoRA adapter	17.1%
w/ QLoRA adater	57.14%

Table 3: Performance of Idefics2-8B with and without finetuning for pole section matching.

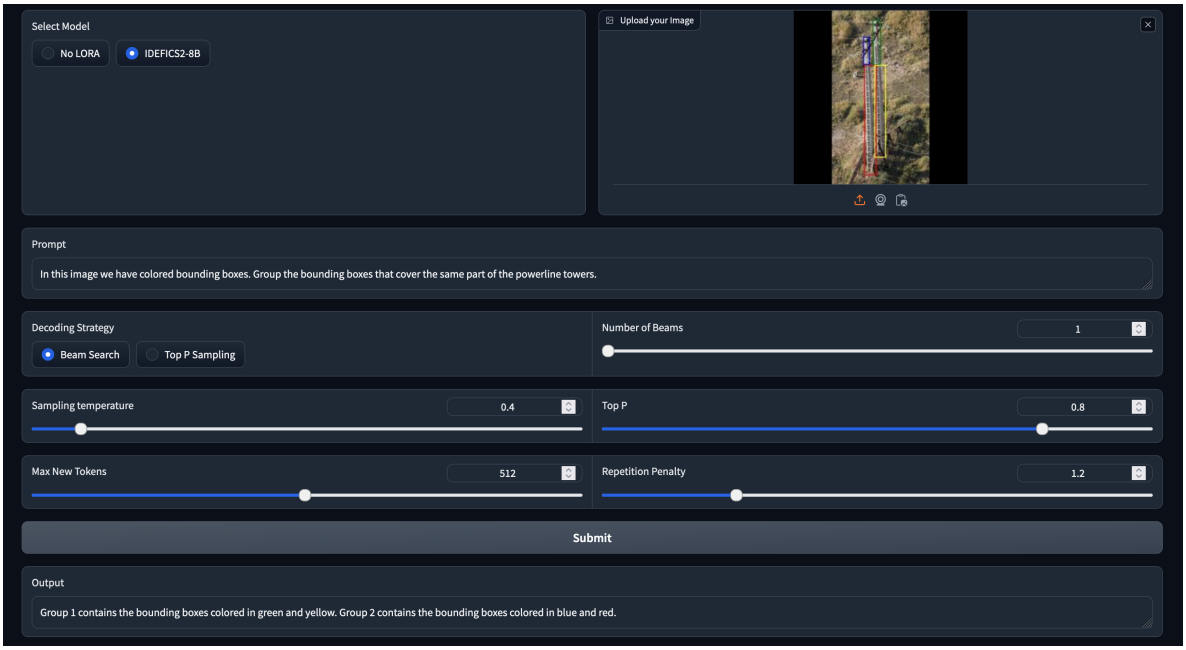

Figure 3: Demo for the pole section matching models

The Idefics2-8B model without finetuning is also not quite satisfactory, However, by finetuning with QLoRA for insulator disc counting, the model was able to achieve somewhat adequate performance of 72% (See Table 2). An interesting phenomenon is that the finetuned Idefics2-8B model only returns a single number as the answer despite what query and images were fed in the model.

For pole section matching, Idefics2-8B is also not capable of producing accurate grouping consistently. Typically, the default either only produces answers of one group even though there are multiple groups or produces nonsensical answers. By finetuning with QLoRA using the formulated dataset mentioned previously. The model gains the capability to produce answers including multiple groups in the same format as the training dataset and the accuracy gets a considerable boost (See Table 3).

Overall, we observe that both vision language models show potential to be finetuned with LoRA or QLoRA adapters to solve the insulator disc counting and pole section matching tasks. However, the current generated models are yet to reach high accuracy and, thus, there is still a lot of room to improve. The training datasets can be made more plentiful and more representative of the two tasks. In addition, other techniques such as prompt tuning, mixture of experts, tree of thought, chain of thought, etc... can be applied to potential achieve more consistent and robust results.

We provide the demos on github. The demo provides an intuitive UI that allows you to switch between models including both fine-tuned and original versions (See Figure **??**). We also beam search and top-p sampling decoding strategies, which are ways the models choose the output tokens based on their probabilities, with adjustable parameters. Users can upload their own image and change the query, however, the models might work best with the queries they are trained with.

5 Conclusion

In conclusion, MiniCPM-V-2 and Idefics2-8B models were fine-tuned with LoRA for insulator disc counting and pole section matching problems. It can be observed that Idefics2-8B is better than MiniCPM-V-2 probably due to its superior size. The results show evidence of adapting to the tasks but there is still more work to be done to improve the models. Demos were provided for public testing.

References

[DPHZ23] Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. Qlora: Efficient finetuning of quantized llms. *arXiv preprint arXiv:2305.14314*, 2023.

[Fac23] Hugging Face. Introducing idefics2: A powerful 8b vision-language model for the community. https://huggingface.co/blog/idefics2, 2023. Accessed: 2024-06-04.

[HSW+21] J. Edward Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *ArXiv*, abs/2106.09685, 2021.

[Ope23a] OpenBMB. Minicpm-v. https://github.com/OpenBMB/MiniCPM-V, 2023. Accessed: 2024-06-04.

[Ope23b] OpenBMB. Minicpm-v 2.0: An efficient end-side mllm with strong ocr and understanding capabilities. https://openbmb.vercel.app/minicpm-v-2-en, 2023. Accessed: 2024-06-04.