CSE 535: Information Retrieval – Project 3

Sri Guna Kaushik Undru
Shri Harsha Adapala Tirumala
Sailesh Reddy Sirigireddy

- **Introduction**

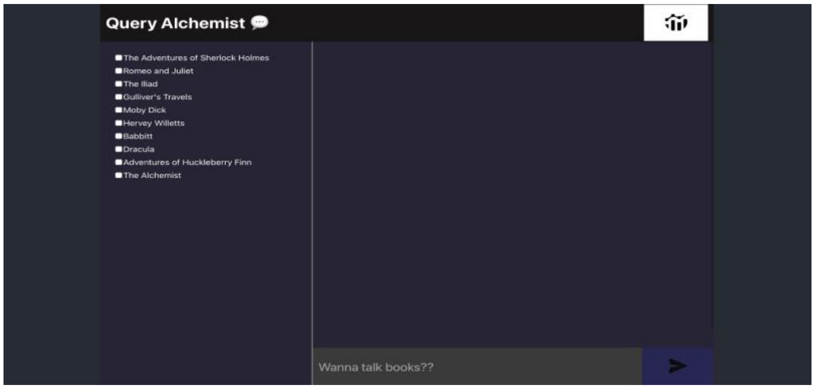
  Our project is an innovative Q/A (question and answer) system that also has a chat feature, designed especially for fans of classic books. It includes a collection of 10 well-known novels, like "The Adventures of Sherlock Holmes," taken from the Gutenberg Project. Our system uses advanced computer programs, Chatterbot, to have engaging, AI-powered conversations that are more advanced than basic chatbots. It has a smart feature that can figure out which novel a user is asking about, even if they don't mention it directly. The main part of the system is a Q/A bot that uses special technology to give answers that make sense in the context of the conversation. The system is designed to work smoothly, avoiding any crashes or problems during chats. It also has a user-friendly website where users can interact with the chatbot and see detailed analyses of their conversations, making the experience enjoyable and rich in literary content with advanced technology.
- **Methodology**:
  - Novel Collection and Processing:
    - We began by curating a dataset of classic novels, ensuring a diverse literary spectrum. "The Adventures of Sherlock Holmes" by Arthur Conan Doyle was chosen as a primary text, alongside nine other novels from the Gutenberg Project. This repository provided us with high-quality, public domain texts, which we processed into a machine-readable format for our system to access and analyze.
    - Here are the key points of the preprocessing method:
      - Text Cleaning: The text undergoes initial cleaning to remove various punctuation marks and special characters, such as single quotes, double quotes, underscores, dashes, and hyphens. This helps standardize the text for processing.
      - Tokenization and Stopword Removal: The cleaned text is tokenized into words, and stopwords (common words that often don't carry significant meaning, like "and", "the", etc.) and punctuation are removed. This focuses the dataset on more meaningful words for analysis.
      - Case Normalization: All tokens are converted to lowercase to ensure that the same words are recognized as identical, regardless of their case in the original text.
      - Lemmatization: Words are lemmatized, which means they're converted to their base or dictionary form. This process is more

sophisticated than stemming as it takes into consideration the context of the word to convert it to its meaningful base form.

    - Data Structure Creation: The code constructs a dictionary that maps book titles and authors to their respective paragraphs after preprocessing.
    - Text Segmentation: The text of each book is segmented into paragraphs, and these paragraphs are further chunked into groups of ten for manageable processing.
    - Metadata Extraction: For each book, the title and author are extracted from the text, assuming a specific structure in the text file.
    - Content Filtering: The text between the start and end markers of the Project Gutenberg eBook is extracted, eliminating headers and footers that are not part of the original book content.
    - DataFrame Compilation: The preprocessed paragraphs are then compiled into a panda DataFrame, with each row containing the title, author, and a combined string of ten paragraphs of text from the book.
    - Identification Indexing: Each paragraph is given a unique identifier (`para_id`) based on its index in the DataFrame for easy reference.
  - These preprocessing steps are designed to clean and prepare the text data, making it suitable for natural language processing tasks such as feeding into a machine learning model for a Q/A system.
- Language Model Integration:
  - For the chit-chat component, we integrated basic chatbot model called Chatterbot. We trained the model with default dataset provided along with the chatterbot package and augmented it with data to address farewell prompts and trained on it. Along with the default dataset we also trained it on chit-chat dataset to enhance their conversational abilities and nfl06 data set to give ability to answer some questions, ensuring they can engage users in natural and contextually relevant dialogues.
- Novel Classifier and Prompt Classifier:
  - To enable our system to identify and focus on specific novels within user queries, we developed a zero-shot language model classifier DeBERTa-v3-large-mnli-fever-anli-ling-wanli by MortizLaurer. We deployed it local to the backend server to speed up the response. We also used the same model to classify prompts – if it is intended for normal chit-chat or Information-Retrieval QA Bot. We also used same model to classify the intention of the prompt – if he is sending a farewell prompt to end the chat or his intention is to continue it.

- Q/A Bot Configuration:
    - The Q/A bot, which forms the core of our system, was set up using an Information Retrieval System and A RAG generative model. At first based on the input question we will pre-process it and extract relevant docs from IR system powered by SOLR. Later we will append all the documents separated by next line character. Then we will send all this to Hugging Face Embedding Models to extract complex word embeddings after breaking the whole IR output data into multiple chunks based on delimiters and character length. Then we will feed the extracted words embeddings of the IR data, input query and relevant prompt, that askes to answer the question with a word limit, to the OPEN AI Large Language Model, which will use the information provided via word embeddings and external database to answer the query that was inputted.
- Exception Handling Procedures:
    - When Chatbot couldn't answer a query, we are sending it to the RAG Large language Model to get an answer.
    - When we couldn't be able to get relevant documents to the question on the requested Novels, we are asking the user to change relevant Novel filter or rephrase the question in a more elegant way.
- User Interface Design and Hosting:
    - The user interface was designed to be intuitive and accessible, encouraging user interaction. The interface includes features for detailed chat analysis and visualization, offering users insights into their interactions with the system.
- Frond End Development:
    - The web application was developed using ReactJS, a popular JavaScript library to build dynamic and interactive user interfaces. React's component-based architecture facilitates the creation of interactive and efficient UIs, enhancing the overall user experience. This choice aligns with modern web development practices, enabling a dynamic and responsive application.
    - We used axios to make asycronus API calls to the main cloud application.
- Testing and Iteration:
    - Before launch, the system underwent rigorous testing to identify and rectify any issues. User feedback was instrumental in this phase, allowing us to iteratively improve the system's performance and user experience.

- **Sample Screenshots**:
    - Chat Bot initial UI

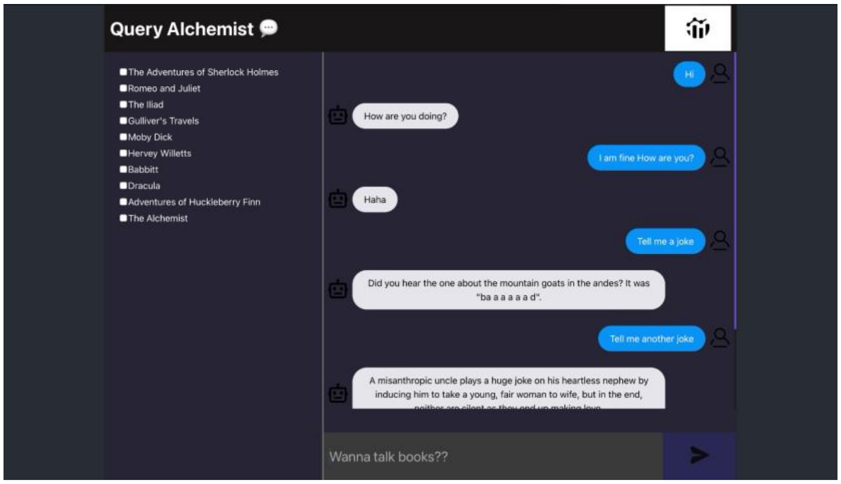


Query Alchemist

- The Adventures of Sherlock Holmes
- Romeo and Juliet
- The Iliad
- Gulliver's Travels
- Moby Dick
- Hervey Willetts
- Babbitt
- Dracula
- Adventures of Huckleberry Finn
- The Alchemist

Wanna talk books??

- User interacting with the Chat Bot:



Query Alchemist

- The Adventures of Sherlock Holmes
- Romeo and Juliet
- The Iliad
- Gulliver's Travels
- Moby Dick
- Hervey Willetts
- Babbitt
- Dracula
- Adventures of Huckleberry Finn
- The Alchemist

Hi

How are you doing?

I am fine How are you?

Haha

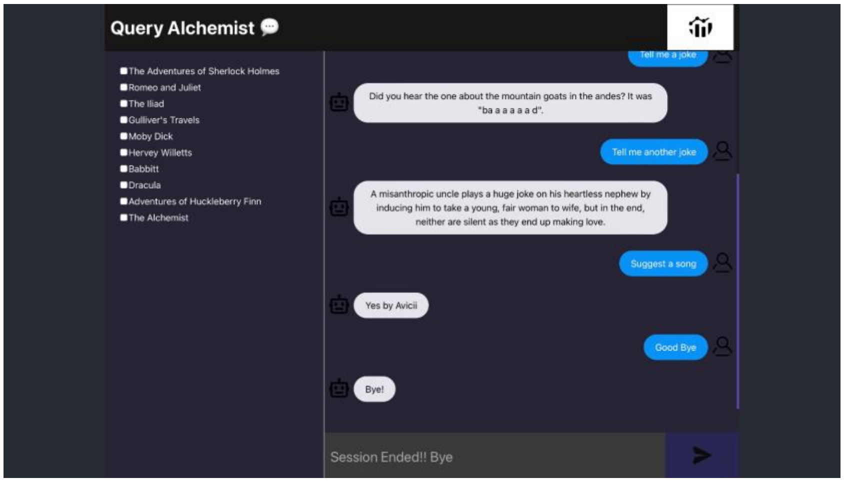
Tell me a joke

Did you hear the one about the mountain goats in the andes? It was "ba a a a a a d".

Tell me another joke

A misanthropic uncle plays a huge joke on his heartless nephew by inducing him to take a young, fair woman to wife, but in the end, neither are silent as they end up making love.

Wanna talk books??



Query Alchemist

- The Adventures of Sherlock Holmes
- Romeo and Juliet
- The Iliad
- Gulliver's Travels
- Moby Dick
- Hervey Willetts
- Babbitt
- Dracula
- Adventures of Huckleberry Finn
- The Alchemist

Tell me a joke

Did you hear the one about the mountain goats in the andes? It was "ba a a a a a d".

Tell me another joke

A misanthropic uncle plays a huge joke on his heartless nephew by inducing him to take a young, fair woman to wife, but in the end, neither are silent as they end up making love.

Suggest a song

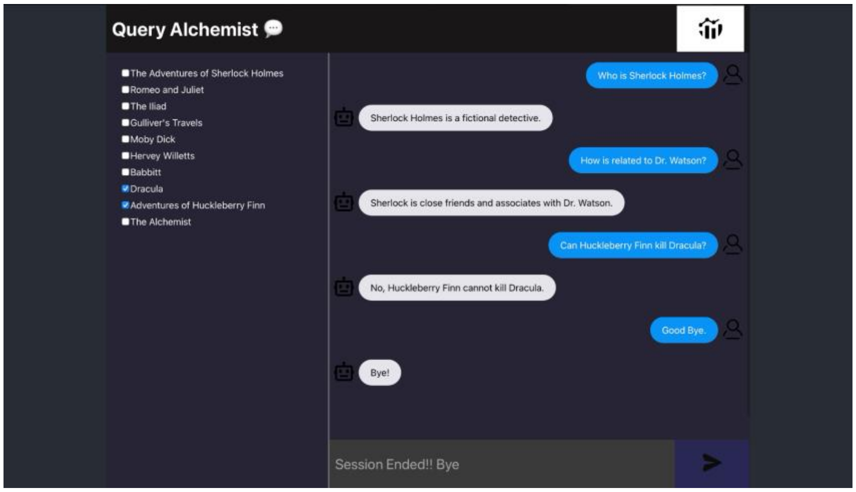
Yes by Avicii

Good Bye

Bye!

Session Ended!! Bye

  - User interacting with Chat Bot along with selecting Novels (multiple in this case):



□ The Adventures of Sherlock Holmes
□ Romeo and Juliet
□ The Iliad
□ Gulliver's Travels
□ Moby Dick
□ Hervey Willetts
□ Babbitt
☑ Dracula
☑ Adventures of Huckleberry Finn
□ The Alchemist

Who is Sherlock Holmes?

Sherlock Holmes is a fictional detective.

How is related to Dr. Watson?

Sherlock is close friends and associates with Dr. Watson.

Can Huckleberry Finn kill Dracula?

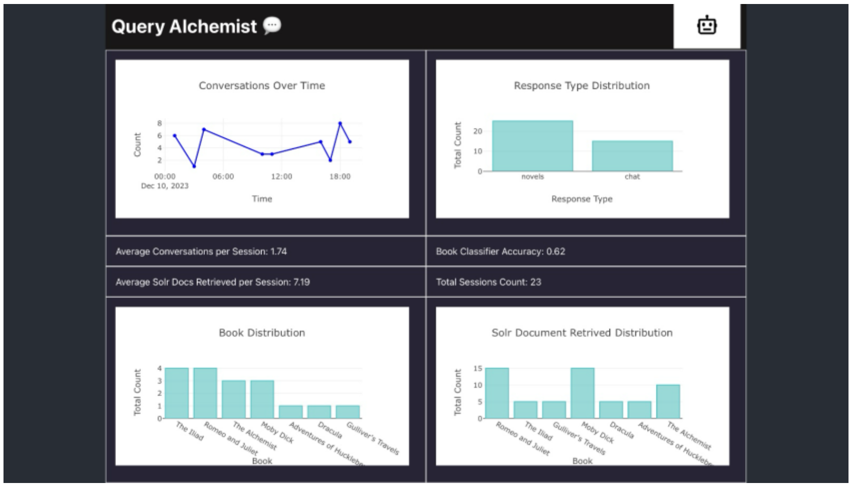
No, Huckleberry Finn cannot kill Dracula.

Good Bye.

Bye!

Session Ended!! Bye

  - Visualizations and analytics:



Average Conversations per Session: 1.74

Book Classifier Accuracy: 0.62

Average Solr Docs Retrieved per Session: 7.19

Total Sessions Count: 23

- **Work breakdown by teammates**:
  - Sri Guna Kaushik Undru
    - Datta Preprocessing the novels.
    - Analytics system design.
    - Indexing to Solr
    - Database architecture design for in-house analytics.
    - Exception Handling
    - Reporting and Documentation
  - Shri Harsha Adapala Tirumala
    - System Architecture Design
    - Zero-shot classification model training and deployment
    - Language Model (RAG) deployment and integration
    - Chit-Chat Model training and integration
    - Topic Analysis Algorithm

        - Exception Handling
    - Sailesh Reddy Sirigireddy
        - System Architecture Design
        - User Interface Design
        - Front-End Development
        - Back-End Server Development
        - Exception Handling
- **Conclusion:**
  In summary, we've built a chat system for classic book lovers, merging a selection of ten novels with the latest in language processing tech. This platform not only provides precise answers but also keeps users engaged in meaningful conversations. It smartly identifies the books being discussed and can handle complex inquiries, showcasing our effort to blend a love for literature with modern tech. User-friendly and accessible from anywhere, our system is set to grow, aiming to enrich user interaction and broaden its literary scope. Thanks to our team's hard work and user feedback, we've established a solid base for a tool that enriches the reading experience, a sign of the exciting possibilities at the crossroads of books and technology.
- **References:**

    - https://www.chatbot.com/academy/chatbot-designer-free-course/error-messages/
    - https://docs.python.org/3/tutorial/errors.html
    - https://chatterbot.readthedocs.io/en/stable/
    - https://github.com/BYU-PCCL/chitchat-dataset
    - https://www.gutenberg.org/
    - https://huggingface.co
    - https://legacy.reactjs.org/docs/getting-started.html