Documentation for Product Matching with Inventory System

Introduction

Spelling mistakes can lead to confusion, especially in inventory management systems where product names must be accurate. This system aims to correct spelling errors by utilizing vector embeddings and a language model to identify the most similar correct product names.

Problem Statement

In inventory management, incorrect product names due to typographical errors can cause errors in processing, lead to incorrect orders, and affect overall efficiency. The challenge is to create a system that can automatically detect and correct these errors using advanced AI models.

Aims

The primary aim of this project is to develop a tool that automatically corrects misspelled product names by finding the closest match from a list of correct names in the inventory dataset.

Objectives

1. To extract and preprocess product names from an inventory dataset.
2. To convert product names into vector embeddings using a pre-trained model.
3. To store these embeddings in a vector database for efficient similarity search.
4. To develop a function that takes an incorrect product name and finds the closest matching correct name.
5. To use a language model to refine the corrections if necessary.

Workflow

1. **Getting the Correct Data**: Load inventory data from an Excel file.
2. **Converting into Embeddings**: Use a pre-trained model to generate embeddings for product names.
3. **Saving Vector Embeddings:** Store the generated embeddings in a text file for easy retrieval.
4. **Putting into Vector Database**: Load the embeddings into a FAISS index for efficient similarity search.

5. **Finding Best Possible Words**: Search for the most similar product names using the FAISS index.
6. **Sending it to LLM**: Utilize a language model to further refine the correction suggestions.

Advantages

1. Automated correction of spelling mistakes.
2. Uses state-of-the-art models for accuracy.
3. Efficient search using vector databases.

Limitations

1. Dependent on the quality of the initial inventory data.
2. Requires computational resources for embedding generation and vector search.