Automated Technical Documentation from Code Repositories: A Literature Review

Adarsh Mishra (24214663)
Shreyas Achary (24217226)

1 Introduction

In the software development lifecycle, developers often spend a significant amount of time drafting technical documentation. These documents are essential for knowledge transfer, debugging, version updates, onboarding, and compliance. However, due to time constraints during the development process, the documentation is often rushed. This project aims to build an LLM-based tool that can parse a GitHub repository and generate well-structured technical documentation.

2 Motivation

As engineers, we have frequently faced the challenge of drafting technical documentation. This process is often time-consuming and requires manual effort to capture all relevant aspects of the application. Additionally, documentation is typically addressed at the end of the development cycle when time is limited. Poor documentation can lead to technical issues, longer onboarding times, and miscommunication. Therefore, the motivation behind this project is to leverage LLMs to automate and streamline the documentation process.

3 Current Line of Thought

Our current line of thought is to develop a system that can automatically generate a structured technical documentation from a GitHub repository. Our plan is to first build a Vector Database that stores semantically meaningful and relevant information extracted from the GitHub repository, this includes code, comments, configuration files, and metadata. We plan on using LangChain to build an agent that will oversee the complete documentation process. This agent would essentially decide the structure of the document, capture the most relevant information from the vector database for each section, and then use a large language model (LLM) to generate the actual content. Since modern LLMs are trained on extensive data and are capable of generating fluent and clear text with a low perplexity score, we expect them to handle language generation effectively, allowing us to focus on ensuring that the retrieved content is accurate and relevant. Ultimately, our goal is to produce a comprehensive and user friendly technical document, generated automatically from the repository content, without requiring any form of manual writing or editing.

4 Technology Overview

- **Vector Database:** We plan to use FAISS or Chroma to store semantic embeddings of repository data which would include code, comments, and metadata for fast and relevant retrieval.
- **Embedding Models:** We will utilize Hugging Face embedding models to convert the text chunks into dense vector representations.
- **Large Language Model (LLM):** Text generation will be performed using GPT-4o, vertex AI or suitable Hugging Face LLMs to create comprehensive and fluent documentation sections.
- **LangChain Framework:** LangChain will be used to implement an agentic process that plans the document structure, retrieves relevant content, and generates each section.
- **GitHub Integration:** GitPython or PyGithub will enable us to clone repositories and extract structured data and metadata from Github repositories.

5 Expected Output

- A robust pipeline that:
  - Takes in the source code and metadata from the GitHub repository.
  - Generates complete Documentation using RAG.
- A user-friendly tool that streamlines and simplifies the software documentation process, significantly reducing the time and effort required to create such documents.

6 Future Enhancements

- **Interactive Docs:** An interface where by the end users can ask questions about the application.
- **Multilingual Support:** Extend capabilities to include documentation in multiple languages.
- **Video guides:** Create step by step video guides on the workings of the application.

7 References

- Naimi, L., Bouziane, E. M., Jakimi, A., Saadane, R., & Chehri, A. *Automating Software Documentation: Employing LLMs for Precise Use Case Description*, Procedia Computer Science, vol. 246, pp. 1346–1354, 2024. https://www.sciencedirect.com/science/article/pii/S1877050924026176
- Chakrabarty, S., & Pal, S. *Free and Customizable Code Documentation with LLMs: A Fine-Tuning Approach*, arXiv preprint arXiv:2412.00726, Dec 2024. https://arxiv.org/abs/2412.00726
- Thota, S. R., Arora, S., & Gupta, S. *AI-Driven Automated Software Documentation Generation for Enhanced Development Productivity.* Proceedings of the 2024 International Conference on Data Science and Network Security (ICDSNS), Visakhapatnam, India. Available at: https://www.researchgate.net/publication/384543547_Al-Driven_Automated_Software_Documentation_Generation_for_Enhanced_Development_Productivity