

Department of Artificial Intelligence and Data Science

“Intelligent Chatbot for Government Education Schemes”

Supervisor
Mr S Suresh kumar
M.E.,(Ph.D) Professor

Nithish S
221801502

Problem Statement and Motivation

- How can we create a chatbot that can provide reliable and relevant information about various government schemes to the citizens, and help them identify their eligibility criteria for different programs, using an intuitive interface, engaging conversation, and diverse scenarios?
- Many citizens may find it challenging to navigate complex government websites and documents. A chatbot can simplify access to information. Automated information dissemination reduces the burden on government offices and call centers. A chatbot can provide tailored information based on user inputs, improving the relevance of the information about the government schemes.

Objectives

- ❑ Creating a chatbot for government education scheme information and eligibility requires a robust framework. It should integrate reliable data sources to ensure accuracy and relevance in its responses. The interface must be intuitive, offering clear navigation and engaging dialogue to guide users effectively. Diverse scenarios should be anticipated and included in its training to handle varied queries and edge cases adeptly.
- ❑ Regular updates are crucial to keep pace with changing eligibility criteria and new programs. Incorporating user feedback mechanisms will further refine the bot's capabilities, enhancing its ability to provide personalized and valuable assistance to citizens seeking government scheme information.

Abstract

The development of a chatbot designed to provide citizens with reliable information about government education schemes, utilizing advanced natural language processing techniques, including BERT (Bidirectional Encoder Representations from Transformers), Retrieval-Augmented Generation (RAG), and Large Language Models (LLMs). BERT enhances the chatbot's ability to understand user queries by capturing context and intent, while RAG retrieves pertinent information from a dynamic database of educational programs, ensuring responses are both accurate and up-to-date. LLMs facilitate the generation of coherent, conversational replies, resulting in a user-friendly interface. By integrating these technologies, the chatbot aims to improve access to educational resources, promote informed decision-making, and ultimately enhance public service delivery in the education sector.

Introduction and Overview of the Project.

In an era where information is abundant yet often overwhelming, chatbots have emerged as effective tools for providing users with instant access to relevant data. Specifically, a chatbot focused on government education schemes can assist citizens in understanding various programs, eligibility criteria, application processes, and benefits. By utilizing advanced Natural Language Processing (NLP) techniques, such as BERT (Bidirectional Encoder Representations from Transformers), Large Language Models (LLM), and Retrieval-Augmented Generation (RAG), developers can create a sophisticated chatbot that delivers accurate and contextually relevant information.

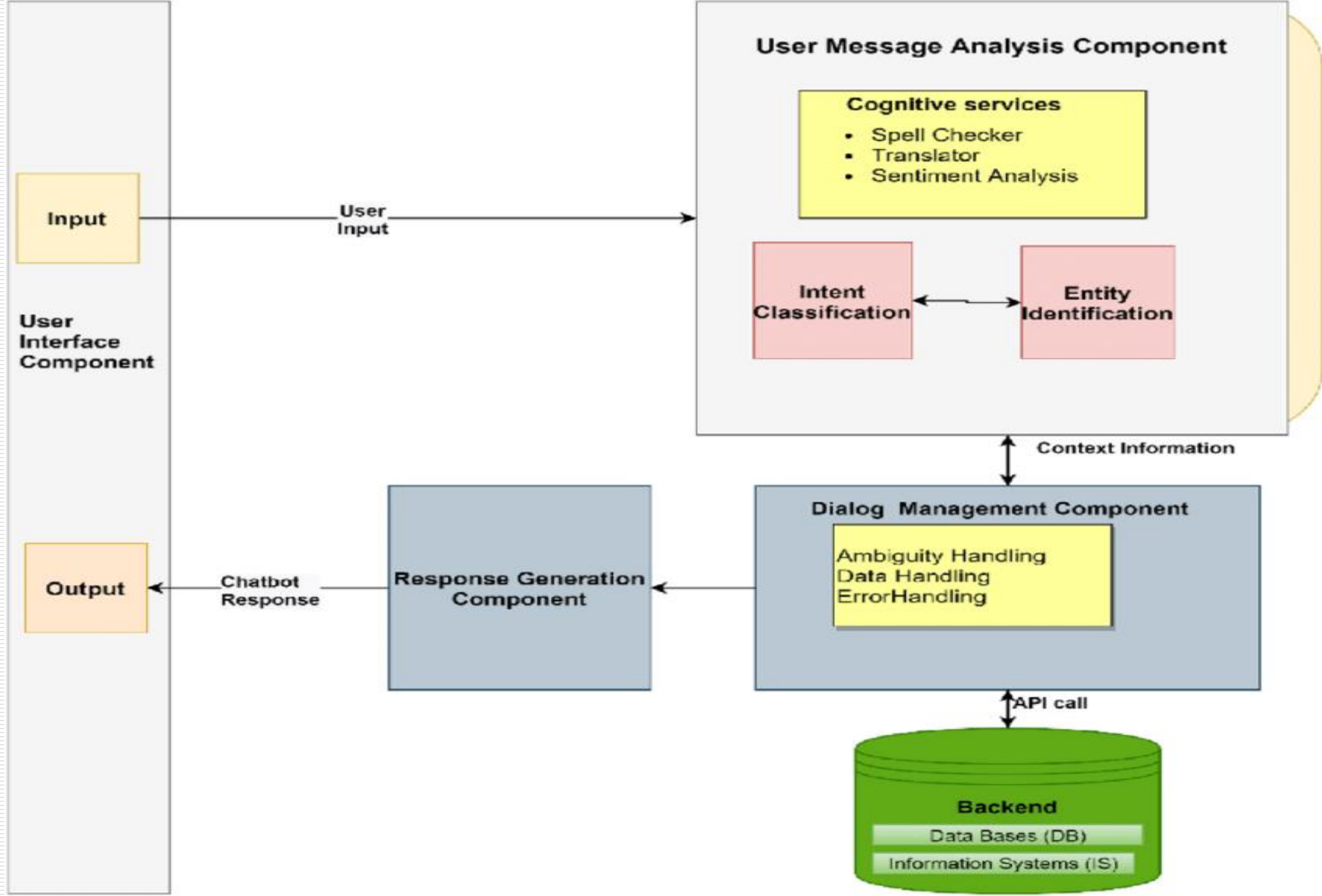
Literature Survey

S.No	Author Name	Paper Title	Description	Journal	Volume/ Year
01	Jin K. Kim	ChatGPT and large language model (LLM) chatbots	Performed a scoping review of available literature to understand the current state of LLM use in medicine and to provide a guideline for future utilization in academia.	IEEE	VOLUME 19
02	Samuel Kernan Freire	Conversational Assistants in Knowledge-Intensive Contexts: An Evaluation of BERT versus Intent-based Systems	Conversational Assistants (CA) are increasingly supporting human workers in knowledge management. Traditionally, CAs respond in specific ways to predefined user intents and conversation patterns.	IEEE	2023
03	Sumit Kumar Dam	A Complete Survey on BERT AI Chatbots	Conversational agents, often referred to as AI chatbots, rely heavily on such data to train large language models (LLMs) and generate new content (knowledge) in response to user prompts.	IEEE	2022

S.No	Author Name	Paper Title	Description	Journals	Volume/Year
04	Juan De Lara	Automating the Development of Task-oriented LLM-based Chatbots	Task-oriented chatbots are increasingly used to access all sorts of services – like booking a flight, or setting a medical appointment – through natural language conversation. There are many technologies for implementing task-oriented chatbots, including Dialogflow, Watson, and Rasa.	IEEE	2023
05	Babymol Kurian	GovInfohub: A Dynamic Government scheme Chatbot for informed Engagement and Accessibility	‘GovInfoHub’ presents a pioneering solution to provide citizens with real-time and accessible information regarding government schemes.	IEEE	2022

Proposed System

- ❑ **Advanced Natural Language Processing Multi-Language Support:** Implement support for multiple languages to make the chatbot accessible to a global audience.
- ❑ **Context Switching:** Allow users to change topics smoothly without losing track of the conversation flow.
- ❑ **Response Variation:** Implement multiple response options for the same query to make interactions feel more natural.
- ❑ **Feedback Collection:** Allow users to rate responses and provide feedback to improve the chatbot.



Algorithm

1. Collect, clean, and format data on government education schemes for training and testing.
2. Train or fine-tune a pre-trained BERT model to understand queries and provide relevant responses about schemes.
3. Set up a Django backend to handle API requests and responses from the BERT-based chatbot.
4. Develop the user-facing website using PHP for server-side logic and HTML/CSS for the interface.
5. Connect the trained BERT model via Django to handle chatbot queries and responses dynamically.
6. Embed the chatbot on the website, ensuring it communicates with the Django backend.
7. Perform unit, integration, and user testing to ensure proper chatbot and website functionality.
8. Deploy the website and backend on a web server, ensuring the chatbot is live for user interaction.
9. Continuously monitor, update the chatbot model, and maintain the website for new schemes or enhancements.

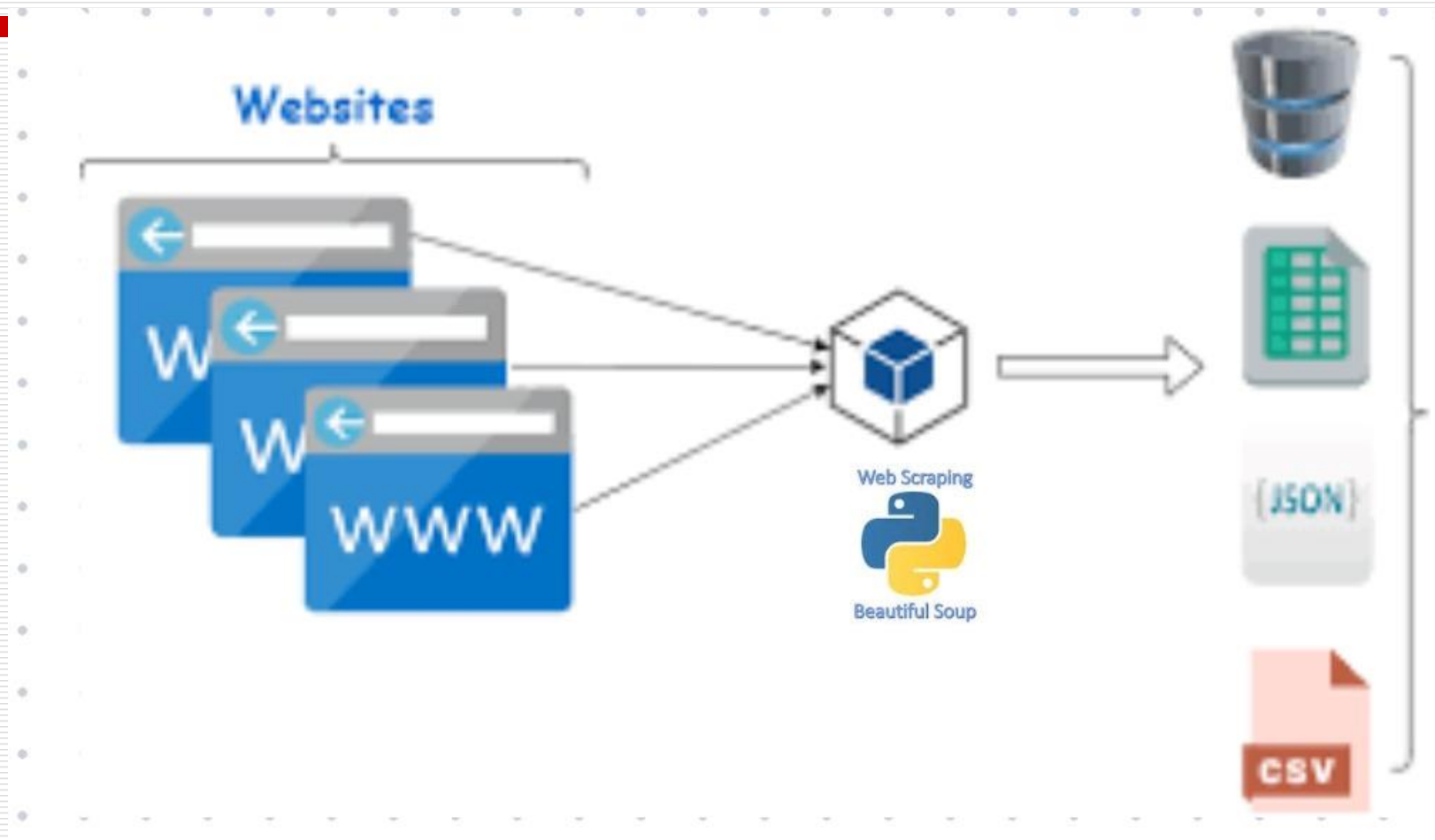
List of modules

1. Data collection
2. Data pre-processing
3. BRET model
4. Frontend Module
5. Integrating backend and frontend

Data collection

- ❑ 1. Install Required Libraries: Install ``beautifulsoup4``, ``requests``, and ``pandas`` using pip.
- ❑ 2. Identify Target Website: Select a website hosting the education schemes data.
- ❑ 3. Fetch the Web Page: Use the ``requests`` library to fetch the webpage content.
- ❑ 4. Parse the Web Page: Parse the fetched HTML content with BeautifulSoup.
- ❑ 5. Identify Data Elements: Inspect the HTML structure to locate relevant tags and classes.
- ❑ 6. Extract Relevant Data: Use BeautifulSoup methods to extract scheme details like name, eligibility, and benefits.
- ❑ 7. Clean and Preprocess Data: Clean the data by removing unnecessary characters or tags.

-
- ❑ 8. Convert Data to a Structured Format: Organize the data into a Pandas DataFrame.
 - ❑ 9. Save the Dataset: Save the structured data as a CSV or Excel file.
 - ❑ 10. Handle Pagination (if applicable): Automate scraping for multiple pages if the data spans several pages.



	A	B	C	D	E	F	G	H	I	J	K
1	Scheme Name	Launch Date	Objective	Eligibility	Criteria	Source					
2	Chief Minister's Breakfast Scheme	September 15, 2022	Provides free breakfast	Students of government schools	All enrolled students are eligible without	(TNPSC Current Affairs) https://www.tnpscshervupettagam.com/articles-detail/recent-tamil					
3	Pudhumai Penn Scheme	September 5, 2022	Supports female students	Girls studying in government schools from Class 6 to 12 who join any	Recent Government Schemes.						
4	Illam Thedi Kalvi	October 27, 2021	Focuses on after-school	Students from grades 1 to 8 in	Available to all students needing extra	Official Scheme Details.					
5	Makkalai Thedi Kalvi	N (TNPSC Current Affairs)	Door-to-door edu	Rural students who missed cla	Focus on children in rural areas with low	Government Scheme Overview.					
6	Chief Minister's Merit Scholarship	2021	Scholarships for top-per	Government school students	High performance in board exams.	(TNPSC Current Affairs) https://www.tnpscshervupettagam.com/articles-detail/recent-ta					
7	Post-Matric Scholarship for SC/ST/OBC	Ongoing	Provides financial assistance for SC, ST, and OBC students to pursue studies beyond Class								
8	Eligibility: Students belonging to S (TNPSC Current Affairs)	Criteria**	Must be enrolled in rec	Students belonging to S (TNPSC Current Affairs)	Criteria**	Must be	Social Welfare Programs.				
9	Mudhalvarin Mugavari Scheme	Feb-24	Integrated helpline for	Open to all residents of Tamil	No specific restrictions; available to any	Official Government Portal.					
10	(TNPSC Current Affairs)Launch Date*: 2011, ongoing		Provides free laptops t	Higher secondary students in	Students must be in Classes 11 and						
11	Source: Education Schemes.					Education Schemes.					
12	Free Bicycle Scheme	2001, ongoing	Provides free bicycles	Students in Class 9 and above	Must be from a rural area and in Class 9	(TNPSC Current Affairs) https://www.tnpscshervupettagam.com/articles-detail/recent-ta					
13	Girl Child Protection Scheme	2012	Financial assistance to	Families with one (TNPSC Current Affairs)	Criteria**	Family must n	Tamil Nadu Welfare Schemes.				
14	Educational Assistance for Children of Differ	1982, ongoing	Offers nutritious meals	All government school studen	Enrolled in government schools from Cla	[Social Welfare Programs] https://www.tnpscshervupettagam.com/articles-detail/recent-ta					
15	Source: Education Department Portal.					Education Department Portal.					
16	Vocational Education Scheme	2022	Integrates vocational t	Students in higher secondary s	Must be enrolled (TNPSC Current Affairs)	Vocational Education Initiatives.					
17	Educational Loan Scheme for Overseas Studi	2019	Provides financial support for students pursuing high	Must meet financial need and academic	Education Finance Programs.						
18	Thalikka (TNPSC Current Affairs)Schen	2010	Provides financial support for girls completing Class 10 or								
19	Eligibility: Girls from economically disadvantaged families.			Girls from economically disad	Must have completed secondary educat	[S (TNPSC Current Affairs) https://www.tnpscshervupettagam.com/articles-detail/re					
20	E.V.R. Maniammai Memorial Free Educat	2012	Offers free education	Students from backward class	Must belo (TNPSC Current Affairs)	Tamil Nadu Welfare Programs.					
21	National Means-cum-Merit Scholarship (NM	2008	Financial aid for disadv	Students in government schools from Classes 8 to							
22	Criteria: Based on merit and financial need.				Based on merit and financial need.	Education Programs.					
23	Free Uniform Scheme	2011	Provides free uniforms	Government school students from Class 1 to							
24	Criteria: All students in government schools qualify.				All students in government schools quali	Social Assistance Programs.					
25	Periyar EVR Nagammai Free Education Scher	2020	Free education for eco	Students from backward class	Must meet the economic need threshol	Government Schemes.					
26	Chief Minister's Uzhavar Scholarship	2022	Scholarships for childre	Farmers' children pursuing edu	Must be enrolled in educational instituti	Social Welfare Programs.					
27											

Data pre-processing

1. Lowercasing
2. Removing Punctuation
3. Tokenization
4. Removing Stop Words
5. Lemmatization / Stemming
6. Removing Numbers
7. Removing Extra Whitespaces
8. Handling Special Characters
9. Spelling Correction
10. Handling Negations


```

scheme_data.txt
File Edit View

### 1. **Chief Minister's Breakfast Scheme**
- **Launch Date**: September 15, 2022
- **Objective**: Provides free breakfast to government school students (classes 1 to 5) to improve concentration and attendance.
- **Eligibility**: Students of government schools in Tamil Nadu (up to Class 5).
- **Criteria**: All enrolled students are eligible without any income or caste barriers.
- **Source**: [Tamil Nadu Government Schemes](10).

### 2. **Pudhumai Penn Scheme**
- **Launch Date**: September 5, 2022
- **Objective**: Supports female students pursuing higher education by providing ₹1,000 per month.
- **Eligibility**: Girls studying in government schools from Class 6 to 12 who join any higher education course.
- **Criteria**: Must continue higher education in recognized institutions.
- **Source**: [Recent Government Schemes](10).

### 3. **Illam Thedi Kalvi**
- **Launch Date**: October 27, 2021
- **Objective**: Focuses on after-school learning and volunteer-based education at homes.
- **Eligibility**: Students from grades 1 to 8 in government schools.
- **Criteria**: Available to all students needing extra educational support.
- **Source**: [Official Scheme Details](10).

### 4. **Makkalai Thedi Kalvi**
- **Launch Date**: November 2021
- **Objective**: Door-to-door educational initiative ensuring education reaches remote and underprivileged children.
- **Eligibility**: Rural students who missed classes during the pandemic.
- **Criteria**: Focus on children in rural areas with low access to education.
- **Source**: [Government Scheme Overview](11).

### 5. **Chief Minister's Merit Scholarship**
- **Launch Date**: 2021
- **Objective**: Scholarships for top-performing students from economically weaker sections.
- **Eligibility**: Government school students from economically disadvantaged backgrounds.
- **Criteria**: High performance in board exams.
- **Source**: [Government Schemes](10).

### 6. **Post-Matric Scholarship for SC/ST/OBC**
- **Launch Date**: Ongoing
- **Objective**: Provides financial assistance for SC, ST, and OBC students to pursue studies beyond Class 10.
- **Eligibility**: Students belonging to SC, ST, OBC categories.
- **Criteria**: Must be enrolled in recognized institutions and meet income criteria.
```

BERT Model

1. **Tokenization:** The input question and context are broken into smaller units called tokens.
2. **Input Representation:** Each token is converted into a dense vector or embedding for numerical processing.
3. **Positional Encoding:** Positional embeddings are added to the tokens to help BERT understand the order of words.
4. **BERT Architecture:** The model consists of multiple Transformer layers that process tokens and build context-aware representations.
5. **Self-Attention Mechanism:** Each token pays attention to other tokens in the sequence to capture word relationships in both directions.
6. **Contextual Encoding:** BERT layers refine token meanings based on the context, enhancing understanding as tokens pass through the model.
7. **Question Understanding:** The [CLS] token at the start of the sequence helps BERT comprehend the question.

BERT Model

8. **Start and End Token Prediction:** BERT predicts the tokens marking the beginning and end of the answer span.
9. **Answer Extraction:** The model selects the most likely answer from the context based on start and end token predictions.
10. **Final Output:** BERT returns the extracted answer from the context as the response to the question.

Tokenization:

1. Basic Tokenization

Split the text into words or subwords using BERT's tokenizer, which supports **WordPiece** or **Subword Tokenization**.

Formula:

$$\mathbf{Ttokens} = \mathbf{BERT_Tokenizer(T)}$$

2. Add Special Tokens

Add special tokens to mark the beginning and end of the input sequence:

[CLS]: Indicates the start of a sequence (used for classification or context embedding).

[SEP]: Separates different sentences or marks the end of a single sentence.

Formula:

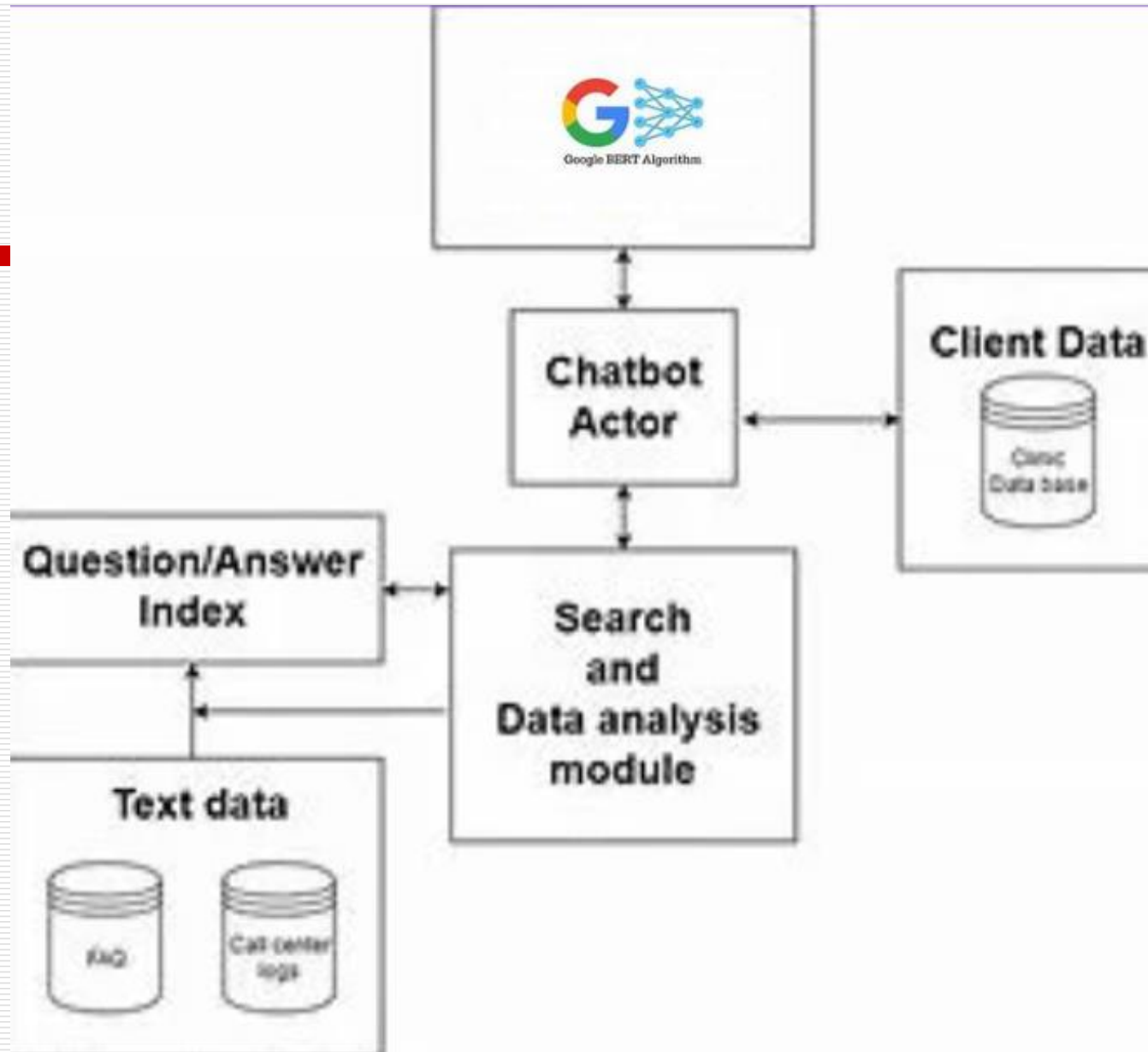
$$\mathbf{Ttokens} = [[\mathbf{CLS}]] + \mathbf{Ttokens} + [[\mathbf{SEP}]]$$


3. Convert Tokens to IDs

Convert each token into its corresponding ID from the BERT vocabulary.

Formula:

$$\mathbf{Tids} = \text{Token_to_ID}(\mathbf{Ttokens})$$



 QndA_Bot.ipynb ☆


File Edit View Insert Runtime Tools Help Last saved at 8:02 PM

Comment Share Settings Profile

+ Code + Text Connect T4 Gemini

```
[ ] from transformers import AutoTokenizer,AutoModelForQuestionAnswering,pipeline

[ ] model=AutoModelForQuestionAnswering.from_pretrained("deepset/roberta-base-squad2")
    tokenizer=AutoTokenizer.from_pretrained("deepset/roberta-base-squad2")
```

 /usr/local/lib/python3.10/dist-packages/huggingface_hub/utils/_token.py:89: UserWarning:
The secret 'HF_TOKEN' does not exist in your Colab secrets.
To authenticate with the Hugging Face Hub, create a token in your settings tab (<https://huggingface.co/settings/tokens>), set it as secret in your Google Colab and restart your session
You will be able to reuse this secret in all of your notebooks.
Please note that authentication is recommended but still optional to access public models or datasets.

warnings.warn(
config.json: 100% ██████████ 571/571 [00:00<00:00, 40.7kB/s]
model.safetensors: 100% ██████████ 496M/496M [00:05<00:00, 218MB/s]
tokenizer_config.json: 100% ██████████ 79.0/79.0 [00:00<00:00, 2.90kB/s]
vocab.json: 100% ██████████ 899k/899k [00:00<00:00, 4.65MB/s]
merges.txt: 100% ██████████ 456k/456k [00:00<00:00, 2.41MB/s]
special_tokens_map.json: 100% ██████████ 772/772 [00:00<00:00, 64.3kB/s]
/usr/local/lib/python3.10/dist-packages/transformers/tokenization_utils_base.py:1601: FutureWarning: `clean_up_tokenization_spaces` was not set. It will be set to `True` by default. T
warnings.warn(
context=''schemes and expansion:

CO

QndA_Bot.ipynb

☆

File Edit View Insert Runtime Tools Help Saving...

Comment

Share

⚙️

+ Code + Text

Reconnect T4 Gemini

⬆️ ⬇️ 🔗 🗨️ ⚙️ 📄 🗑️ ⋮

▶️

context=''schemes and expansion:

1. ****Chief Minister's Breakfast Scheme****

- ****Launch Date****: September 15, 2022
- ****Objective****: Provides free breakfast to government school students (classes 1 to 5) to improve concentration and attendance.
- ****Eligibility****: Students of government schools in Tamil Nadu (up to Class 5).
- ****Criteria****: All enrolled students are eligible without any income or caste barriers.
- ****Source****: [Tamil Nadu Government Schemes](10).

2. ****Pudhumai Penn Scheme****

- ****Launch Date****: September 5, 2022
- ****Objective****: Supports female students pursuing higher education by providing ₹1,000 per month.
- ****Eligibility****: Girls studying in government schools from Class 6 to 12 who join any higher education course.
- ****Criteria****: Must continue higher education in recognized institutions.
- ****Source****: [Recent Government Schemes](10).

3. ****Illam Thedi Kalvi****

- ****Launch Date****: October 27, 2021
- ****Objective****: Focuses on after-school learning and volunteer-based education at homes.
- ****Eligibility****: Students from grades 1 to 8 in government schools.
- ****Criteria****: Available to all students needing extra educational support.
- ****Source****: [Official Scheme Details](10).

4. ****Makkalai Thedi Kalvi****

- ****Launch Date****: November 2021
- ****Objective****: Door-to-door educational initiative ensuring education reaches remote and underprivileged children.
- ****Eligibility****: Rural students who missed classes during the pandemic.
- ****Criteria****: Focus on children in rural areas with low access to education.
- ****Source****: [Government Scheme Overview](11).

✓ 0s completed at 10:22 PM

18. Free Uniform Scheme

Launch Date: 2011

Objective: Provides free uniforms and footwear to reduce economic barriers for students.

Eligibility: Government school students from Class 1 to 8.

Criteria: All students in government schools qualify.

Source: Social Assistance Programs.

```
[13] ask=pipeline('question-answering',model=model,tokenizer=tokenizer)
      result=ask(question="which scheme provide uniform and footwear?",context=context)
```

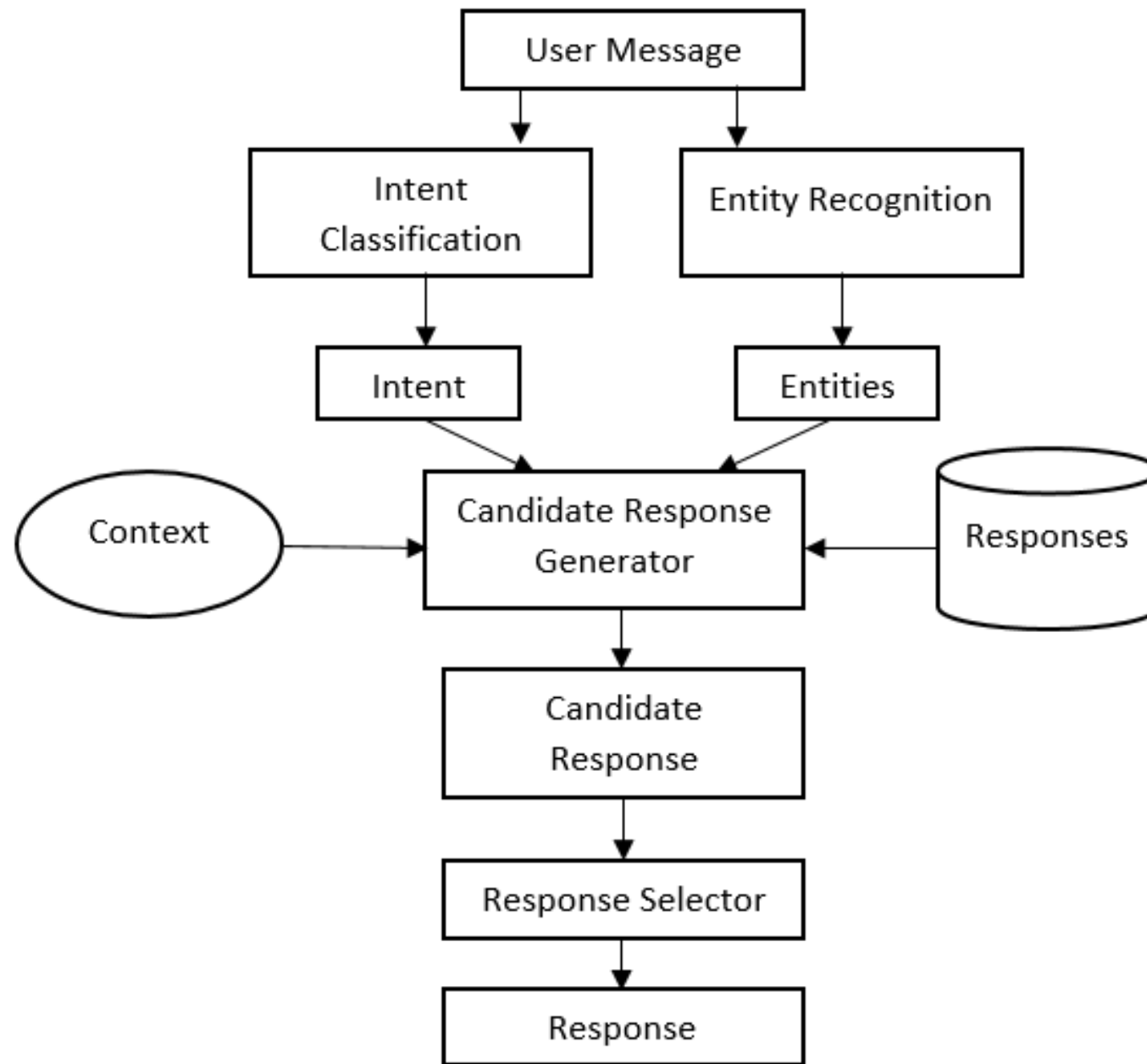
⚡ Hardware accelerator e.g. GPU is available in the environment, but no `device` argument is passed to the `Pipeline` object. Model will be on CPU.

```
[14]
      result['answer']
```

⚡ 'Free Uniform Scheme'

Frontend Design

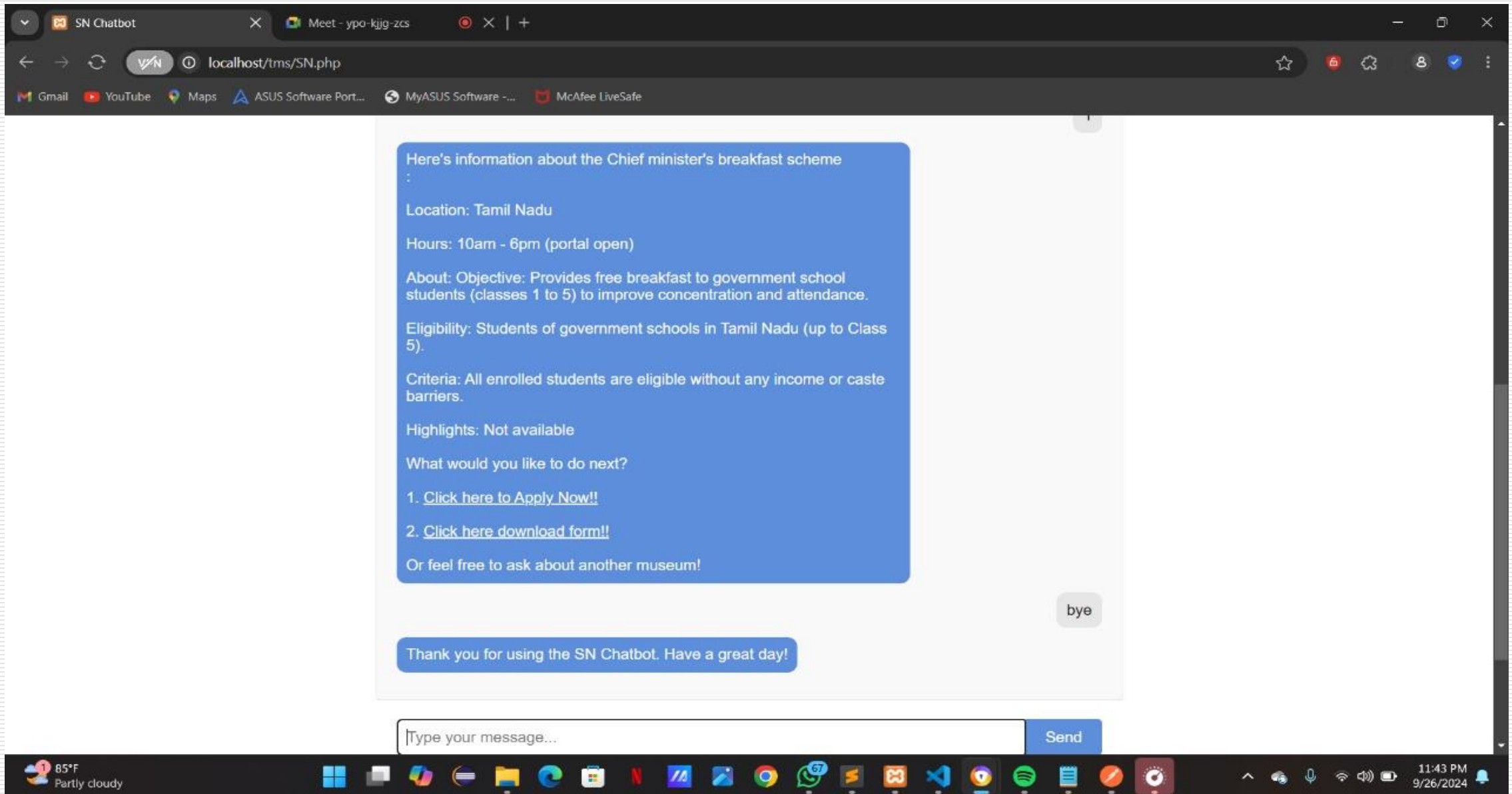
1. Plan Your Chatbot Design: Define chatbot features, layout, and functionality.
2. Set Up the Project Directory: Create a structured directory for frontend and backend files.
3. Create the Frontend (HTML): Design the chatbot interface using HTML.
4. Style the Frontend (CSS): Use CSS to style the chatbot layout and make it visually appealing.
5. Add Client-Side Logic (JavaScript): Write scripts to handle user input and display bot responses.
6. Set Up the Node.js Server: Build a backend server using Node.js and Express.
7. Install Dependencies: Install required packages like ``express`` and ``body-parser``.
8. Test the Website Locally: Run the server and verify the chatbot functionality in the browser.
9. Improve Chatbot Logic: Integrate AI models or APIs to enhance chatbot responses.
10. Deploy the Website: Host the website on platforms like Heroku, Vercel, or AWS.



The screenshot shows a web browser window with two tabs: 'SN Chatbot' and 'Meet - ypo-kjjg-zcs'. The address bar displays 'localhost/tms/SN.php'. The browser's taskbar at the bottom includes icons for Gmail, YouTube, Maps, ASUS Software Port..., MyASUS Software, and McAfee LiveSafe.

The chatbot interface has a blue header with the title 'SN Chatbot', a 'Refresh' button, and a close icon. A circular icon with a document symbol is on the left. The chat area shows the following sequence of messages:

- Bot: Welcome to the SN Chatbot!!
- User: hi
- Bot: Hello! Welcome to the SN Chatbot
- Bot: Fetching scheme data...
- Bot: Here's a list of government schemes I can tell you about:
 - 1. Chief minister's breakfast scheme
 - 2. Pudhumai penn scheme
 - 3. Chief minister's merit scholarship
- Bot: Which scheme would you like to know more about? You can ask by name or number.
- Bot: Additionally, you can view more on our website [Click here download form!!](#)
- User: 1
- Bot: Here's information about the Chief minister's breakfast scheme



Integrating backend and frontend

- 1.Install Django:** Set up Django by installing it via pip (pip install django) and create a new project using django-admin startproject.
- 2.Create an App:** Inside the project, create an app using python manage.py startapp [app_name] where the backend logic will reside.
- 3.Define Models:** In the app's models.py, define the data models that represent the database schema.
- 4.Migrate Database:** Run python manage.py makemigrations and python manage.py migrate to create the necessary database tables based on models.
- 5.Create Views:** In views.py, define the logic that handles requests and processes backend operations like handling forms, interacting with models, and returning responses.
- 6.Set up URLs:** Map views to URLs by configuring urls.py to route frontend requests to the correct backend views.
- 7.Create HTML Templates:** Design frontend HTML templates that will render the data returned from Django views using Django's templating engine.

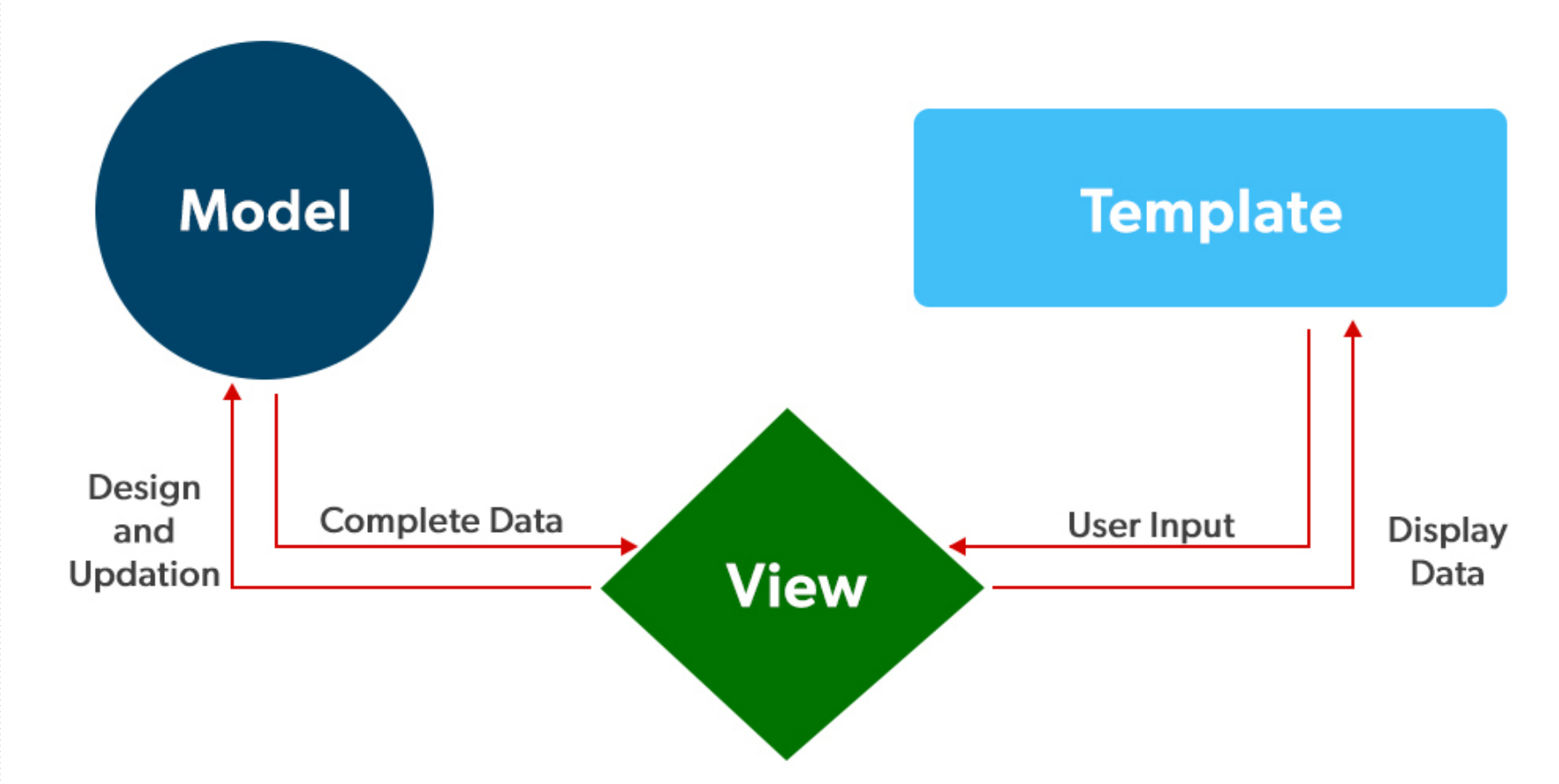
Integrating backend and frontend

8.Connect Views to Templates: In views, return HttpResponse or render() functions that combine the HTML templates with backend data and pass it to the user.

9.Handle Static Files: Serve CSS, JavaScript, and images by configuring STATIC_URL in settings.py and creating a static directory for frontend assets.

10.Run Server: Start the Django development server with python manage.py runserver to connect the backend and frontend, allowing user interaction via the web browser.

DFD



Comparisons & Analysis

S.No	Title Of The Paper	Methodology Used	Limitation(s)	Advantage(s)	Accuracy (In %)
1	A Conversational Agent for Learning: Development of an Educational Chatbot Using BERT	BERT, Intent, Entity Recognition, Response Generation.	Limited dataset size may affect the model's generalization. The complexity of user queries is not fully handled by the current model.	Improved understanding of user intent due to BERT's contextual embeddings. Enhanced user engagement through interactive conversations.	80
2	Enhancing Student Learning through AI: An Intelligent Chatbot Using LLMs	LLM	Dependency on training data; biases in the dataset can reflect in responses.	High-quality response generation due to LLM capabilities. Flexibility in handling a wide range of educational topics.	85
3	Integrating RAG for Enhanced Educational Chatbots: A Case Study	<u>RAG</u> (Retrieval augmented generation)	The performance may vary based on the quality of retrieved documents. Increased complexity in implementation and maintenance.	Combines real-time data retrieval with generative responses, improving accuracy. Can handle more diverse queries with factual support.	90
4	Building a Context-Aware Educational Chatbot Using BERT and LLM	<u>LLM</u> (Large language model), BERT(Bidirectional Encoder Representations from Transformers)	High computational requirements for real-time processing. Limited understanding of domain-specific jargon.	Context-aware conversations leading to improved user satisfaction. Ability to adapt responses based on user interactions.	78
5	High Accuracy Phishing Detection Based on Convolutional Neural Networks	Convolutional Neural Network, Random Forest, Naive Bayes, SVM, J48.	As more features are extracted in this system, the model becomes more complex.	The model achieved high accuracy and had a faster implementation.	97.3

References.

- ❑ **"LoRA: Low-Rank Adaptation of Large Language Models"** Jacob Devlin, Ming-Wei Chang, IEEE Human Language Technologies, vol 11, Date: June 2022, Pages:133-144.
- ❑ **"Towards Intelligent Public Services: An AI-based Chatbot for the Swedish Social Insurance Agency"**, Anders Henriksson, IEEE Access, vol: 7, Date: 2023, Pages:56-65.
- ❑ **"LLM for SoC Security: A Paradigm Shift"**, Gianluca Klopfenstein, IEEE Transactions on data,vol:10, Pages: 32-43.
- ❑ **"LLM for SoC Security: A Paradigm Shift"**, Yingqiang Ge, Wenyue Hua, IEEE Access, Date: 2023, Pages:11-20

Conclusion

In conclusion, developing a chatbot for education schemes using BERT, Retrieval-Augmented Generation (RAG), and Large Language Models (LLMs) offers a powerful framework for providing accurate, contextually relevant information to users. By leveraging BERT's capabilities for understanding user queries and generating embeddings, combined with RAG's ability to retrieve pertinent documents and LLMs' proficiency in generating coherent responses, the chatbot can effectively address diverse inquiries related to educational schemes. This integrated approach not only enhances user experience through natural and engaging interactions but also ensures that the information provided is up-to-date and precise, ultimately empowering citizens with valuable knowledge about government education initiatives.



Thank You