Tela de computador com texto preto sobre fundo branco

Descrição gerada automaticamente com confiança baixa

TechChallenge Fase 3

Pós-Tech em IA para DEVS

FIAP

**INTEGRANTES DO GRUPO 18**

Beatriz Cardoso Cunha  
Francisco Giuan Miranda Ferreira  
Maurício Lachaitis da Silva

17 de dezembro de 2024

🚀 Relembrando o Desafio do Tech Challenge

Objetivo Principal:

1. Executar o fine-tuning de um modelo de linguagem (ex.: LLaMA, BERT, GPT) usando o dataset "The AmazonTitles-1.3MM".
2. Receber perguntas dos usuários com base em um contexto (título do produto).
3. Gerar respostas baseadas na descrição do produto após o fine-tuning.
4. Documentar e Apresentar:
5. Explicar os parâmetros, ajustes e resultados.
6. Criar um vídeo demonstrando o modelo em ação.

Dataset:

O dataset contém títulos de produtos e suas descrições provenientes da Amazon.

🔗Links Principais:

* <https://github.com/beatrizcardc/TC_fase3_Fine_tuning_llama3.git>
* <https://youtu.be/yFqfh31Yg10>

**📌 Pipeline Completo do Trabalho**

**🗂️ Escolha do Dataset**

* **Dataset**: The AmazonTitles-1.3MM.
* **Campos**: Utilizar os campos **título** e **descrição** dos produtos.
* 📥 **Download e Armazenamento**: Realizar download e armazenar de forma segura no **Google Drive**.

**🧹 Preparação do Dataset**

* 📤 **Carregar e Inspecionar**: Carregar e revisar os dados.
* 🧼 **Limpeza**:
  + Remover linhas com valores **ausentes**.
  + Remover **duplicatas**.
  + Normalizar o texto (remoção de **caracteres especiais**).
* 📉 **Redução**: Reduzir o tamanho do dataset para **40k registros**.
* 🔀 **Divisão**: Separar em **80% treino** e **20% validação**.

**📊 Análise de Comprimento**

* 📏 **Analisar Comprimentos** dos prompts e responses.
* 🎯 **Definir max\_length**:
  + **Prompt**: 75 tokens.
  + **Response**: 450 tokens (para acomodar até 300 palavras).
  + **Total**: 525 tokens.
* 🛠️ **Ajuste de Hyperparameters**: Para evitar erros de **Out Of Memory (OOM)** durante o treinamento.

**🔧 Carregamento do Modelo**

* 📥 **Modelo Pré-treinado**: Llama 3.2-1B ou similar.
* 🧪 **Quantização**: Aplicar **8-bit** ou **4-bit** para otimização de memória.
* 🛠️ **LoRA**: Configurar **LoRA** para fine-tuning eficiente.

**🔄 Pré-processamento e Tokenização**

* 📝 **Função de Pré-processamento**:
  + Combinar prompts e responses em um único formato.
  + **Tokenização** com truncamento e padding.
* 🔗 **Definir pad\_token** como eos\_token para compatibilidade.

**⚙️ Configuração do Treinamento**

* 📊 **Hiperparâmetros**:
  + **Epochs**: 3 a 5.
  + **Batch Size**: 4 a 2 (ajustado para evitar OOM).
  + **max\_length**: 525 tokens (ou conforme análise – 700 - 1024).
  + **Checkpoints**: Salvar checkpoints a cada save\_steps.
* 📊 **Monitoramento**: Usar **WandB** e **TensorBoard** para acompanhar métricas como:
  + eval\_loss
  + validation\_loss
  + Entrega de 3 relatórios, dias 13, 15 e 17 de dezembro
* 💾 **Callbacks**: Salvar os **dois últimos checkpoints** automaticamente.

**🚀 Execução do Fine-Tuning**

* ▶️ **Iniciar Treinamento**: Utilizar o Trainer para treinar o modelo.
* 📉 **Validação**: Avaliar durante o treinamento para acompanhar eval\_loss e validation\_loss.

**🧪 Avaliação e Geração de Respostas**

* 📝 **Dados de Teste**: Carregar os dados de teste e o **Ground Truth**.
* 📥 **Modelos**: Carregar os modelos **pré-treinado** e **fine-tuned**.
* 🗨️ **Gerar Respostas**: Produzir respostas para os dados de teste.
* 🔄 **Retreinamento com Ajustes**: Modificar hiperparâmetros e formatar o dataset com title e content.
* 📈 **Métricas de Desempenho**: Deixado como documentação – não exigido no tech Challenge
  + **BLEU**
  + **ROUGE**

**📦 Entrega do Projeto**

* 📄 **Documento Detalhado**:
  + Descrição da seleção e preparação do dataset.
  + Processo de fine-tuning com parâmetros utilizados.
* 💻 **Código-Fonte**: Repositório com o código do fine-tuning.
* 🎥 **Vídeo de Demonstração**: Mostrando o modelo gerando respostas.