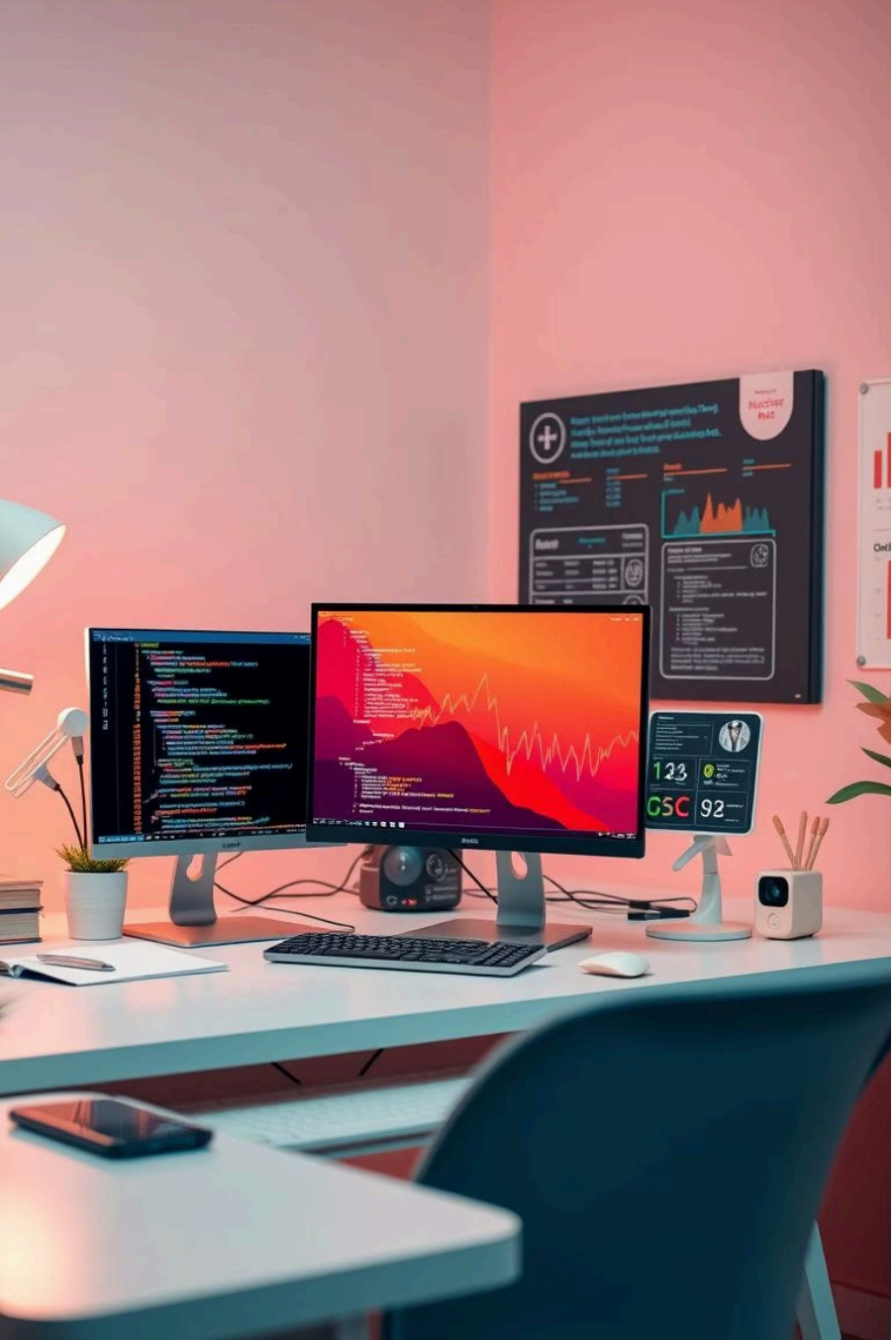

Medical QA Competition – Kaggle Solution Walkthrough

Presented by XDEAM

Data Cleaning

Numeric Answer Removal

Filtered out answers containing only numbers like lab values and codes.

Short Answer Filtering

Excluded answers with fewer than 3 words to retain meaningful context.

Goal

Train on descriptive and relevant text to enhance model understanding.

Model Architecture

Fine-tuned Model

Adapted the Llama 3.2 8B specifically for question answering.

Model Choice Rationale

Chose Llama 3.2 8B based on balance of accuracy and computational efficiency.

Training Methodology

Blazing fast fine-tuning

For fine-tuning, we leveraged **Unsloth**, a fast and memory-efficient library for adapting large language models.

Optimizer & Schedule

We used a pretrained model Meta-Llama-3.1-8B-Instruct-bnb-4bit and fine-tuned it on our cleaned dataset using the SFTTrainer provided by Unsloth.

Parameters & Hardware

Batch size optimized for GPU memory; trained over 1 epoch utilizing LoRA and PEFT for efficient parameter updates.

```
model = FastLanguageModel.get_peft_model(
    model,
    r = 32, # Choose any number > 0 ! Suggeste
    target_modules = ["q_proj", "k_proj", "v_p
                      "gate_proj", "up_proj",
    lora_alpha = 32,
    lora_dropout = 0, # Supports any, but = 0
    bias = "none",    # Supports any, but = "n
    # [NEW] "unsloth" uses 30% less VRAM, fits
    use_gradient_checkpointing = "unsloth", #
    random_state = 3407,
    use_rslora = False,  # We support rank sta
    loftq_config = None, # And LoftQ
```

```
trainer = SFTTrainer(
    model = model,
    tokenizer = tokenizer,
    train_dataset = dataset,
    dataset_text_field = "text",
    max_seq_length = max_seq_length,
    dataset_num_proc = 2,
    compute_metrics = compute_metrics,
    packing = False, # Can make training 5x faster for sh
    args = TrainingArguments(
        per_device_train_batch_size = 4,
        gradient_accumulation_steps = 8,
        warmup_steps = 5,
        # num_train_epochs = 1, # Set this for 1 full tra
        max_steps = 60,
        learning_rate = 2e-4,
        fp16 = not is_bfloat16_supported(),
        bf16 = is_bfloat16_supported(),
        logging_steps = 1,
        optim = "adamw_8bit",
        weight_decay = 0.01,
        lr_scheduler_type = "linear",
        seed = 3407,
        output_dir = "outputs",
        report_to = "none", # Use this for WandB etc
    ),
```

Training parameters

Results:

reached loss of 1.206 after 60 steps in the best submission

```
60          1.206500
```

Inference and Submission

Post-processing

Refined model outputs to generate accurate answer spans.

Submission File

Formatted output to meet Kaggle competition requirements.