IBM Innovation Center for Education

PHEMESOFT Software That Matters
TRUSTED DELIVERY PARTNER for IBM ICE

YENEPOYA INSTITUTE OF ARTS, SCIENCE, COMMERCE AND MANAGEMENT
A CONSTITUENT UNIT OF YENEPOYA (DEEMED TO BE UNIVERSITY)
BALMATTA, MANGALORE

CYBERSECURITY AWARNESS CHATBOT

PROJECT SYNOPSIS

CYBERSECURITY AWARNESS CHATBOT

BACHELOR OF SCIENCE

Cyber Forensics, Data analytics & Cyber Security

SUBMITTED BY:

Ahzaaf S 21613 (Team Leader)

Adarsh A 21592

Kiran A L 21596

Devika Murali 22172

GUIDED BY

Mr. Shashank

PROJECT: Cybersecurity Awareness Chatbot

Synopsis

The Cybersecurity Awareness Chatbot is an AI-powered conversational agent developed to enhance public and organizational understanding of cybersecurity principles, threats, and best practices. The tool aims to provide an accessible, interactive, and reliable source of information, making complex cybersecurity concepts easier to grasp for users with varying levels of technical expertise. It utilizes a Retrieval Augmented Generation (RAG) architecture to deliver accurate and contextually relevant information.

The primary goal of the Cybersecurity Awareness Chatbot is to empower users with the knowledge needed to protect themselves and their digital assets by automating the dissemination of curated cybersecurity information and providing instant, understandable answers to their queries. It performs the following key operations:

- Knowledge Base Ingestion and Vectorization: Processes a curated collection of cybersecurity awareness documents (PDFs). The textual content is extracted, segmented into manageable chunks, and then converted into numerical vector embeddings using a sophisticated NVIDIA embeddings model. These embeddings, representing the semantic meaning of the text, are stored and indexed in a ChromaDB vector database.

- User Query Understanding and Semantic Retrieval: When a user poses a question, the chatbot converts the natural language query into a vector embedding using the same NVIDIA model. It then performs a semantic similarity search within the ChromaDB to retrieve the most relevant text chunks from the ingested knowledge base. This ensures that the information provided is grounded in authoritative sources.

- Retrieval Augmented Generation (RAG) for Enhanced Responses: The retrieved relevant text chunks, along with the user's original query, are passed as context to Google's Gemini API. Langchain orchestrates this process. The Gemini API, a powerful large language model, then synthesizes this information to generate a comprehensive, coherent, and "supercharged" answer that is not only accurate but also explanatory and engaging.

- Langchain Orchestration: Utilizes the Langchain framework to seamlessly connect and manage the various components of the RAG pipeline, including the user interface, embedding model, vector store (ChromaDB), and the generative LLM (Gemini API), ensuring efficient data flow and interaction logic.

The field of this project is at the intersection of Artificial Intelligence (specifically Natural Language Processing and Conversational AI) and Cybersecurity Education. Key technical terms include Retrieval Augmented Generation (RAG), Large Language Models (LLMs), Vector Embeddings, Semantic Search, Vector Databases (ChromaDB), NVIDIA Embeddings, Gemini API, and Langchain.

TABLE OF CONTENTS

1. INTRODUCTION

The proliferation of digital technologies has made cybersecurity awareness paramount for individuals and organizations alike. However, effectively disseminating complex cybersecurity information remains a challenge. This project proposes the development of a "Cybersecurity Awareness Chatbot," an intelligent conversational agent designed to educate users on various cybersecurity topics in an interactive and engaging manner.

The core of this project lies in the implementation of a Retrieval Augmented Generation (RAG) architecture. This approach combines the strengths of large language models (LLMs) with information retrieval from a specialized knowledge base. Specifically, the chatbot will leverage a curated corpus of cybersecurity awareness PDFs. These documents will be processed and stored as vector embeddings in a ChromaDB vector store, using an NVIDIA embeddings model for efficient and semantically rich representation.

When a user poses a query, it will be converted into an embedding. This embedding will then be used to retrieve the most relevant information chunks from ChromaDB. These retrieved chunks, along with the original query, will be fed into Google's Gemini API. The Gemini API, acting as the generative component, will synthesize this information to provide an accurate, contextually relevant, and "supercharged" response, going beyond simple information retrieval by offering explanations, examples, and actionable advice. Langchain will serve as the orchestration framework, seamlessly connecting these components – user query, embedding model, vector store, and the LLM.

The field of this project is at the intersection of Artificial Intelligence (specifically Natural Language Processing and Conversational AI) and Cybersecurity Education. Key technical terms include **Retrieval Augmented Generation (RAG)**, **Large Language Models (LLMs)**, **Vector Embeddings**, **Semantic Search**, and **Vector Databases**. The primary objective is to create a reliable, informative, and user-friendly tool that enhances cybersecurity literacy.

2. OBJECTIVES

The development of the Cybersecurity Awareness Chatbot is guided by key objectives aimed at enhancing the accessibility, accuracy, and engagement of cybersecurity education. Each objective is designed to address the practical need for improved cyber literacy among a broad audience and to support the system's future growth and relevance.

1. **Automating and Simplifying Access to Cybersecurity Knowledge**
   One of the primary goals is to significantly reduce the barriers users face when seeking cybersecurity information. Traditional methods often involve sifting through dense documentation or multiple online sources. The Cybersecurity Awareness Chatbot automates the process of finding relevant information within its curated knowledge base (derived from trusted PDFs) and presents it in an easily digestible, conversational format, thereby saving user time and effort.

2. **Ensuring Accurate and Reliable Information Delivery through Retrieval Augmented Generation (RAG)**
   To combat misinformation and the potential for LLM "hallucinations," a core objective is to provide trustworthy and verifiable answers. By implementing a RAG architecture, the chatbot grounds its responses in factual content retrieved from the specialized cybersecurity PDF corpus stored in ChromaDB. The Gemini API then enhances this retrieved information, ensuring that the answers are not only accurate but also contextually rich and comprehensive, rather than speculative.

3. **Providing an Interactive and Engaging Learning Experience**
   Moving beyond static FAQs or passive reading, the chatbot aims to foster active learning through an interactive conversational interface. Users can ask follow-up questions, seek clarifications, and explore topics in a dynamic way. This engagement is intended to improve knowledge retention and make the learning process more appealing and effective for users of all technical backgrounds.

4. **Designing a Scalable and Maintainable Knowledge Architecture**
   The chatbot is being built with future adaptability in mind. The cybersecurity landscape is constantly evolving, so the system's architecture, leveraging Langchain for orchestration and ChromaDB for vector storage, is designed to allow for straightforward updates and expansion of its knowledge base. New cybersecurity awareness documents can be easily processed and integrated, ensuring the chatbot remains a current and relevant educational resource over time.

3.METHODOLOGY/DEVELOPMENT PLAN

The development of the Cybersecurity Awareness Chatbot will follow a structured approach, encompassing the following key phases:

1. Data Collection and Preparation:
    - Gathering relevant and up-to-date cybersecurity awareness documents, primarily in PDF format, from reputable sources (e.g., government agencies, cybersecurity firms, academic institutions).
    - Preprocessing the PDFs: Extracting text, cleaning data, and segmenting content into manageable chunks suitable for embedding.
2. Vector Database Setup and Population:
    - Setting up ChromaDB as the vector store.
    - Utilizing an NVIDIA embeddings model (e.g., through NVIDIA NIM or an appropriate API) to convert the prepared text chunks into high-dimensional vector embeddings.
    - Storing these embeddings and their corresponding text in ChromaDB, creating a searchable knowledge base.
3. RAG Pipeline Implementation with Langchain:
    - Using Langchain as the orchestration framework to build the RAG pipeline.
    - Developing the retrieval mechanism: When a user query is received, it will be converted into an embedding using the same NVIDIA model. Langchain will facilitate a semantic similarity search in ChromaDB to fetch the most relevant document chunks.
    - Constructing the prompt for the LLM, incorporating the retrieved context and the user's original query.
4. Large Language Model (LLM) Integration:
    - Integrating Google's Gemini API as the generative component of the RAG system.
    - Sending the constructed prompt (query + retrieved context) to the Gemini API via Langchain.

- Receiving the "supercharged," contextually aware, and informative response generated by Gemini.

5. User Interface (UI) Development (Basic):

    - Developing a simple web-based or command-line interface for users to interact with the chatbot, allowing them to ask questions and receive answers.

6. Testing and Evaluation:

    - Conducting thorough testing to assess the chatbot's accuracy, relevance of responses, and user experience.

    - Iteratively refining the data, embeddings, retrieval strategy, or prompts based on test results.

4. TOOLS AND TECHNOLOGIES USED

The development of **ChatBot** relies on a combination of software and hardware resources that together support its full-stack functionality. The tool is designed to be lightweight, portable, and accessible on standard computing setups.

Software:

- Programming Language: Python (Version 3.8 or higher)
- Core Libraries:
    - Langchain: For orchestrating the RAG pipeline.
    - ChromaDB: For vector storage and retrieval.
    - google-generativeai: Python SDK for Gemini API.
    - NVIDIA AI SDK/API client: For accessing NVIDIA embedding models (e.g., sentence-transformers if using a compatible open model, or specific NVIDIA client libraries).
    - pypdf2 or PyMuPDF (fitz): For PDF text extraction.
    - nltk or spaCy: For text processing and chunking (optional, Langchain offers utilities).
- Development Environment:
    - IDE: VS Code, PyCharm, or Jupyter Notebooks.
    - Operating System: Windows, macOS, or Linux.
- API Access:
    - Google Gemini API Key.
    - Access to NVIDIA embedding models (potentially via API key if using a cloud service, or local setup if using self-hosted models like NIM).
- Version Control: Git / GitHub.

Hardware:

- Computer System: A standard desktop or laptop computer.
    - Processor: Intel Core i5/AMD Ryzen 5 or equivalent (minimum).
    - RAM: Minimum 8GB RAM (16GB recommended for smoother local operations, especially if handling larger datasets or models).
    - Storage: Sufficient disk space for project files, Python environment, and local ChromaDB instance (e.g., 50-100GB SSD recommended).
- Internet Connection: Stable internet access for API calls, library downloads, and research.

- GPU (Optional but Recommended): If performing embedding generation locally with larger NVIDIA models, a dedicated NVIDIA GPU would significantly speed up the process. However, if relying on API-based embedding services, this is less critical.

7.REFERENCES

The development of this project will refer to a variety of study materials, including but not limited to:

- Official documentation for
  - Langchain- https://python.langchain.com/docs/get_started/introduction
  - ChromaDB : https://docs.trychroma.com/
  - Google Gemini API-https://ai.google.dev/docs/gemini_api_overview
  - NVIDIA embedding models: - https://developer.nvidia.com/nemo-framework

- Academic papers on topics like Retrieval Augmented Generation (RAG) (e.g., Lewis et al., 2020), Large Language Models (LLMs), and vector embeddings. https://arxiv.org/abs/2005.11401 https://proceedings.neurips.cc/paper/2020/hash/6b493230205f780e1bc26945df7481e5-Abstract.html

- Cybersecurity awareness publications from organizations such as NIST, SANS Institute, and ENISA. https://www.nist.gov/cybersecurity