



Relatório Projeto

Inteligência Artificial

Bárbara Martins 41403

1. Introdução (Motivação e Objetivos)

Nos últimos anos, a Inteligência Artificial (IA) tem-se afirmado como uma tecnologia essencial em diversas áreas, incluindo a educação e o apoio ao utilizador. A capacidade de compreender e gerar linguagem natural tornou os Modelos de Linguagem de Grande Escala (LLMs) especialmente promissores para aplicações como assistentes virtuais e chatbots informativos.

Este projeto tem como objetivo o desenvolvimento de um chatbot baseado no modelo Gemma da Google, com capacidade de responder em português e inglês a perguntas sobre a Universidade Fernando Pessoa (UFP). O foco principal está nos cursos de Engenharia Informática, tanto a Licenciatura (INF) como o Mestrado (MINF), com a intenção de auxiliar estudantes e interessados a obterem informação clara e objetiva sobre estas formações.

2. Descrição do Problema

A UFP disponibiliza informação sobre os seus cursos através de websites institucionais, mas essa informação nem sempre é facilmente acessível ou interativa. Utilizadores com dúvidas específicas enfrentam dificuldades em obter respostas rápidas e personalizadas. Este projeto pretende colmatar essa lacuna através da criação de um chatbot especializado, que compreenda linguagem natural e devolva respostas fidedignas, estruturadas e úteis.

3. Estado da Arte

Entre as soluções mais populares de assistentes baseados em IA estão ChatGPT (OpenAI), Gemini (Google DeepMind) e Claude (Anthropic), todos baseados em LLMs.

O modelo **Gemma**, desenvolvido pela Google, apresenta-se como uma alternativa leve, eficiente e open-source, com variantes de 2B e 7B parâmetros, adaptadas a diferentes contextos de aplicação. Técnicas como **Few-Shot Prompting** permitem adaptar rapidamente modelos a domínios específicos com apenas alguns exemplos. A abordagem **RAG (Retrieval-Augmented Generation)** combina a capacidade de geração dos LLMs com uma base de conhecimento externa. Já o **LoRA (Low-Rank Adaptation)** permite fazer fine-tuning eficiente com poucos recursos. Por fim, **MediaPipe** é uma framework promissora para integração em dispositivos móveis.

4. Descrição do Trabalho Realizado

O trabalho foi desenvolvido em Python, utilizando as bibliotecas transformers, sentence-transformers e faiss. O projeto passou pelas seguintes etapas:

- Instalação e execução do modelo **Gemma-2B-it** em ambiente local.

- Criação de um chatbot por consola com pipeline("text-generation").
- Integração de **few-shot prompting**, usando exemplos representativos guardados no ficheiro exemplos_ufp_prompt.txt, para orientar o comportamento do modelo.
- Desenvolvimento de uma versão com **RAG**, que utiliza o modelo de embeddings all-MiniLM-L6-v2, indexa os textos com FAISS e recupera os parágrafos mais relevantes a partir de documentos_ufp.txt.
- Implementação de logging automático em ficheiros como conversa_ufp_*.txt e conversa_rag_*.txt.
- Preparação para futura integração de LoRA e compatibilidade com MediaPipe.

5. Análise de Resultados

Os testes realizados mostram que o modelo responde corretamente a perguntas como:

- "Que cursos existem na UFP?"
- "Que unidades curriculares tem a licenciatura em Engenharia Informática?"
- "Quantos ECTS tem o Mestrado?"

A versão com RAG obteve melhores resultados em perguntas específicas, pois recupera contextos reais antes de gerar resposta.

Sem RAG, o modelo tende a dar respostas genéricas, mesmo com prompting. Com RAG, as respostas incluem informação precisa como "Bases de Dados (6 ECTS)" ou "Dissertação: 30 ECTS".

6. Conclusões e Perspetivas de Desenvolvimento

Este projeto mostrou que é possível criar um chatbot inteligente, especializado na divulgação de cursos da UFP, com recurso ao modelo Gemma e às técnicas de IA mais recentes. A introdução do mecanismo RAG foi fundamental para garantir respostas mais fidedignas e centradas na documentação oficial.

Como desenvolvimento futuro, pretende-se:

- Aplicar **LoRA fine-tuning** para melhorar a performance do modelo.
- Integrar o chatbot com **MediaPipe** para interação multimodal (voz, gesto).
- Expandir a base de dados com mais detalhes sobre candidaturas, propinas e apoio ao estudante.

- Criar uma interface gráfica acessível via web e/ou móvel.

Anexo: Links e Recursos

- Ficheiros importantes:
 - chatbot_ufp.py (versão sem RAG)
 - chatbot_ufp_rag.py (versão com RAG)
 - exemplos_ufp_prompt.txt (few-shot prompts)
 - documentos_ufp.txt (base de conhecimento para RAG)
 - conversa_ufp_.txt e *conversa_rag_.txt* (logs)
 - requirements.txt (dependências do projeto)