Title Slide

Google Hackathon Report

Team 10: Prompt to Slides

November 4, 2023

Aniruddh Ingle, Sylvanne Braganza, Sherry Li, Hritik Bhansali, Cece Zhang

Project Overview

Our project aimed to transform text-based content into engaging presentations with the help of Google's powerful PaLM2 API. By harnessing the capabilities of this cutting-edge natural language processing tool, we streamlined the process of creating dynamic presentations from textual information. Whether it was converting lengthy reports into informative slides or transforming written content into compelling spoken presentations, our team enhanced the efficiency and effectiveness of communication. With the PaLM2 API, we automated the structuring and summarizing of information, allowing users to effortlessly transform their ideas and data into engaging spoken presentations. This project is a game-changer in simplifying the presentation creation process, making it an invaluable tool for professionals, educators, and anyone looking to communicate their ideas effectively.

MakerSuite for Prompt Engineering

In our project, we leveraged Google's MakerSuite for prompt engineering. Initially, we explored different prompt types, including text prompts, data prompts, and chat prompts. After experimenting with these options, we found that the data prompt was the most suitable choice for our project aims due to its ability to tailor the input and output structure according to our project's specific requirements (text to Markdown-formatted script). Data prompts work by taking custom instructions provided by users. In our case, we asked it to generate a presentation in Markdown script format tailored to a specific age group. Then we give the model specific input and output examples to work with. These prompts give the model examples to replicate. Google calls these "few-shot prompts." They served as training data for the model. Over time, we fine-tuned these examples based on the specific content structure and style we wanted for our presentations. This iterative process allowed the model to learn and adapt to our requirements.

Document Search with Embeddings

This technique allows for searching documents by comparing their embedded representations, or vectors of numbers that capture the semantic meaning of text. The steps are as follows. First, we must generate embedded representations for all of the text in the document(s). This can be done using a pre-trained embedding model, a few of which are provided by Google. Next, we must generate an

embedded representation of the query text. This is done using the same embedding model as used in the first step. Now, we can compare the embedded representation of the query text to the embedded representations of the input document(s). This is done using similarity metrics, such as distance determinations. Finally, we must return the results with the highest scores.

Google API

Our final step was connecting our input and output PDF files to our Google Drive account for ease of use and seamless access. We were able to do this using Google APIs Explorer. The Google Drive API allows our model to access resources from Google Drive.

How to Use

To use our project, users can follow these steps:

1. Upload a PDF document into a folder in their Google Drive.
2. Adjust the query to specify age or education level.
3. Run our python script to generate a PDF presentation of slides, which will be saved back into their Google Drive.

Future Directions

We plan to continue to build on this project for the remainder of the mini. As previously discussed, we are presently utilizing the PaLM API to generate embeddings that enable us to perform searches within PDF documents stored in a Google Drive using our queries. So far, our testing has predominantly focused on prompts in the English language. However, as document search with embedded text is adaptable and capable of searching in various languages, we intend to extend our testing to encompass languages beyond English, including Chinese and Hindi, to build on this project. Furthermore, we intend to broaden our scope by exploring additional types and lengths of documents that can be searched using our queries. Currently, our testing is limited to searching through PDF documents. However, we are planning to expand our search capabilities by including Word documents and websites in our testing, aiming to enhance the versatility and adaptability of our query.