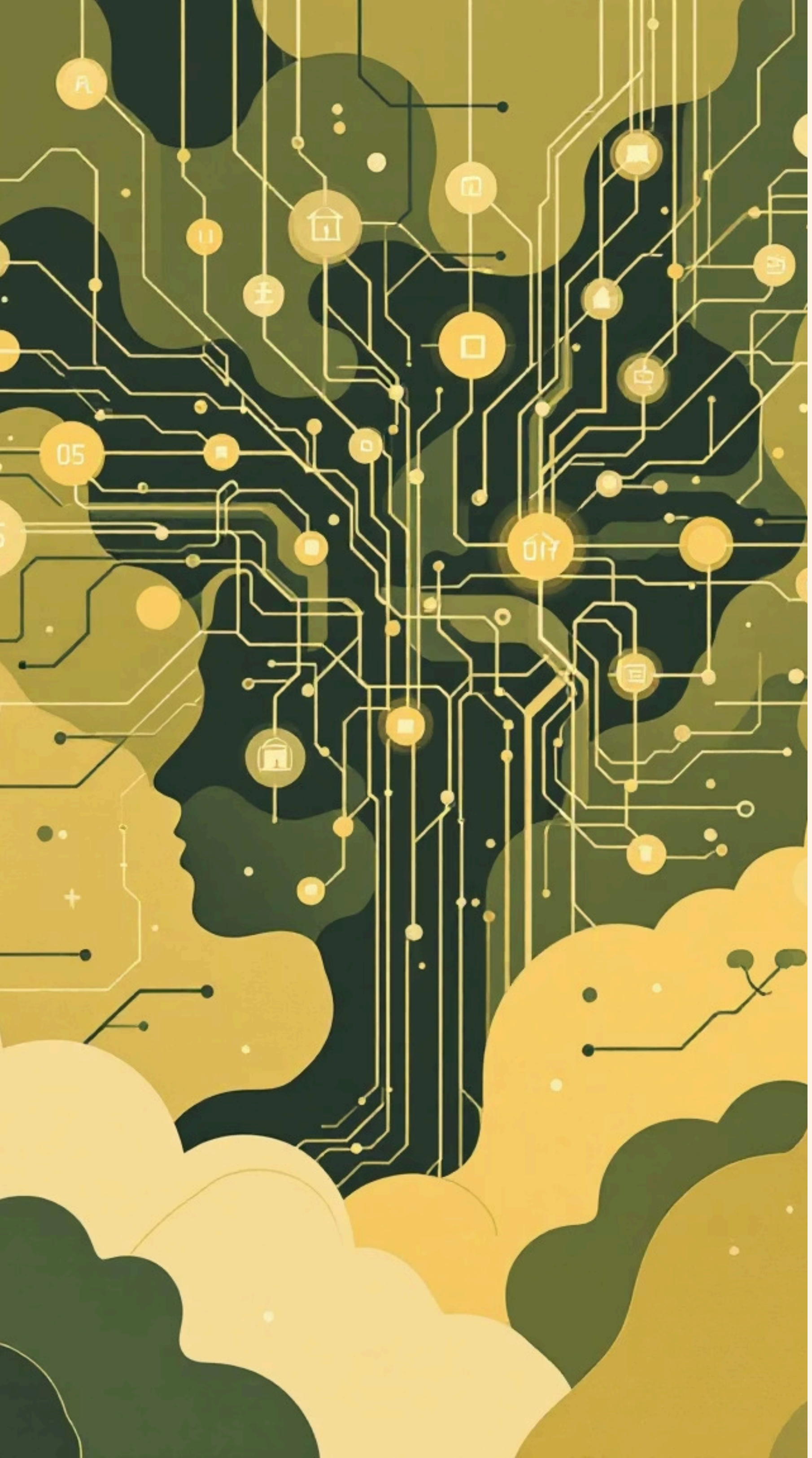
Project KIIT Chatbot: Leveraging LLMs for Intelligent Conversations

Discover how we built an advanced chatbot using cutting-edge Large Language Models and innovative AI techniques.



The Foundation: Large Language Models (LLMs)

Our project is built upon the robust capabilities of Large Language Models, the backbone of modern conversational AI.

Generative AI Foundations

Understanding the core principles of how LLMs generate human-like text and learn from vast datasets.

HuggingFace Model Selection

Choosing powerful models like GPT-Neo, FLAN-T5, LLaMA, and Mistral for optimal performance and efficiency.

Model Inference

Implementing efficient inference using pipelines or custom transformers for real-time response generation.

Enhancing LLM Power with LangChain

The LangChain Framework provided the essential tools to orchestrate complex LLM interactions and build sophisticated conversational flows.

LLM Wrappers

Seamless integration with HuggingFaceHub and HuggingFacePipeline.

Prompt Templates

Crafting dynamic and reusable prompt structures for diverse queries.

Chains

Orchestrating multi-step operations using SimpleSequentialChain and LLMChain.

Memory Management

Implementing robust memory solutions: ConversationBufferMemory, ConversationKGMemory, ConversationSummaryMemory.

The Art of Prompt Engineering

Carefully designed prompts unlock the full potential of LLMs, guiding them to generate precise and controlled responses.

Dynamic Prompt Templates

Utilizing variables to personalize and contextualize chatbot interactions effectively.

Zero, One, Few-Shot Prompts

Strategically employing different prompting techniques for varying levels of model guidance.

Role-Based Messaging

Defining System, Human, and AI message roles to structure conversations naturally.

Instructional Design

Crafting clear instructions to ensure predictable and controlled LLM outputs.

Deploying Models with HuggingFace

Harnessing the HuggingFace ecosystem for efficient model deployment and optimized inference.

Transformers Library

Leveraging the comprehensive HuggingFace transformers library for model management.

Text Generation Pipeline

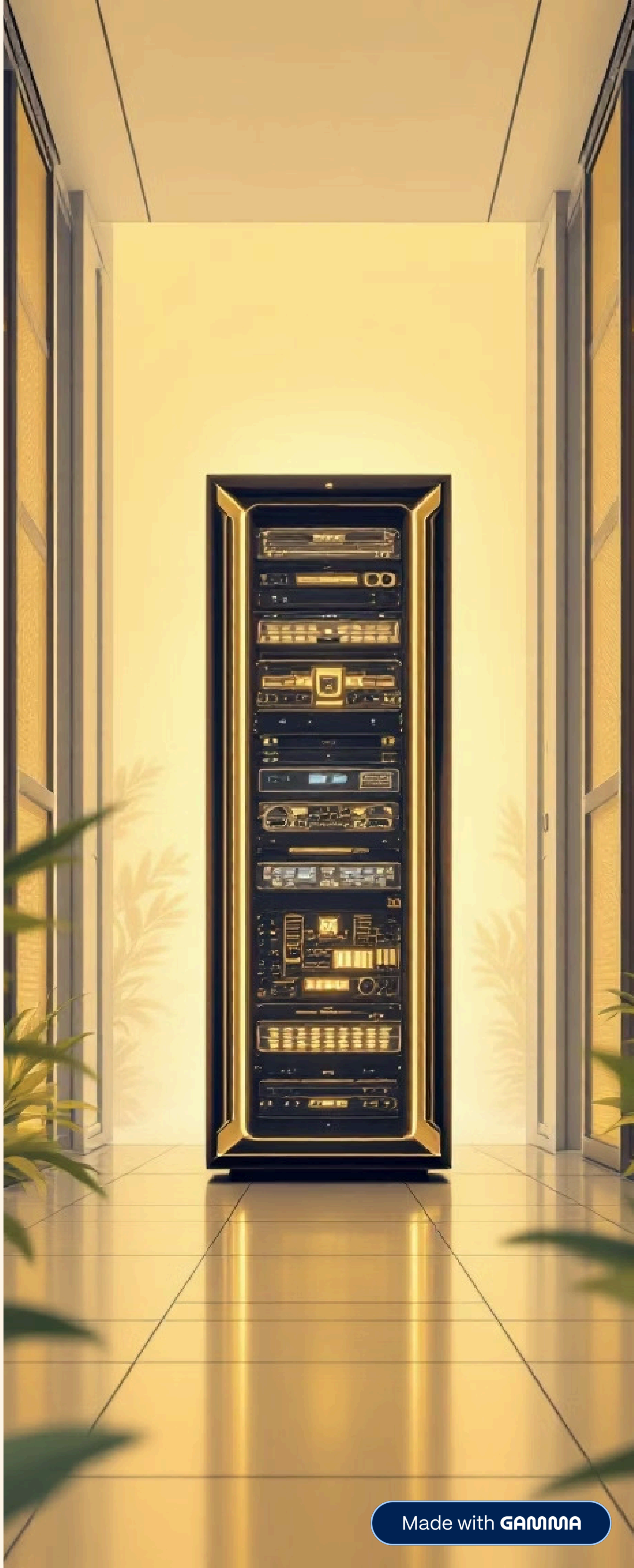
Streamlining the process of generating text with pre-built pipelines.

Tokenizer & Model Loading

Ensuring accurate tokenization and efficient loading of chosen LLMs.

Inference Optimization

Fine-tuning for performance on both GPU and CPU environments for speed and scalability.



Vector Databases & Embeddings for Context

Integrating vector databases and embeddings is crucial for providing our chatbot with long-term memory and contextual understanding.

Embedding Models

Utilizing Sentence Transformers and BERT variants to create meaningful data representations.

Creating Embeddings

Transforming raw text into numerical vector embeddings for efficient similarity search.

Vector DB Storage

Storing embeddings in specialized databases like FAISS and ChromaDB.

Semantic Search

Enabling intelligent context retrieval based on semantic similarity, not just keywords.

Retrieval-Augmented Generation (RAG)

When accuracy and up-to-date information are paramount, RAG empowers the chatbot to answer from specific documents.

Document Loading

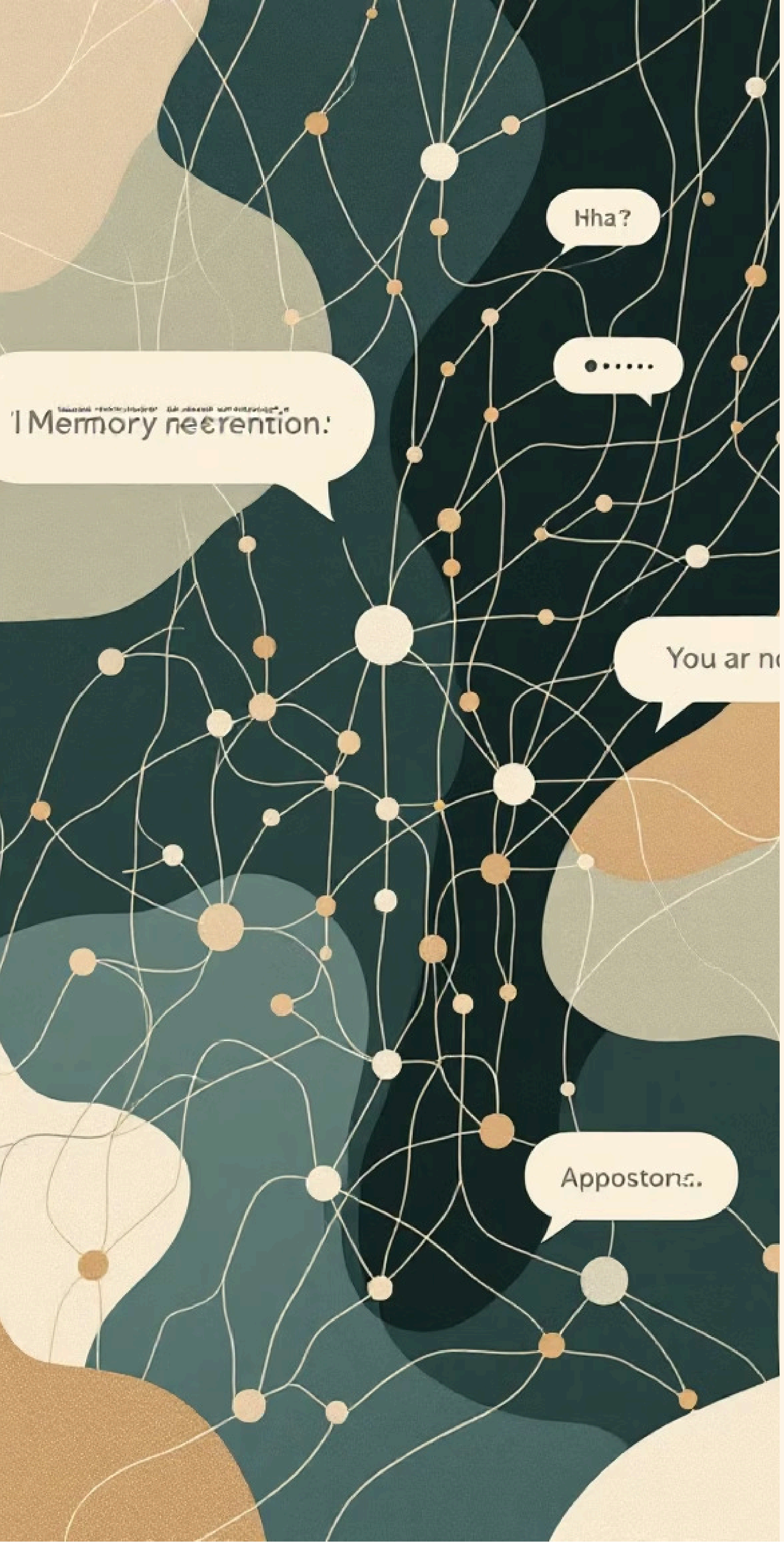
Ingesting data from diverse sources: PDFs, text files, and web content.

Text Splitting

Breaking down documents into manageable chunks using RecursiveCharacterTextSplitter.

Retrieval Chain

Implementing RetrievalQA and ConversationalRetrievalQA for intelligent information retrieval.



Sustaining Dialogue: Conversation Memory

Effective memory management is key to maintaining coherent and personalized long-running conversations.

Buffer Memory

Storing recent turns of a conversation for immediate context recall.

Summary Memory

Condensing longer conversations into concise summaries to preserve context over time.

Token Efficiency

Optimizing memory usage for long conversations to manage LLM token limits.

Prompt Integration

Seamlessly incorporating chat history into prompts for contextual responses.

The Chatbot Pipeline: From Input to Response

A clear, structured pipeline ensures smooth processing from user query to intelligent chatbot response.

User Input

Capturing and initiating the user's query.

Preprocessing

Cleaning and preparing the input for further processing.

Context Retrieval

Fetching relevant information (if RAG is employed).

Prompt Generation

Constructing the final prompt for the LLM.

LLM Response

Generating the chatbot's intelligent reply.

Post-processing

Refining and formatting the LLM output for the user.

Future Directions & Impact

Project KIIT Chatbot exemplifies the power of integrating advanced LLMs and thoughtful design for impactful conversational AI solutions.

Enhanced User Experience

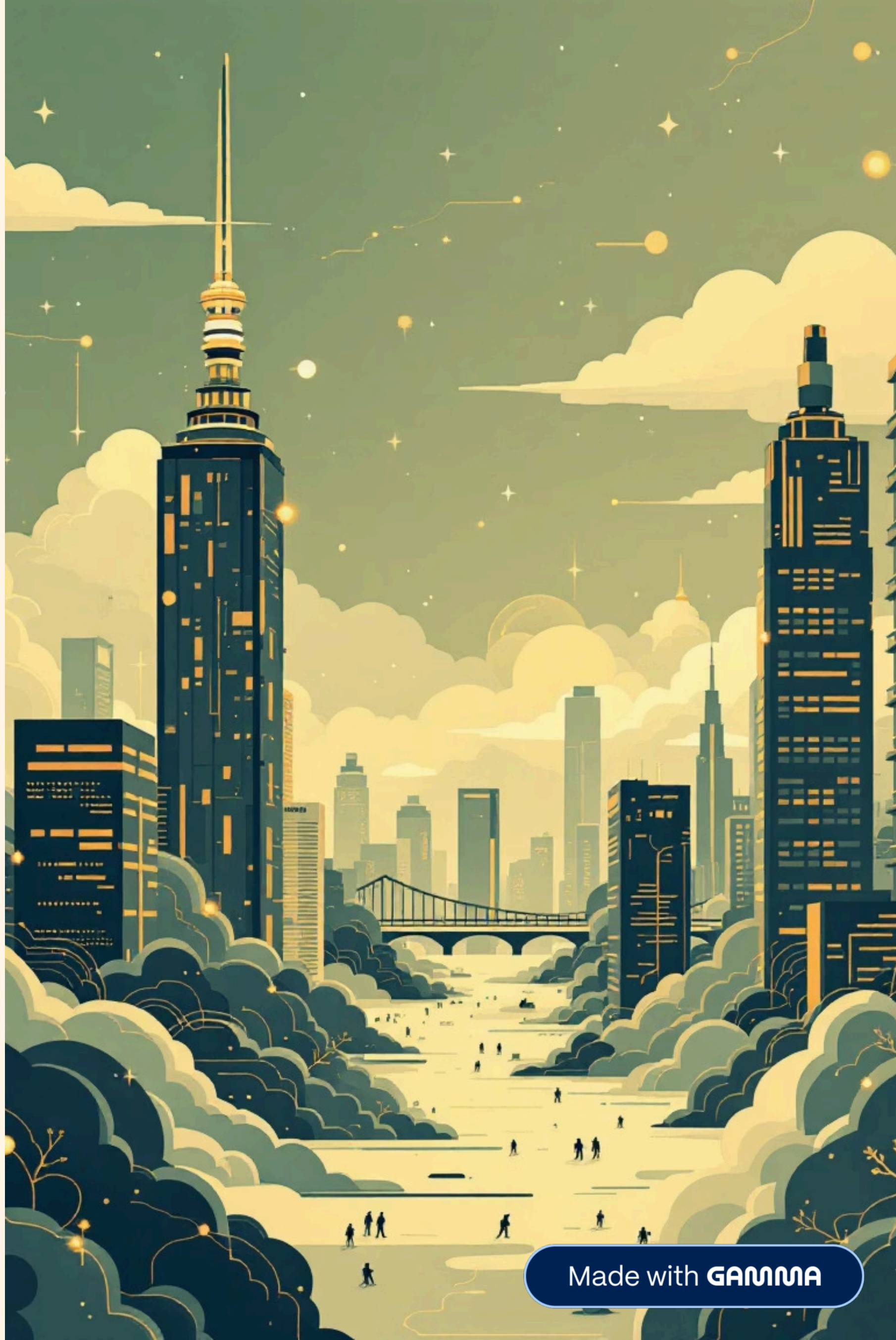
Creating more natural, helpful, and efficient user interactions.

Scalable Architecture

Designing a system ready to handle growing demands and complex queries.

Continuous Improvement

Setting the stage for ongoing model fine-tuning and feature expansion.



Thank You

Submitted to **Dr. Saswati Patra**

Submitted by:

- Om Prakash Pradhan (2205996)
- Amit Kumar Biswal (2205612)
- Vaibhav Tomar (22051299)
- Swaraj Kumar Mallick (22054178)
- Manoj (22051435)