Generative AI in IT Infrastructure Networking

1. Project Overview

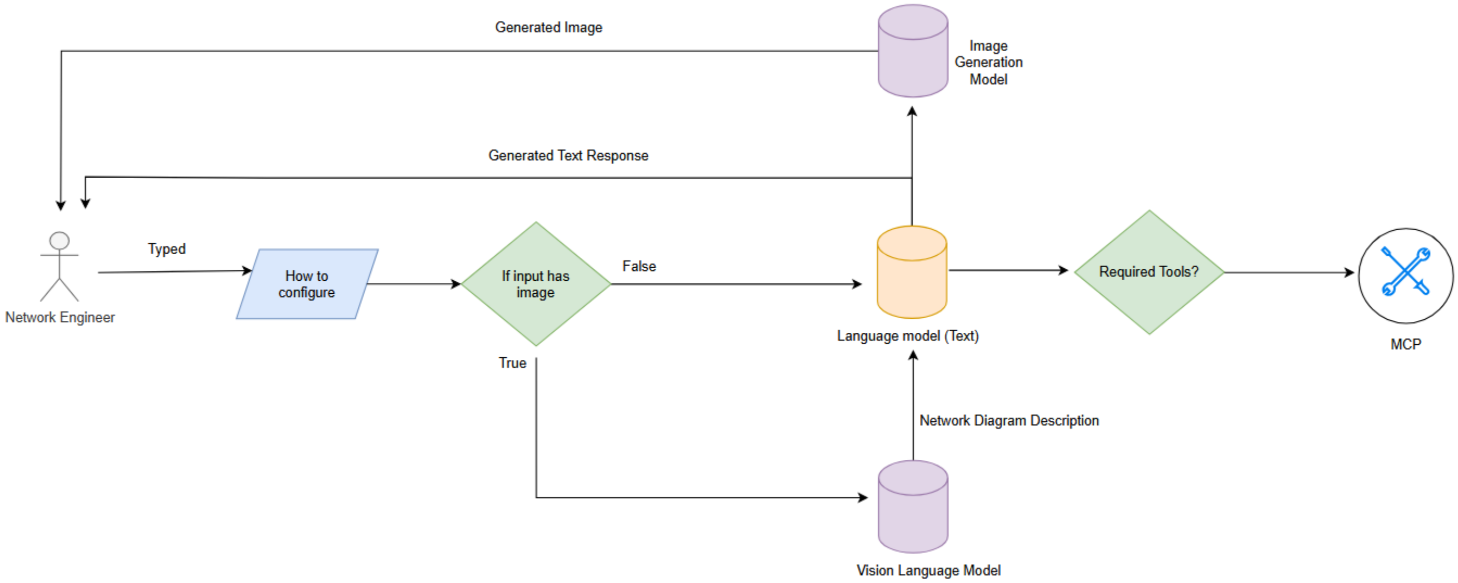


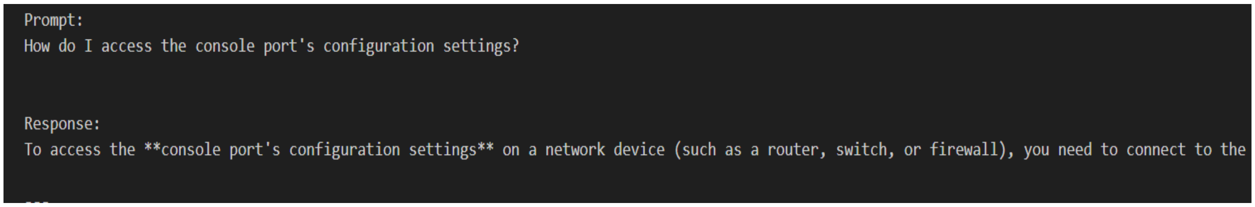
Figure 1: Architecture of GenAI infrastructure

2. Dataset

- The dataset has 2 data types for LLM and Vision Language model (VLM). For image dataset there is different networking devices and topologies developed and with each image there is a description of that image containing in a text file. For the LLM dataset thousands of prompts and responses style text dataset is generated using publicly available LLM with very large no of parameters model.

  Dataset content is based on CCNA and CCNP exams.

- 200-301 CCNA – Cisco Certified Network Associate
- 350-401 ENCOR – Implementing Cisco Enterprise Network Core Technologies
- 300-410 ENARSI – Implementing Cisco Enterprise Advanced Routing and Services
- 300-415 ENSDWI – Implementing Cisco SD-WAN Solutions
- 300-420 ENSLD – Designing Cisco Enterprise Networks
- 300-425 ENWLSD – Designing Cisco Enterprise Wireless Networks
- 300-430 ENWLSI – Implementing Cisco Enterprise Wireless Networks
- 300-435 ENAUTO – Automating Cisco Enterprise Solutions



```
Prompt:
How do I access the console port's configuration settings?


Response:
To access the **console port's configuration settings** on a network device (such as a router, switch, or firewall), you need to connect to the
```

Figure 2 : Example text Dataset

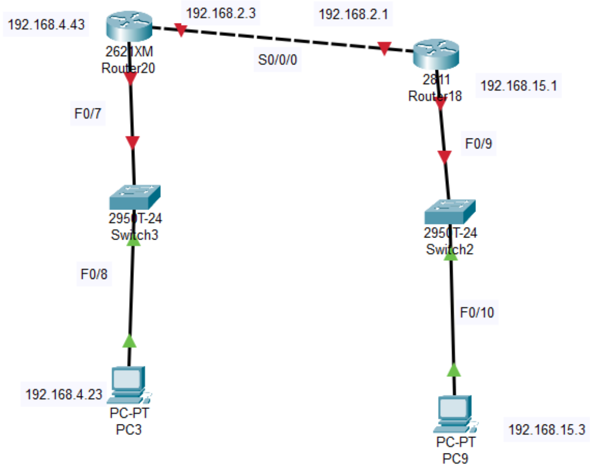
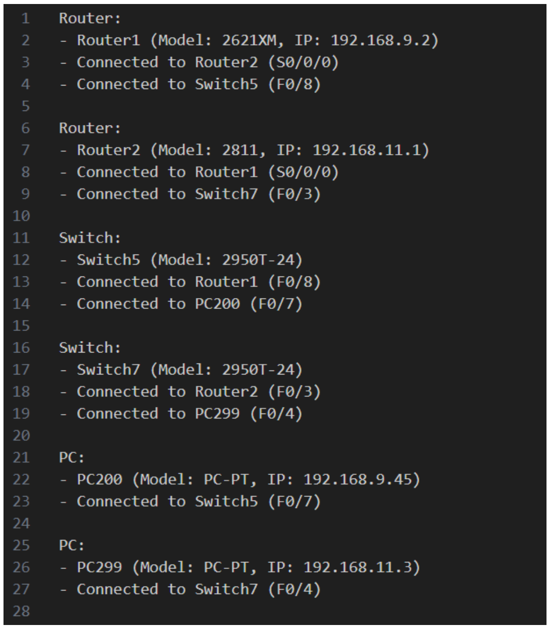


Figure 3: Example image use to finetune



```
Router:
- Router1 (Model: 2621XM, IP: 192.168.9.2)
- Connected to Router2 (S0/0/0)
- Connected to Switch5 (F0/8)

Router:
- Router2 (Model: 2811, IP: 192.168.11.1)
- Connected to Router1 (S0/0/0)
- Connected to Switch7 (F0/3)

Switch:
- Switch5 (Model: 2950T-24)
- Connected to Router1 (F0/8)
- Connected to PC200 (F0/7)

Switch:
- Switch7 (Model: 2950T-24)
- Connected to Router2 (F0/3)
- Connected to PC299 (F0/4)

PC:
- PC200 (Model: PC-PT, IP: 192.168.9.45)
- Connected to Switch5 (F0/7)

PC:
- PC299 (Model: PC-PT, IP: 192.168.11.3)
- Connected to Switch7 (F0/4)
```

Figure 4: Example Image Description

3. Model Training

- Model training is done using Unsloth models and notebooks. The reason unsloth models have been used is they are optimized version of transformer library provided by huggingface.

- Particular model family (ex: Qwen, Gemma, Llama) has not been considered. But the model parameters are considered (ex: 4B, 11B, 70B). Not taking very small sized model which have to produce vast amount of data to generalize also not taking very large sized model so deployment model finetuning less costly.

- Multi models has not been taken because when training it should have same amount of text and images. If not, model is biased, for our case model is largely based on text. So, the vision capability of the model might be less effective also training is less time consuming and less VRAM is needed because of model is trained on 2 separate instances not one.

  - LLM model link

https://colab.research.google.com/github/unslothai/notebooks/blob/main/nb/Llama3.1_(8B)-Alpaca.ipynb

- Vision language model link

https://colab.research.google.com/github/unslothai/notebooks/blob/main/nb/Gemma3_(4B)-Vision.ipynb

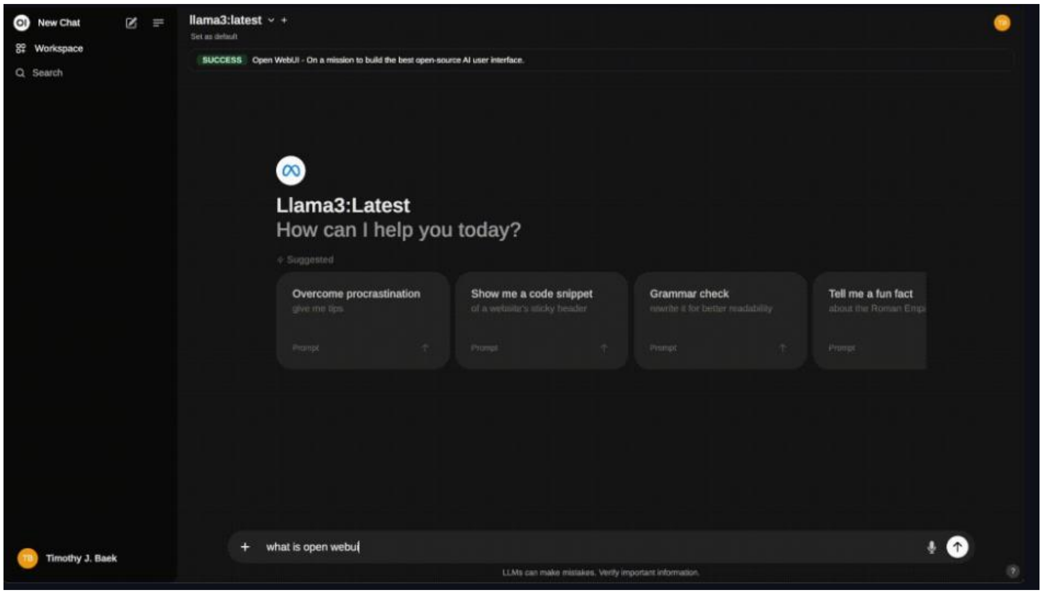
- In the link all the codes are provided only changes done is dataset uploading and model hyperparameter tuning.

4. Image Generation

- For the image generation custom dataset has manually created and finetuned a stable diffusion model using the previously mentioned images that used for finetuning vision language model has been used.

5. Web Interface

- For the web interface to users to interact OpenWebUI has been chosen. This is open-source repository. This is similar to OpenAI ChatGPT web interface. For our task we have fork from OpenWebUI and then customized the website for the company preferences.



SUCCESS Open WebUI - On a mission to build the best open-source AI user interface.

Llama3:Latest

How can I help you today?

what is open webui

LLMs can make mistakes. Verify important information.

Figure 5: Web Interface

6. Model Deployment

- This has to be cost effective and different service providers has to be tested with their pricing. Models can be host locally but response generation might not be fast and it may make the model useless. Therefore, deploying with GPU is suitable

- Also, the quantization also option but this will reduce the accuracy and also loose the purpose of the model. It is ok to do this but need to test it before deploying the model