IntelliHack 5.0

Machine Learning hackathon

Q3 - Training process with code

The Code Rushers

01. Instructions for Running the Program

1.1 Installation of Dependencies

```
pip install torch transformers peft bitsandbytes accelerate datasets huggingface_hub pymupdf pandas docx
pip install ragas
pip install python-docx
```

1.2 Loading the Model

```
from transformers import AutoModelForCausalLM, AutoTokenizer

model_name = "Qwen/Qwen2.5-3B-Instruct"
tokenizer = AutoTokenizer.from_pretrained(model_name)
model = AutoModelForCausalLM.from_pretrained(model_name, device_map="auto", torch_dtype="auto")
```

1.3 Data Extraction from PDFs

```
import fitz  # PyMuPDF

def extract_text_from_pdf(pdf_path):
    text_data = []
    try:
        doc = fitz.open(pdf_path)
        for page in doc:
            text_data.append(page.get_text("text"))
    except Exception as e:
        print(f"Error: {e}")
    return "\n".join(text_data)
```

1.4 Fine-Tuning the Model with LoRA

```
from peft import LoraConfig, get_peft_model, TaskType

config = LoraConfig(task_type=TaskType.CAUSAL_LM, r=8, lora_alpha=32, lora_dropout=0.05)
model = get_peft_model(model, config)
```

1.5 Training Execution

```
from transformers import TrainingArguments, Trainer

training_args = TrainingArguments(
    output_dir="./results",
    num_train_epochs=3,
    per_device_train_batch_size=8,
    save_steps=10_000,
    save_total_limit=2,
    evaluation_strategy="epoch",
    learning_rate=5e-5,
    weight_decay=0.01,
    logging_dir="./logs",
)

trainer = Trainer(
    model=model,
    args=training_args,
    train_dataset=dataset,
    eval_dataset=eval_dataset,
)

trainer.train()
```