Using ChatGPT API in Applications

Understanding Large language Models (LLM)

# Cover

Hi all. My name is Konstantin Voloshenko, I am a BA from St. Petersburg.

One of my hobbies is neural networks.

Last time I told you about neural networks from three sides: how it looks for a PM, for a BA, and for a developer.

Today I will talk about how to integrate ChatGPT into your application.

# Agenda

You can see the Agenda. In addition to the story, today there will be a demonstration of a prototype solution and everyone will receive a link to the source codes of the examples.

# Overview. What is and how ChatGPT works

ChatGPT is a language model developed by OpenAI. It uses a technique called Generative Pre-trained Transformer (GPT) to generate human-like text responses based on the input it receives. The model has been trained on a massive dataset of text from the internet, allowing it to learn patterns, grammar, and even some level of understanding of different topics. It operates in a conversational manner, where users can provide prompts or questions, and ChatGPT generates a relevant response. However, it's important to note that while ChatGPT is generally helpful, clever, and friendly, it may sometimes produce incorrect or nonsensical information. In such cases, it's always good to verify the responses from reliable sources.

# Step 1: Simple Request. Using ChatGPT API in Applications

Using the ChatGPT API allows you to integrate ChatGPT's capabilities directly into your application or software. With the API, you can access the powerful language model and utilize its conversational abilities to provide value to your users. Instead of relying on a separate interface, the API enables you to seamlessly incorporate ChatGPT into your existing system, making it more interactive and engaging for users. Additionally, the API provides a streamlined and efficient way of sending and receiving requests, enabling real-time interactions with the model. By using the ChatGPT API, you can leverage the full potential of ChatGPT within your application and deliver a more personalized and dynamic user experience.

Let's see how to interact with GPT through the API.

First, we will pass a message to the model and ask it to generate a response.

ChatGPT uses the concept of "roles" for different participants in the dialogue. Each message in the messages list is a dictionary with two fields: role and content. Role can take one of three values: "system", "user", or "assistant", indicating who is "speaking" the message. Content contains the actual message.

In the standard case, three main roles can be included in the chat:

* "system": The system role is used to set the context of the dialogue. This is like a general instruction for the model, defining its behavior during the dialogue. Messages from "system" are usually placed at the beginning of the dialogue and may contain information on how the model should behave as an assistant.
* "user": This role is for the human user who is engaging in the dialogue with the model. Your requests to the model will be set under this role.
* "assistant": This role is for the AI model that responds to user queries.

When creating a chat with the model, you provide a list of messages. Each message in this list is a dictionary containing two elements: "role" and "content". "Role" indicates the role (mentioned above: "system", "user", or "assistant"), and "content" contains the actual text of the message.

# Step 2: Prompt and Knowledge base

## Prompt

When creating prompts for the ChatGPT model, there are several principles that can help you achieve better results:

* **Specificity:** The more specific your prompt, the more specific answer you will receive. Your instructions should be clear and specific. If the instructions are too general or ambiguous, the model may not understand what you want from it.
* **System message**: Using a system message to indicate the model's behavior can be very useful. For example, you can specify in the system message: "You are an assistant specializing in art history, always striving to provide detailed and accurate answers." This message sets the context for the model. A system message allows you to set tasks or give instructions to the model. The model sees this message, but it is not considered part of the dialogue with the user.
* **Instructions in the user message:** A brief instruction can also be placed in the user message. For example: "Answer the customer's question based on the information provided to you."

## Knowledge base

The structure of the company's knowledge base, which is designed to provide answers to client questions in a text format, is a key aspect of its effective functioning. Remember that chunks most relevant to the user's query will be used for answering questions. Thus, the knowledge base structure should be organized, logical, and convenient for information retrieval in order to ensure maximum usefulness and completeness of information in the chunks presented for analysis. Here are several key principles to consider when forming the knowledge base structure:

* Categorization and classification: The knowledge base should be divided into clear and logically organized blocks corresponding to different knowledge areas or types of client questions. For example, these could be sections related to products, services, support, payment issues, etc. Each block can have sub-blocks for further detail. This will allow LangChain to easily find relevant chunks corresponding to the query.
* Hierarchy: The knowledge base can be organized in a hierarchical structure where broader topics are located at the top level and more specific questions and answers are found at lower levels. This makes it easier to control the relevance of information in the knowledge base and update it as needed.

# Step 3: TG bot and all together

Telegram bot and all together

The diagram illustrates the process of integrating a Telegram Bot and ChatGPT to provide responses to user questions. The diagram shows the flow of information and interactions between different components.

1. The Application downloads two files from Google Drive: one file contains prompt, and the other contains a knowledge base.
2. The Application splits the knowledge base into chunks using a CharacterTextSplitter component.
3. The Application creates an index database and converts the knowledge base fragments into embeddings.
4. The diagram shows a group labeled "async text(update, context) # TG bot function for text messages" which represents the asynchronous process of handling text messages in the Telegram Bot. Within this group, the following steps occur:

* The User sends a question to the Telegram Bot.
* The Application receives the user's question.
* The Application searches for relevant chunks in the knowledge base based on the user's question.
* The Application sends a request to ChatGPT, including the prompt, message content (question and relevant chunks), and a temperature parameter controlling the randomness of the response.
* ChatGPT generates a response and sends it back to the Application.

1. The response from ChatGPT is then sent back

# Summary

1. ChatGPT API is key point to integrated application.
2. Need your own Knowledge Base.
3. Telegram bot is the easiest integration.
4. Other LLM is Llama models family: Llama-cpp, Alpaca, Saiga, Vicuna.

In addition to ChatGTP, there are other LLMs, the most famous is a Llama models family: Llama-cpp, Alpaca, Saiga, Vicuna.

# Unsorted

**Calling ChatGPT via API. Roles. Influence of the instruction on the response of the model.**

I want to make a speech for programmers. I have a plan. Check this plan and change it if you can better. Here is my plan:

* What is and how ChatGPT works.

ChatGPT API is key point to integrated application.

Need your own Knowledge Base.

Telegram bot is the easiest integration.

Other LLM is Llama models family: Llama-cpp, Alpaca, Saiga, Vicuna.

What is and how ChatGPT works

* The value of the ChatGPT API
* Need your own Knowledge Base
* Putting it all together

Your plan looks great! It covers the main points about ChatGPT and the importance of using the ChatGPT API. However, I have a few suggestions to enhance it:

1. Introduction: Begin your speech by introducing the audience to the topic of language models and their applications in programming. Highlight the significance of natural language processing and the need for conversational interfaces.
2. What is and how ChatGPT works: Maintain this as the first point in your plan. Explain in simple terms what ChatGPT is, its underlying technology (Generative Pre-trained Transformer), and how it generates human-like text responses.
3. The value of the ChatGPT API: After explaining ChatGPT, elaborate on the role of the API in integrating ChatGPT capabilities into applications. Discuss the benefits of real-time interactions, seamless integration, and deliver a more personalized user experience.
4. Utilizing a Knowledge Base: Add a section to explain the importance and benefits of having a well-curated knowledge base. Discuss how combining ChatGPT's abilities with a knowledge base can provide accurate and reliable information to users.
5. Putting it all together: Keep this section as it is, but consider giving practical examples or use cases of how ChatGPT API can be utilized in different programming scenarios.
6. Introduce other Llama models: You mentioned Llama models in the plan, but it might be helpful to briefly explain what Llama models are, their purpose, and how they relate to Chat

who won the f1 championship in 2021?

This is a polite but incorrect answer.

Привет! У меня накопился новый материал про AI, на этот раз про ChatGPT - как ее интегрировать через API в свое приложение. Могу выступить с докладом 20-25 минут с показом готового прототипа. Презу прилагаю:

Creating Telegram Bots with ChatGPT:

Integration of Telegram bot and ChatGPT

# Prompt

You are a technical writer. Rewrite this text briefly in good technical English. The text is for public speaking. Here is the text: