Project: Fashion Search AI on Myntra Dataset

Introduction:

To Create a generative search system capable of searching a plethora of product descriptions to find and recommend appropriate choices against a user query. You may use this <u>Myntra dataset</u> on Kaggle to build such a system.

Objectives:

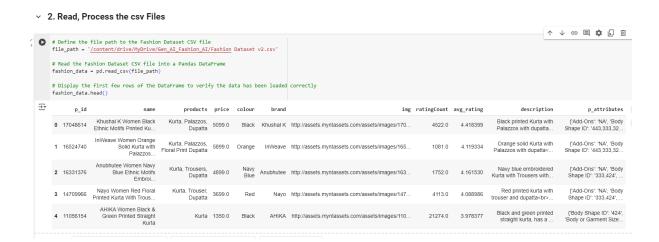
The main objective of this project is to develop a fashion query response system that utilizes AI models to provide detailed and user-friendly responses to fashion-related queries. The system aims to enhance user experience by generating informative and contextually relevant answers, thereby assisting users in finding fashion items based on their preferences.

Design:

The project involves two main layers: the search layer and the generation layer. The search layer is responsible for retrieving relevant fashion items from the dataset based on keyword matching or predefined criteria. The generation layer utilizes advanced AI models like GPT-3.5 to generate detailed responses to user queries, incorporating context and generating natural language responses.

Implementation:

The implementation involved several steps, including data preprocessing, model integration, and query response generation. Data Preprocessing: The CSV dataset was formatted to enhance data quality and readability. Blank entries were replaced, decimal points were standardized, and unnecessary columns were removed. Text columns were cleaned to remove HTML tags and extra characters.



Model Integration:

Advanced AI models like GPT-3.5 were integrated into the system to generate responses to user queries. Queries were passed through the model to generate detailed and contextually relevant responses.

Query Response Generation:

User queries were processed by both the search layer and the generation layer. The search layer retrieved relevant fashion items from the dataset. The generation layer utilized AI models to generate detailed responses to user queries, incorporating context and generating natural language responses.

Here are the few queries that were used to test the model:

#Query 1:

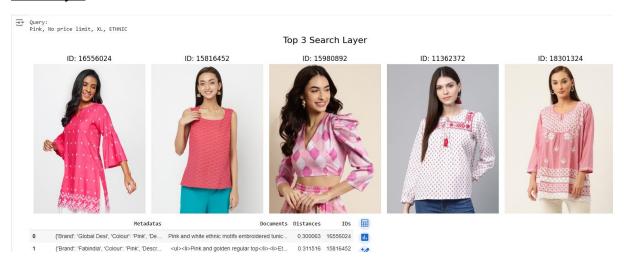
an orange summer dress or kurta to wear over blue denim jeans.

```
[14] # Prompt the user to enter a fashion-related query with specific requirements query = input('Please enter your fashion-related query. For example, you can mention the color, size, price range, style, and any specific patterns or designs you prefer: ")

#Query 1: a orange summer dress or kurta to wear over blue denim jeans .
#Query 2: I'm looking for office wear sarees in elegant colors like pink, violet, or green. I prefer sarees with full embroidered designs, suitable for professional settings. There's no specific price #Query 3: I'm searching for a versatile black leather jacket, suitable for various occasions and effortlessly complementing any outfit. Preferably in size XL.
#Query 4: "I'm searching for a versatile black leather jacket, suitable for various occasions and effortlessly complementing any outfit. Preferably in size XL.
#Query 4: "I'm searching for a versatile black leather jacket, suitable for various occasions and effortlessly complementing any outfit. Preferably in size XL.
#Query 5: A silk purple summer dress, for beach party

**Please enter your fashion-related query. For example, you can mention the color, size, price range, style, and any specific patterns or designs you prefer: Pink, No price limit, XL, ETHNIC
```

Search Layer:



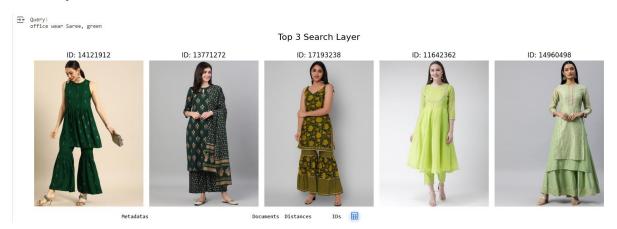
Generation layer:



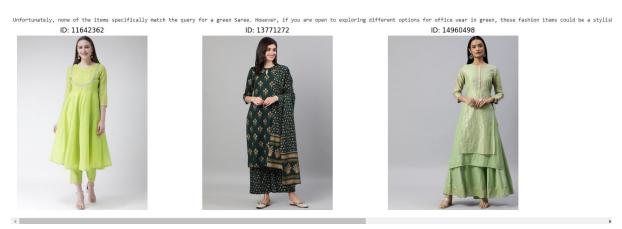
#Query 2:

I'm looking for office wear sarees in elegant colors like pink, violet, or green. I prefer sarees with full embroidered designs, suitable for professional settings. There's no specific price range, as quality is my priority. These sarees will be worn for office meetings and formal events.

Search Layer:



Generation layer:

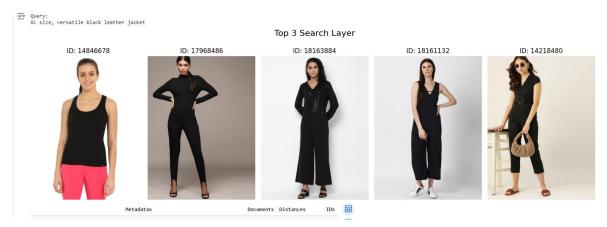


#Query 3:

I'm searching for a versatile black leather jacket, suitable for various occasions and effortlessly complementing any outfit. Preferably in size XL.



Search Layer:



Generation layer:

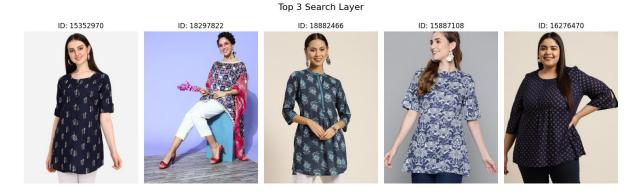


#Query 4:

"I'm seeking ethnic attire suitable for adults, with a preference for Ishin brand. These outfits are intended for festival wear. The clothing set I'm looking for should include a Navy Blue and golden foil printed top with a round neck and three-quarter sleeves, along with a matching skirt featuring a drawstring closure. Both pieces should be made of pure cotton.



Search Layer:



Generation layer:







There were many more queries that was used, for project purpose I have listed only few here as well as at the code terminal.

Challenges:

Several challenges were encountered during the implementation process:

Metadata Processing:

Initially, there were issues with processing metadata, which affected the display of results. This was resolved through debugging and refining the metadata processing code.

Dataset Chunking:

Loading the entire dataset without chunking posed a challenge due to memory constraints. However, chunking was not implemented due to time constraints and the manageable size of the dataset.

Lessons Learned:

Proper data preprocessing is crucial for ensuring data quality and readability. Integrating advanced AI models can significantly enhance the system's capabilities in generating detailed

and contextually relevant responses. Handling large datasets requires careful consideration of memory constraints and implementation of efficient data processing techniques.

Future Scope:

As a future scope, the project can be extended by implementing it as a Flask web application and hosting it on a website. This would provide better interaction and aesthetics for users. Additionally, rephrasing prompts and introducing interactive sessions with criteria-based filters could further enhance the system's performance and user experience.

Conclusion:

Upon comparison of the search query outcomes from both the search and generation layers, it becomes apparent that the generation layer produces more detailed and comprehensible results. While the search layer retrieves relevant information through keyword matching or set criteria, the generation layer utilizes advanced AI models like GPT-3.5 to dynamically generate responses, incorporating context and furnishing tailored and informative answers.

The generation layer's capability to comprehend query context and furnish responses in natural language enhances overall readability and user engagement. Its flexibility allows it to adapt to various query formats and provide customized responses, catering to diverse user preferences. This versatility makes it a valuable tool for tasks necessitating comprehensive understanding and communication of information.

In summary, while the search layer efficiently retrieves relevant information, the generation layer significantly enhances output quality and readability, making it a preferred choice for tasks requiring detailed and user-friendly responses.