Building Generative AI-Powered Chat Applications

Introduction

This lesson covers:

- Techniques for efficiently building and integrating chat applications.
- How to apply customization and fine-tuning to applications.
- Strategies and considerations to effectively monitor chat applications.

Learning Goals

By the end of this lesson, you'll be able to:

- Describe considerations for building and integrating chat applications into existing systems.
- Customize chat applications for specific use-cases.
- Identify key metrics and considerations to effectively monitor and maintain the quality of AI-powered chat applications.
- Ensure chat applications leverage AI responsibly.

Integrating Generative AI into Chat Applications

Chatbot or Chat application?

Chatbot	Generative AI-Powered Chat Application
Task-Focused and rule based	Context-aware
Often integrated into larger systems	May host one or multiple chatbots
Limited to programmed functions	Incorporates generative AI models
Specialized & structured interactions	Capable of open-domain discussions

Leveraging pre-built functionalities with SDKs and APIs

- Expedites the development process and reduces overhead
- Better performance
- Easier maintenance
- Access to cutting edge technology

User Experience (UX)

General UX principles apply to chat applications, but here are some additional considerations that become particularly important due to the machine learning components involved.

- **Mechanism for addressing ambiguity**
- **Context retention**
- **Personalization**

Microsoft's System Message Framework for Large Language Models

Microsoft has provided guidance for writing effective system messages when generating responses from LLMs broken down into 4 areas:

1. Defining who the model is for, as well as its capabilities and limitations.
2. Defining the model's output format.
3. Providing specific examples that demonstrate intended behavior of the model.
4. Providing additional behavioral guardrails.

Accessibility

Whether a user has visual, auditory, motor, or cognitive impairments, a well-designed chat application should be usable by all. The following list breaks down specific features aimed at enhancing accessibility for various user impairments.

- **Features for Visual Impairment**: High contrast themes and resizable text, screen reader compatibility.
- **Features for Auditory Impairment**: Text-to-speech and speech-to-text functions, visual cues for audio notifications.
- **Features for Motor Impairment**: Keyboard navigation support, voice commands.
- **Features for Cognitive Impairment**: Simplified language options.

Customization and Fine-tuning for Domain-Specific Language Models

Imagine a chat application that understands your company's jargon and anticipates the specific queries its user base commonly has. There are a couple of approaches worth mentioning:

- **Leveraging DSL models**. DSL stands for domain specific language. You can leverage a so called DSL model trained on a specific domain to understand it's concepts and scenarios.
- **Apply fine-tuning**. Fine-tuning is the process of further training your model with specific data.

Customization: Using a DSL

Leveraging a domain-specific language models (DSL Models) can enhance user engagement and by providing specialized, contextually relevant interactions. It's a model that is trained or fine-tuned to understand and generate text related to a specific field, industry, or subject. Options for using a DSL model can vary from training one from scratch, to using pre-existing ones through SDKs and APIs. Another option is fine-tuning, which involves taking an existing pre-trained model and adapting it for a specific domain.

Customization: Apply fine-tuning

Fine-tuning is often considered when a pre-trained model falls short in a specialized domain or specific task.

For instance, medical queries are complex and require a lot of context. When a medical professional diagnoses a patient it's based on a variety of factors such as lifestyle or pre-existing conditions, and may even rely on recent medical journals to validate their diagnosis. In such nuanced scenarios, a general-purpose AI chat application cannot be a reliable source.