Fashion Bot: become stylish

—

By Kaygi Erman and Vieillevigne Julien
Chatbots & Recommender Systems
EFREI Paris

Introduction

We hear the same thoughts all the time! What should I wear today? Should I wear a t-shirt or a shirt in light or dark colors? How can I find a creative outfit that will make me stand out from the crowd?

It was with this problem in mind that we decided to create the Bot Fashion chatbot. An interactive chatbot whose aim is to advise its users on the outfits they're wearing, with illustrations to top it all off.

To do this, we've used all the knowledge we've acquired over the course of the year, along with the latest cutting-edge tools: generative artificial intelligence.

Targets

1. **Engaging in fashion conversations**: The algorithm should be capable of engaging in conversations about fashion and clothing with users and asking them questions.
2. **Collecting user information**: It should gather relevant information from users regarding their age, gender, origin, style preferences, favorite brands, and accessories.
3. **Generating fashion advice**: Based on the user's responses and the collected information, the algorithm should provide fashion advice and outfit suggestions.
4. **Displaying recommendations in images**: It should display the generated fashion recommendations in images.

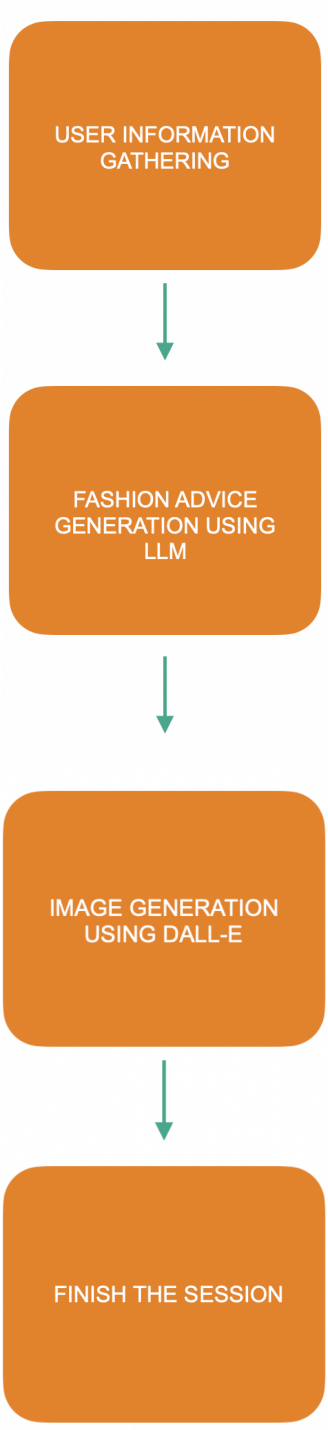
Structure of our algorithm

The algorithm's initial steps involve collecting important information from the user. This is achieved by engaging in a conversation with the user, asking relevant questions, and using regular expressions (regex) to identify user intents. Key information such as age, gender, origin, clothing style, and other preferences are extracted from the conversation. This phase is essential to personalize fashion recommendations based on the user's preferences.

Once the algorithm has gathered meaningful user information, it utilizes OpenAI's GPT-3.5 Turbo API to generate fashion advice. User information is incorporated into a prompt that guides the model in generating personalized fashion recommendations. The GPT-3.5 Turbo model can understand the context of the conversation and provide relevant suggestions based on the user's preferences and information.

To complement fashion recommendations, the algorithm invokes the DALL-E API, which is based on the e3 model of DALL-E. This step involves retrieving the clothing style recommended by GPT based on key user information. Then, this information is preprocessed to create a specific prompt for DALL-E. The DALL-E model then generates a color image matching the fashion recommendations provided by the user. The generated image is intended to help the user visualize the recommended outfit.

Once the algorithm has provided fashion recommendations and images, it concludes the conversation session in a friendly manner. It thanks the user for their participation and conversation. This step marks the end of the interaction between the algorithm and the user, although the user may choose to continue the conversation or end it at any time. The goal is to make the user's experience enjoyable and complete.



Main stages

I. API Key

In order to use the ChatGPT and DALL-E APIs, we need to retrieve an API Key that we import at the start of our project.

II. ChatGPT Function (advise_fashion)

In order to give the best fashion advice to the user, we chose to use ChatGPT's API because it is fast, gives consistent answers and understands the context. ChatGPT allows us to adapt to each user. That's why we decided to use it.

We've created the function advise fashion that uses ChatGPT as an assistant and takes the prompt we give it as a parameter. This function will have the generated text as output.

III. DALL-E Function (url_image & display_image)

The function using the DALL-E model uses the same principle as the ChatGPT function, with a prompt as input to generate an image. In our case, this image should illustrate a human resembling our user, with the outfit recommended by our model. The output will be the url link for our image. This function is url_image.

We've also created the display_image function, which takes a prompt as input and passes it to the url_image function to retrieve the url. With this url, the function will be able to display the image in our console.

IV. Chatbot Class

The `FashionBot` class has three dictionaries. "regex_patterns" is a dictionary that associates intentions (such as age, gender, origin, etc.) with regular expressions to detect these intentions in user questions. For example, it looks for words like "age" or "which age" to determine that the intention is related to age.

"Responses" is another dictionary that associates intentions with lists of possible responses that the chatbot can give when it detects that intention. For example, if it detects the "age" intention, it might respond with one of the sentences specified in the response list.

"user_responses" is an empty dictionary where user responses will be stored.

Then we have a method to retrieve questions and another one to initiate the conversation.

V. Prompt Engineering

Many functions require a prompt. We created this prompt using the error-proofing process. We had a target result and to get that result, we use promt to guide our models to get the information we need and have them return it to us in the right way.

We have two prompts in our code. The first is the one that will give advice and information directly to the user. The response must be effective, must get straight to the point and give concrete examples of outfits and improvements, taking into account the information given by the user.

The second prompt will use the information provided by the user and the advice and recommendations from ChatGPT's first prompt to create a prompt for DALL-E. To make this prompt we have given it a template to respect, a template that we have created, and through several orders and directions, the model must give us a prompt specific to this user and on the recommended outfit.

VI. Main & Display

Throughout the run of our main function, which will call on everything we've created previously, a number of things will be displayed so that the user can understand the advice being given.

Firstly, the discussion between the chatbot and the user is displayed, and when it's over, we display a summary of the discussion. Next comes the fashion advice on the style to adopt and the clothes to wear, which are communicated directly to the user, before being illustrated by a 1024x1024 image.

Conclusion

This project has been very rewarding for us because it touches on a subject we're passionate about, which is fashion. To do it we used what we've learnt but also things we've never done before, like prompt engineering.

We've created a chatbot that's useful, easy to use and uses the latest cutting-edge technologies that aren't always easily accessible to the average person.

With Bot Fashion, become fashionable.