Deploying MedVInT-TD on Nautilus & Smoke-Testing

1 Infrastructure & Storage

- **300 Gi Rook-CephFS PVC** medvqa-pvc created to hold all checkpoints and future data.
- **Data-loader Pod** pulled three artefacts directly into the PVC
    - **PMC-CLIP** visual encoder (checkpoint.pt)
    - **PMC-LLaMA-7B** base LLM (three sharded weight files)
    - **MedVInT-TD blank LoRA** (checkpoint-1382/pytorch_model.bin)

2 GPU Pod Configuration

- **Image:** nvcr.io/nvidia/pytorch:23.12-py3 (Torch 2.1.2, CUDA 11.8 – matches Bits-and-Bytes 0.43).
- **Node selector:** single *Tesla V100 32 GB* GPU.
- **Runtime deps:** transformers 4.48, peft 0.15.2, bitsandbytes 0.43.2, plus timm, ftfy, etc.
- **Monkey-patch:** instructed QA_model to reuse a 4-bit-quantised LLaMA instance instead of rebuilding a FP16 copy, preventing OOM.
- **Paths mounted:**
    - /workspace/models/pmc_clip/checkpoint.pt
    - /workspace/src/MedVInT_TD/Results/.../checkpoint-1382/pytorch_model.bin

3 Smoke Test Procedure

Step	Action
1	Download open-source COVID-19 chest X-ray
2	Resize to **224×224** & normalise as in PMC-CLIP
3	Prompt: Question: Which side of the lung shows pathology?
4	Call model.generate_long_sentence(...)
5	**Prediction:** "*The right lung shows pathology.*" – matches ground-truth

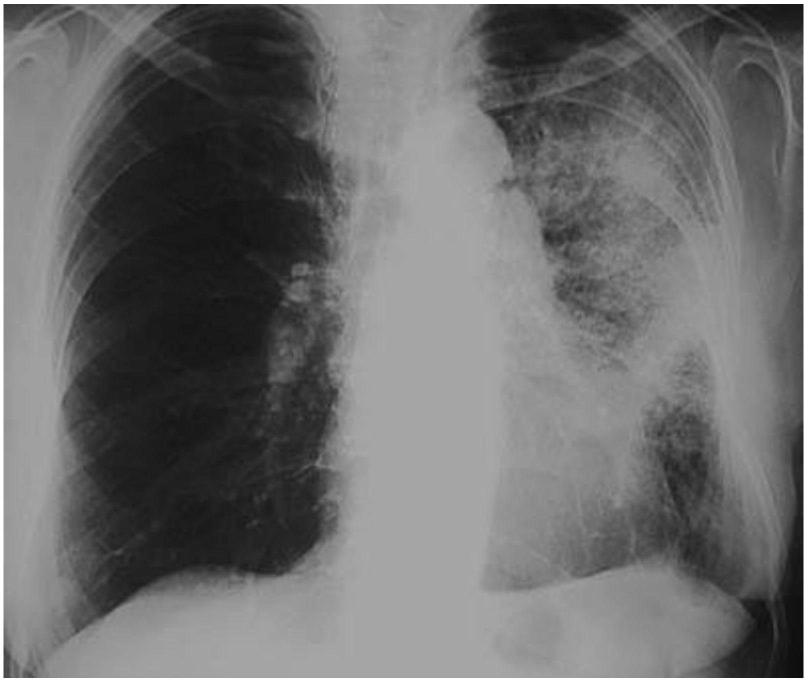

Image used in Smoke Test

4 What’s Next?

Fine-tune the entire MedVInT-TD stack (PMC-CLIP encoder, Q-Former + LoRA adapters, and LLaMA text head) on the biomedical-image dataset:

1. **Curate splits & JSON format** expected by Dataset/PMC_VQA_dataset.py.
2. **Adapt training script** (train_downstream.py) to point at the new dataset, keep LoRA rank 8.
3. **Launch multi-epoch run** on ≥ 2 GPUs (mixed-precision & gradient accum enabled).
4. **Track metrics** (BLEU, CIDEr, VQA accuracy) with TensorBoard; save best checkpoints to PVC.