National University
of computer and emerging sciences

Assignment 03 Report

RAG Bot Implementation

Submitted By:

Azhar Ali (19I-0564)

1. Introduction

The implemented RAG (Retrieval-Augmented Generation) bot is designed to engage in coherent and contextually appropriate conversations in Urdu. It integrates state-of-the-art NLP techniques, Speech-to-Text (STT), and Text-to-Speech (TTS) modules. This report details the algorithms, data preprocessing steps, and module integration.

2. Algorithms and Models

2.1 Retrieval-Augmented Generation (RAG) Model

We employed the "facebook/rag-sequence-nq" model for RAG. It combines a sequence generation model with a Dense Passage Retriever (DPR) for efficient information retrieval. The model is fine-tuned on Natural Questions data, enhancing its performance in answering user queries.

2.2 Speech-to-Text (STT) Model

The STT module utilises the "facebook/hf-seamless-m4t-medium" model, which is a seamless and efficient Multimodal M4T (Many-to-Text) model. It converts spoken Urdu into text, enabling the RAG bot to understand user speech.

2.3 Text-to-Speech (TTS) Model

For TTS, we use the same "facebook/hf-seamless-m4t-medium" model. It synthesises contextually appropriate speech responses in Urdu based on the generated text.

3. Data Preprocessing

3.1 Urdu Dataset Cleaning

The provided Urdu dataset underwent a rigorous cleaning process. Urdu text was normalised by removing diacritics, punctuations, and numeric digits. Extra whitespaces were removed to enhance readability. The cleaned dataset is then tokenized for further processing.

4. Integration of STT and TTS Modules

4.1 Speech-to-Text Integration

The STT module is seamlessly integrated into the RAG bot. User speech input, recognized using the STT model, is converted to text. This text is then used as a query for the RAG model.

4.2 Text-to-Speech Integration

Generated responses from the RAG model are processed through the TTS module. The resulting text is transformed into natural-sounding speech, providing a coherent and contextually appropriate spoken output.

5. Running and Interacting with the Bot

1. Requirements:

- Install necessary Python packages: `transformers`, `torch`, `soundfile`, `flask`, `gtts`.

- Download and place the required models in the project directory.

2. Running the Bot:

- Execute `python app.py` in the terminal.

- Open a web browser and go to `http://localhost:5000/`.

3. Interacting:

- Type your queries in the input box.

- For speech input, click the microphone icon and speak into your microphone.

6. Conclusion

The implemented RAG bot showcases the seamless integration of advanced NLP models and modules for speech interaction. The system's modularity allows easy adaptation to other domains. The clean and organised codebase ensures maintainability and real-time response generation.