TEXT EMBEDDING AND SIMILARITY SEARCH WITH OPENAI AND PINECONE

28 APRIL, 2023

OVERVIEW

1. Introduction

The goal of this project is to demonstrate a practical application of text embedding and similarity search using OpenAI's API and Pinecone's platform. The project involves vectorizing a set of text documents using OpenAI's text embedding model, storing the embeddings in Pinecone's index, and performing a similarity search for a given query string.

2. Methodology

The program starts by importing the necessary libraries such as os, tqdm, io, numpy, multiprocessing, dotenv, pinecone, openai, and gdown. We then load our Pinecone API key and OpenAI API key using the dotenv package. After initializing Pinecone and creating an index with the desired dimension and shards, we define a function to vectorize a single file using OpenAI. This function reads the file, breaks it into batches of size batch_size, and vectors each batch using OpenAI's text embedding API. The resulting embeddings and their corresponding metadata are stored in the Pinecone index using the upsert function.

Next, we define a loop that iterates over all files in a specified local folder, constructs the full path to each file, and calls the vectorize_file function to vectorize the text data in each file. We also show how to query the index using a user-provided query, retrieve the closest embeddings, and print their corresponding metadata.

Finally, we delete the Pinecone index, and remove all files from the local folder.

3. Results

The project code successfully vectorized a set of text documents using OpenAI's text embedding model and stored the embeddings in Pinecone's index. The similarity search for a given query string returned the top 10 most similar documents based on their embeddings

4. Discussion

Vectorizing text data is a crucial step in many natural language processing (NLP) tasks, such as sentiment analysis, text classification, and document similarity. OpenAI's text embedding API provides a convenient and efficient way to vectorize text data, and Pinecone provides a scalable and easy-to-use platform for storing and querying high-dimensional embeddings. This program demonstrates how to use *these tools together to vectorize large amounts of text data and perform fast and accurate similarity searches*

5. Conclusion

In this project, we showed how to vectorize text data using OpenAI's text embedding API and store the resulting embeddings in a Pinecone index. We also demonstrated how to query the index to retrieve the closest embeddings to a user-provided query. This program provides a practical example of how to leverage state-of-the-art NLP tools to process and analyze large amounts of text data.