



Mini Project Report On

AcroMemory

*Submitted in partial fulfillment of the requirements for the
award of the degree of*

Bachelor of Technology

in

Computer Science & Engineering

By

Issac Mathew Jaimon (U2103104)

Jerin Varghese (U2103110)

Jerin Varghese Tom (U2103111)

Madhav M Menon (U2103129)

Under the guidance of

Ms.Anu Maria Joykutty

**Department of Computer Science & Engineering
Rajagiri School of Engineering & Technology (Autonomous)
(Affiliated to APJ Abdul Kalam Technological University)**

Rajagiri Valley, Kakkanad, Kochi, 682039

May 2024

CERTIFICATE

*This is to certify that the mini project report entitled "**AcroMemory**" is a bonafide record of the work done by **Issac Mathew Jaimon (U2103104)**, **Jerin Varghese (U2103110)**, **Jerin Varghese Tom (U2103111)**, **Madhav M Menon (U2103129)**, submitted to the APJ Abdul Kalam Technological University in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology (B. Tech.) in Computer Science and Engineering during the academic year 2023-2024.*

Ms.Anu Maria Joykutty
Assistant Professor
Dept. of CSE
RSET

Dr.Saritha S
Professor
Dept. of CSE
RSET

Dr. Preetha KG
Head of the Department
Dept. of CSE
RSET

ACKNOWLEDGEMENTS

We wish to express my sincere gratitude towards **Dr P. S. Sreejith**, Principal of RSET, and **Dr. Preetha K.G.**, Head of the Department of Computer Science and Engineering for providing me with the opportunity to undertake my mini project, "AcroMemory".

I am highly indebted to my project coordinators, **Dr.Saritha S**, Designation, Department of Computer Science and Engineering for their valuable support.

It is indeed my pleasure and a moment of satisfaction for me to express my sincere gratitude to my project guide **Ms.Anu Maria Joykutty** for her patience and all the priceless advice and wisdom he/she has shared with me.

Last but not the least, I would like to express my sincere gratitude towards all other teachers and friends for their continuous support and constructive ideas.

Issac Mathew Jaimon

Jerin Varghese

Jerin Varghese Tom

Madhav M Menon

Abstract

AcroMemory is an innovative software used for content summarization through automated acronym generation, employing Natural Language Processing (NLP) algorithms to distill key concepts from sentences or points. Its two-step process involves advanced comprehension techniques followed by keyword extraction, ensuring the generated acronyms accurately encapsulate the essence of the input information. What sets AcroMemory apart is its integration of semantic understanding, surpassing conventional keyword matching by capturing contextual relationships between words, thus enhancing clarity and memorability. The software's user customization feature enables tailoring of the acronym generation process, allowing individuals to prioritize specific keyword types for personalized summarization experiences. With applications spanning education, business, research, and beyond, AcroMemory enhances productivity and comprehension by offering a streamlined approach to information retention and communication. Its ability to condense complex topics into easily digestible acronyms aids students in learning, facilitates streamlined communication and decision-making in business environments, and supports researchers in data analysis and dissemination. Ultimately, AcroMemory contributes to increased efficiency and understanding, empowering users to effortlessly create memorable summaries that drive productivity and knowledge dissemination across diverse domains.

Contents

Acknowledgements	i
Abstract	ii
List of Figures	v
1 Introduction	1
1.1 Background	1
1.2 Problem Definition	2
1.3 Scope and Motivation	2
1.4 Objectives	3
1.5 Challenges	3
1.6 Assumptions	4
1.7 Societal / Industrial Relevance	4
1.8 Organization of the Report	5
2 Software Requirements Specification	6
2.1 Introduction	6
2.2 Overall Description	6
2.3 External Interface Requirements	8
2.4 System Features	11
2.5 Other Nonfunctional Requirements	15
3 System Architecture and Design	18
3.1 System Overview	18
3.2 Architectural Design	19
3.3 Dataset identified	19
3.4 Proposed Methodology/Algorithms	19
3.5 User Interface Design	21

3.6	Database Design	22
3.7	Description of Implementation Strategies	22
3.8	Module Division	23
3.9	Work Schedule - Gantt Chart	24
4	Results and Discussions	25
4.1	Overview	25
4.2	Testing	25
4.3	Quantitative Results	26
4.4	Graphical Analysis	28
4.5	Discussion	28
5	Conclusion	29
5.1	Conclusion	29
5.2	Future Scope	29
	References	30
	Appendix A: Presentation	31
	Appendix B: Vision, Mission, Programme Outcomes and Course Outcomes	43
	Vision, Mission, POs, PSOs and COs	ii
	Appendix C: CO-PO-PSO Mapping	v

List of Figures

3.1	Architectural Model	18
3.2	Architectural Diagram	19
3.3	User Interface	22
3.4	Gantt Chart	24
4.1	Interface to input text	25
4.2	Output generation	26
4.3	Text document	26
4.4	Accuracy of Random Cases	27
4.5	Graph Showing Accuracy of Model	28

Chapter 1

Introduction

1.1 Background

AcroMemory is an innovative software designed for content summarization through automated acronym generation. Leveraging Natural Language Processing (NLP) algorithms, it efficiently distills key concepts from sentences or points, enhancing comprehension and retention. Its two-step process involves advanced comprehension techniques followed by keyword extraction, ensuring the generated acronyms accurately encapsulate the essence of the input information.

Current Scenarios: In today's information-rich world, individuals and organizations are inundated with vast amounts of data and content. However, the ability to quickly comprehend and retain essential information is paramount for productivity and decision-making. Conventional methods of summarization often fall short in capturing the essence of complex topics concisely.

Importance: AcroMemory addresses this challenge by offering a sophisticated solution that goes beyond simple keyword matching. By integrating semantic understanding, it captures contextual relationships between words, enhancing clarity and memorability. This feature sets it apart from conventional summarization tools.

Moreover, the software's user customization feature allows individuals to tailor the acronym generation process according to their specific needs and preferences. This personalization enhances the summarization experience, enabling users to prioritize specific keyword types for more relevant and effective summaries.

Applications: AcroMemory finds applications across various domains including education, business, research, and beyond. In education, it aids students in learning by condensing complex topics into easily digestible acronyms, facilitating comprehension and retention. In business environments, it supports streamlined communication and

decision-making by providing concise summaries of key information. Additionally, in research, it assists researchers in data analysis and dissemination, enhancing efficiency and understanding.

1.2 Problem Definition

The challenge lies in efficiently summarizing complex information to enhance comprehension and retention. The aim of the project is to develop a software solution, AcroMemory, that employs automated acronym generation and NLP algorithms to distill key concepts from text, ultimately improving productivity and knowledge dissemination across diverse domains.

1.3 Scope and Motivation

The scope of the project encompasses the development and implementation of AcroMemory, an innovative software tool for content summarization through automated acronym generation. This tool will be designed to process text input and employ advanced Natural Language Processing (NLP) algorithms to distill key concepts into concise acronyms. The scope also includes the integration of semantic understanding to capture contextual relationships between words, thereby enhancing clarity and memorability. AcroMemory will offer a user customization feature, allowing individuals to tailor the summarization process to prioritize specific keyword types for personalized experiences. The software's applications span across education, business, research, and beyond, aiming to enhance productivity and comprehension by offering a streamlined approach to information retention and communication.

The motivation behind the development of AcroMemory stems from the increasing need for efficient information summarization in today's fast-paced world. With the exponential growth of data and content, individuals and organizations face challenges in quickly comprehending and retaining essential information. AcroMemory seeks to address this challenge by providing a solution that goes beyond traditional summarization methods, employing advanced NLP techniques to generate concise and memorable summaries. By condensing complex topics into easily digestible acronyms, AcroMemory aims to facilitate learning for students, streamline communication and decision-making in business

environments, and support researchers in data analysis and dissemination. Ultimately, the motivation behind AcroMemory is to empower users with the ability to effortlessly create memorable summaries that drive productivity and knowledge dissemination across diverse domains.

1.4 Objectives

- Develop and refine the automated acronym generation algorithm within AcroMemory to accurately distill key concepts from text inputs, ensuring the generated acronyms effectively encapsulate the essence of the information.
- Implement advanced Natural Language Processing (NLP) techniques to enhance the semantic understanding capabilities of AcroMemory, allowing it to capture contextual relationships between words and thereby improve the clarity and memorability of generated acronyms.
- Design and integrate a user customization feature into AcroMemory, enabling individuals to tailor the acronym generation process to prioritize specific keyword types for personalized summarization experiences, thereby enhancing relevance and effectiveness.
- Conduct extensive testing and validation of AcroMemory across diverse domains, including education, business, and research, to ensure its functionality and effectiveness in enhancing productivity, comprehension, and knowledge dissemination.
- Provide comprehensive documentation and user support resources for AcroMemory, including tutorials, user guides, and troubleshooting materials, to facilitate seamless adoption and usage by individuals and organizations seeking to improve information retention and communication efficiency.

1.5 Challenges

The project faces challenges related to the complexity of Natural Language Processing (NLP) algorithms required for accurate acronym generation and semantic understanding.

Additionally, ensuring the scalability and robustness of AcroMemory across diverse text inputs and domains presents a significant challenge. Furthermore, balancing the trade-offs between comprehensiveness and conciseness in summarizing complex information poses an ongoing challenge throughout the development and refinement process. The project also encounters challenges due to the lack of powerful hardware resources, such as GPUs, which are essential for accelerating the processing of large datasets and optimizing algorithm performance. Integrating AcroMemory's advanced functionalities while operating within hardware constraints presents an additional hurdle in achieving optimal performance and efficiency.

1.6 Assumptions

- The project assumes the availability of a sufficient volume of textual data for training and testing the NLP algorithms within AcroMemory.
- It is assumed that users have a basic understanding of the concepts and terminology relevant to the text inputs processed by AcroMemory, ensuring the generated acronyms are contextually relevant and comprehensible.
- The project assumes a reasonable level of computational resources for running AcroMemory, although it acknowledges potential constraints such as limited processing power or memory on user devices.

1.7 Societal / Industrial Relevance

The project's applications span various sectors, offering relevance and benefits both to society and industry:

Education: AcroMemory can revolutionize learning by providing students with concise summaries of complex topics, aiding comprehension and retention. It facilitates efficient studying, potentially improving academic performance across diverse subjects and educational levels.

Business: In corporate settings, AcroMemory streamlines communication and decision-making processes by condensing essential information into easily digestible acronyms. It

enhances productivity by enabling quick understanding of key concepts, supporting more efficient workflow and collaboration.

Research: Researchers can leverage AcroMemory to analyze and disseminate vast amounts of data more effectively. By summarizing research findings and complex datasets into memorable acronyms, it accelerates knowledge dissemination, fostering advancements in various fields.

Healthcare: In the healthcare sector, AcroMemory can aid medical professionals in quickly comprehending complex medical literature, protocols, and patient records. It enhances communication and decision-making, potentially leading to improved patient care and outcomes.

1.8 Organization of the Report

The organization of the report is as follows:

Chapter 1- Introduction: This chapter introduces the project by providing the background and societal/industrial relevance of the project, the aim and objectives, and the challenges and assumptions faced by the project.

Chapter 2 - Software Requirements Specification: This chapter provides an overall description of the project, such as the product perspective, functions, operating environment, design and implementation constraints faced by the project, and assumptions and dependencies. It also describes the user interfaces, software interfaces, and communication interfaces for the project. The system features and the nonfunctional requirements are also discussed in this section.

Chapter 3 - System Architecture and Design: This chapter provides an overview of the project with a detailed architecture diagram, a sequence diagram, the user interface designs, and a schema of the database used. It describes the properties of the dataset, algorithms, and implementation strategies used in the project. This section also outlines the different modules involved in the project and the work schedule

Chapter 2

Software Requirements Specification

Insert your SRS document here.

2.1 Introduction

Purpose

The purpose of the AcroMemory Software is to ease learning and to create a convenient and easy-to-use application for students to create various memorable and meaningful acronyms for terms and phrases. The system is based on a relational database with its word library and generative functions. By automating the process of acronym generation, this tool aims to assist users in generating accurate and relevant acronyms that capture the essence of the original expression.

Product Scope

The product aims to provide users with a quick and efficient solution for creating catchy and relevant acronyms, enhancing communication and branding across various domains. Target Audience involves writers, content creators, and marketers looking to enhance branding and communication. Professionals in diverse industries seeking creative acronyms for projects and presentations. Students and educators for educational and mnemonic purposes.

2.2 Overall Description

Product Perspective

The Acronym Generator serves as a follow-on member of a product family that relates to generative algorithms designed to meet the growing demand for efficient and creative acronym generation across various domains. This software is not a replacement for existing systems but rather a specialized tool that complements communication strategies in

diverse applications. It can be utilized as a component within content creation platforms, naming conventions, or branding strategies. It can be utilized as a component within content creation platforms, naming conventions, or branding strategies.

Product Functions

1. Acronym Creation
2. Variety of Styles
3. Dictionary Integration
4. Language Support
5. Randomization Feature
6. User-defined Rules
7. Save and Share Functionality
8. Offline Mode
9. Feedback Mechanism

Operating Environment

Database: MySQL

Connectivity: JDBC (Java Database Connectivity)

Programming Language: Python

Operating System: Windows

User Interface Technology: HTML5, CSS3

Design and Implementation Constraints

1. Corporate Policies and Regulatory Compliance: Adherence to corporate policies regarding data privacy, intellectual property rights, and regulatory requirements governing the use of certain words or acronyms.
2. Hardware Limitations: Consideration of timing and memory requirements to ensure optimal performance, especially if the acronym generator is intended for deployment on resource-constrained devices.
3. Language Requirements: Compliance with language specifications and localization requirements if the acronym generator is intended for use in multilingual environments.
4. Interfaces to Other Applications: Compatibility with existing systems and applications, ensuring smooth integration without causing conflicts or disruptions.

Assumptions and Dependencies

Assumptions for Acronym Generator:

- 1.Third-Party API: Assuming continuous reliability of the third-party acronym database API.
- 2.Data Quality: Assuming the initial acronym dataset meets project requirements for accuracy and completeness.
- 3.Development Environment: Assuming a stable and consistent development environment to prevent compatibility issues.
- 4.User Input: Assuming consistent user input formats to facilitate accurate acronym generation.

Dependencies for AcroMemory:

- 1.Database Integration: Dependency on seamless integration with an external database for acronym storage and retrieval.
- 2.External Service Uptime: Dependency on reliable external web services for real-time data, crucial for proper acronym generation.
- 3.Open Source Libraries: Dependency on specific open-source libraries for NLP and string manipulation, subject to changes or discontinuation.

2.3 External Interface Requirements

User Interfaces

The user interface of AcroMemory would likely consist of several key components to facilitate efficient interaction with the software:

1. Input Area: This component allows users to enter the sentences or key points from which they want to generate acronyms. It may support various input formats such as text boxes, file upload options, or copy-paste functionality.
2. Acronym Generation Button: A prominent button or action element that triggers the acronym generation process once the input is provided. Clicking this button initiates the NLP algorithms to analyze the input and generate relevant acronyms.
3. Output Display: The area where the generated acronyms are displayed to the user. This could be a list or a grid format, presenting each acronym along with its corresponding expanded meaning.
4. Semantic Understanding Feedback: This component provides feedback or suggestions

related to the semantic understanding of the input. It may highlight ambiguous terms or offer options to refine the input for better results.

5. Customization Options: User-friendly controls or settings that allow users to customize the acronym generation process according to their preferences. This may include options to prioritize certain types of keywords, adjust the level of summarization, or specify formatting preferences.

Hardware Interfaces

The hardware interfaces required for AcroMemory would depend on the devices through which users access the software. Here are the potential hardware interfaces:

1. Input Devices: Users would need input devices to interact with AcroMemory, including:
 - Keyboards: For typing in sentences or key points.
 - Mice or Trackpads: For navigating the user interface and clicking on elements.
2. Display Devices: Users require display devices to view the AcroMemory user interface and generated acronyms, such as Monitors for desktop and laptop computers, displaying the software interface and results.
3. Network Connectivity: AcroMemory may require network connectivity to access cloud-based features, download updates, or communicate with remote servers for advanced processing tasks.

Software Interfaces

The software interfaces for AcroMemory encompass the interaction points between different software components within the system. Here are the key software interfaces:

1. User Interface (UI): Graphical User Interface (GUI): The GUI interface allows users to interact with AcroMemory through visual elements such as buttons, input fields, and displays. Users input sentences or key points, customize settings, and view generated acronyms. Command-Line Interface (CLI): For users who prefer text-based interaction or for automation purposes, AcroMemory may offer a CLI interface. This interface allows users to interact with the software through command-line commands, providing similar functionality to the GUI but without graphical elements.
2. Backend : The backend logic of AcroMemory includes the core functionalities respon-

sible for processing user input, generating acronyms, and managing data. This logic may be implemented using Python programming language along with libraries such as TensorFlow for natural language processing (NLP) tasks.

3. Database Interface: AcroMemory interacts with the MySQL database through a database interface. This interface facilitates communication between the software and the database for storing and retrieving user preferences, input data, generated acronyms, and other relevant information.

4. API (Application Programming Interface): AcroMemory may offer an API that allows other software systems or developers to integrate its functionality into their applications. This API provides endpoints for submitting input text, retrieving generated acronyms, and customizing settings programmatically.

5. External Services or APIs: AcroMemory may interact with external services or APIs for additional functionalities such as cloud storage, semantic analysis, or integration with other systems. This interaction is facilitated through software interfaces provided by the external services, allowing AcroMemory to access and utilize their functionalities.

6. Web Development Framework: If AcroMemory offers a web-based interface, a web development framework such as Flask. These frameworks facilitate the development of web applications and provide tools for handling HTTP requests, routing, and rendering HTML and CSS templates.

Communications Interfaces

1. Database Communication: Database Connectivity: AcroMemory communicates with the MySQL database using database connectivity interfaces such as MySQL Connector/Python. This allows AcroMemory to execute database queries, fetch or insert data, and manage database transactions.

2. External Service Communication: HTTP(S) Requests: AcroMemory may communicate with external services or APIs using HTTP(S) requests. This enables AcroMemory to send requests to external endpoints, receive responses, and exchange data with external systems over the internet.

3. Operating System Communication: System Calls: AcroMemory interacts with the underlying operating system through system calls, which are requests made by the software

to the operating system kernel for various operating system services. This includes services such as file I/O operations, network communication, process management, and system resource allocation. Platform-Specific APIs: AcroMemory may utilize platform-specific APIs provided by the operating system for tasks such as accessing hardware devices, managing system processes, or performing system-level configurations. These APIs facilitate communication with the underlying operating system and enable AcroMemory to leverage platform-specific functionalities.

2.4 System Features

Automated Acronym Generation

Description and Priority:

AcroMemory employs advanced Natural Language Processing (NLP) algorithms to automatically generate concise acronyms from input content, facilitating efficient content summarization. This feature streamlines the process of condensing complex information into easily digestible formats, aiding in comprehension and information retention.

Benefit: 9 (Facilitates efficient content summarization)

Penalty: 2 (Potential for inaccuracies in acronym generation)

Cost: 4 (Development and maintenance of NLP algorithms)

Risk: 3 (Dependence on the accuracy of NLP algorithms)

Stimulus/Response Sequences:

- i. User Action: User inputs a piece of content into AcroMemory.
- ii. System Response: AcroMemory analyzes the content using NLP algorithms and generates an acronym summarizing the key concepts.

Functional Requirements:

REQ-1: The system shall accept input text from the user.

REQ-2: The system shall utilize NLP algorithms to analyze the input text and identify key concepts.

REQ-3: The system shall generate an acronym based on the identified key concepts.

REQ-4: The system shall provide the generated acronym to the user as output.

REQ-5: Error Handling: If the input text is invalid or cannot be processed, the system shall display an error message to the user and prompt for valid input.

Two-Step Process

Description and Priority:

AcroMemory utilizes a sophisticated two-step process for content summarization, first employing advanced comprehension techniques to understand the input content, followed by keyword extraction to distill key concepts. This approach ensures accuracy in capturing essential information and generating acronyms that effectively encapsulate the essence of the input.

Priority: High

Benefit: 8 (Improves accuracy of summarized content)

Penalty: 2 (Increased processing time)

Cost: 5 (Development and integration of two-step process)

Risk: 4 (Potential complexity in implementation)

Stimulus/Response Sequences:

- i. User Action: User provides a paragraph of text to AcroMemory.
- ii. System Response: AcroMemory first comprehends the text using advanced techniques, then extracts keywords to generate an acronym summarizing the main points.

Functional Requirements:

REQ-1: The system shall accept input text from the user.

REQ-2: The system shall employ advanced comprehension techniques to understand the input text.

REQ-3: The system shall extract keywords from the comprehended text.

REQ-4: The system shall generate an acronym based on the extracted keywords.

REQ-5: The system shall provide the generated acronym to the user as output.

REQ-6: Error Handling: If the input text is invalid or cannot be processed, the system shall display an error message to the user and prompt for valid input.

Semantic Understanding

Description and Priority:

AcroMemory integrates semantic understanding to capture contextual relationships between words, enhancing the clarity and memorability of generated acronyms. This feature ensures that the software interprets and conveys the nuanced meaning of the input content, improving the quality of summarized information.

Priority: High

Benefit: 9 (Enhances clarity and memorability of generated acronyms)

Penalty: 1 (Potential for misinterpretation of context)

Cost: 6 (Development and integration of semantic understanding algorithms)

Risk: 4 (Dependence on the accuracy of semantic understanding algorithms)

Stimulus/Response Sequences:

- i. User Action: User enters a sentence with contextual nuances into AcroMemory.
- ii. System Response: AcroMemory interprets the contextual relationships between words in the sentence and generates an acronym that accurately reflects the nuanced meaning.

Functional Requirements:

REQ-1: The system shall accept input text from the user.

REQ-2: The system shall analyze the input text to identify contextual relationships between words.

REQ-3: The system shall utilize the identified contextual relationships to generate an acronym.

REQ-4: The system shall provide the generated acronym to the user as output.

REQ-5: Error Handling: If the input text is invalid or cannot be processed, the system shall display an error message to the user and prompt for valid input.

User Customization

Description and Priority:

AcroMemory provides options for user customization of the acronym generation process, allowing individuals to tailor the summarization experience to their preferences. This feature enhances user satisfaction and customization capabilities by enabling users to pri-

oritize specific keyword types for acronym generation.

Priority: Medium

Benefit: 7 (Enhances user experience and customization)

Penalty: 3 (Increased complexity for users)

Cost: 5 (Development and implementation of customization features)

Risk: 3 (Potential for user confusion)

Stimulus/Response Sequences:

- i. User Action: User accesses the customization settings in AcroMemory.
- ii. System Response: AcroMemory presents options for prioritizing specific keyword types, allowing the user to tailor the acronym generation process according to their preferences.

Functional Requirements:

REQ-1: The system shall provide options for user customization of acronym generation process.

REQ-2: The system shall allow users to prioritize specific keyword types for acronym generation.

REQ-3: The system shall save user preferences for future use.

REQ-4: Error Handling: If user input for customization is invalid, the system shall display an error message and prompt for valid input.

Enhanced Learning

Description and Priority:

AcroMemory aids students in learning by condensing complex topics into easily digestible acronyms, facilitating efficient information retention. This feature supports the educational process by providing succinct summaries of key concepts, improving learning efficiency, and aiding in the comprehension of complex subjects.

Priority: High

Benefit: 8 (Improves learning efficiency)

Penalty: 2 (Potential oversimplification of complex topics)

Cost: 3 (Integration with educational platforms)

Risk: 3 (Potential for misunderstanding or misinterpretation)

Stimulus/Response Sequences:

- i. User Action: Student inputs a complex topic or concept into AcroMemory.
- ii. System Response: AcroMemory generates an acronym summarizing the key points of the topic, aiding in the student's comprehension and retention of the information.

Functional Requirements

REQ-1: The system shall accept input text related to a topic from the user.

REQ-2: The system shall generate an acronym summarizing the key points of the topic.

REQ-3: The system shall provide the generated acronym to the user as output.

REQ-4: Error Handling: If the input text is invalid or cannot be processed, the system shall display an error message and prompt for valid input.

2.5 Other Nonfunctional Requirements

Performance Requirements

Response Time: Acronyms should be generated within 3s under normal conditions, ensuring swift access to the summarized content.

Scalability: The software must support 100 concurrent users and input texts up to 1000 words without performance degradation, guaranteeing smooth operation as user demand grows.

Accuracy: Achieve a 95 percentage accuracy rate in concept extraction to ensure that generated acronyms effectively capture the essence of the content.

Real-Time Processing: Acronyms should be generated within 1s for time-sensitive tasks, ensuring immediate access to summarized information.

Resource Utilization: Efficiently utilize system resources to ensure smooth operation across various hardware configurations, minimizing hardware requirements and optimizing performance.

Customization Flexibility: Provide users with options to prioritize specific keyword types, enhancing relevance and user satisfaction.

Safety Requirements

Data security: Securely store user data and implement encryption protocols to prevent unauthorized access. Utilize access controls to restrict access to sensitive information.

Prevention of misuse: Implement user authentication mechanisms to ensure only authorized users can access and use the software. Monitor usage patterns to detect and prevent malicious activities.

compliance: Comply with relevant data privacy regulations and obtain necessary certifications to protect user privacy. Implement measures to prevent unauthorized use or disclosure of personal information.

Ethical use: Define guidelines for ethical use and educate users about responsible usage practices. Ensure users understand the potential consequences of misuse or abuse.

Error handling: Implement robust error handling mechanisms to prevent data loss or corruption in case of software errors. Provide clear error messages to guide users in resolving issues.

Safety certification: Obtain relevant safety certifications to demonstrate compliance with industry standards. Conduct thorough testing and validation to meet certification requirements and ensure software safety and security.

Security Requirements

Data Encryption: Utilize strong encryption methods for storing and transmitting user data to safeguard against unauthorized access. Adhere to industry-standard encryption protocols to ensure the protection of sensitive information.

Access Controls: Implement role-based access control mechanisms to regulate user permissions and restrict access to sensitive data. Enforce strict authentication requirements to verify user identities before granting access.

Data Privacy Compliance: Ensure compliance with data privacy regulations like GDPR or CCPA to protect user privacy and ensure lawful processing of personal information. Implement privacy-by-design principles and obtain relevant certifications to demonstrate compliance.

Secure Transmission: Use HTTPS or other secure communication protocols to encrypt data transmitted between client and server components. Prevent unauthorized interception of data during transmission to maintain confidentiality.

Privacy Policies: Provide clear privacy policies informing users about data collection, use, and sharing practices. Obtain explicit consent from users before processing their personal information.

Software Quality Attributes

Usability: Achieve a user satisfaction score of at least 90 percentage in usability surveys, ensuring an intuitive interface and clear instructions.

Maintainability: Keep code complexity below 10 and ensure thorough documentation for easy maintenance and updates.

Robustness: Handle unexpected inputs gracefully, with stress tests confirming resilience under load spikes.

Testability: Attain a test coverage of at least 80 percentage through comprehensive test suites covering unit, integration, and end-to-end tests.

Adaptability: Utilize modular design for easy integration of new features, ensuring adaptability to changing requirements.

Interoperability: Test compatibility with popular platforms and systems, ensuring seamless integration.

Chapter 3

System Architecture and Design

3.1 System Overview

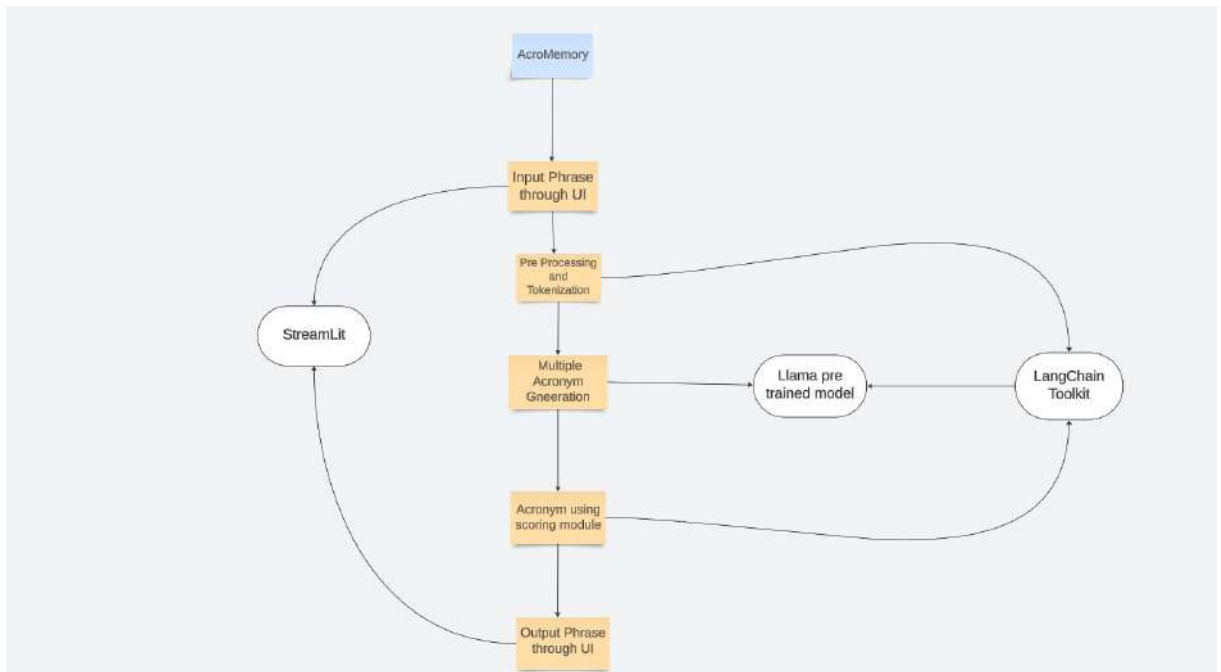


Figure 3.1: Architectural Model

The AcroMemory web browser is designed with a user interface enabling interaction via inputting sentences to generate coherent sentences from the acronym of important words, utilizing a pre-trained language model named Llama. This core browser application manages user requests, processes input sentences, and communicates with the Llama model. The Llama model, trained specifically for generating coherent sentences from acronyms, receives input from the browser, processes the significant words, and constructs a coherent sentence based on the acronym. To support these functionalities, a Natural Language Processing Module provides necessary tasks like tokenization and feature extraction. Users input a sentence via the interface, the browser extracts essential words, and the Llama

model generates an acronym and constructs a coherent sentence, all displayed through the user interface. The browser’s efficiency lies in its ability to quickly generate coherent sentences, facilitating mnemonic devices, summarization, or creative writing.

3.2 Architectural Design

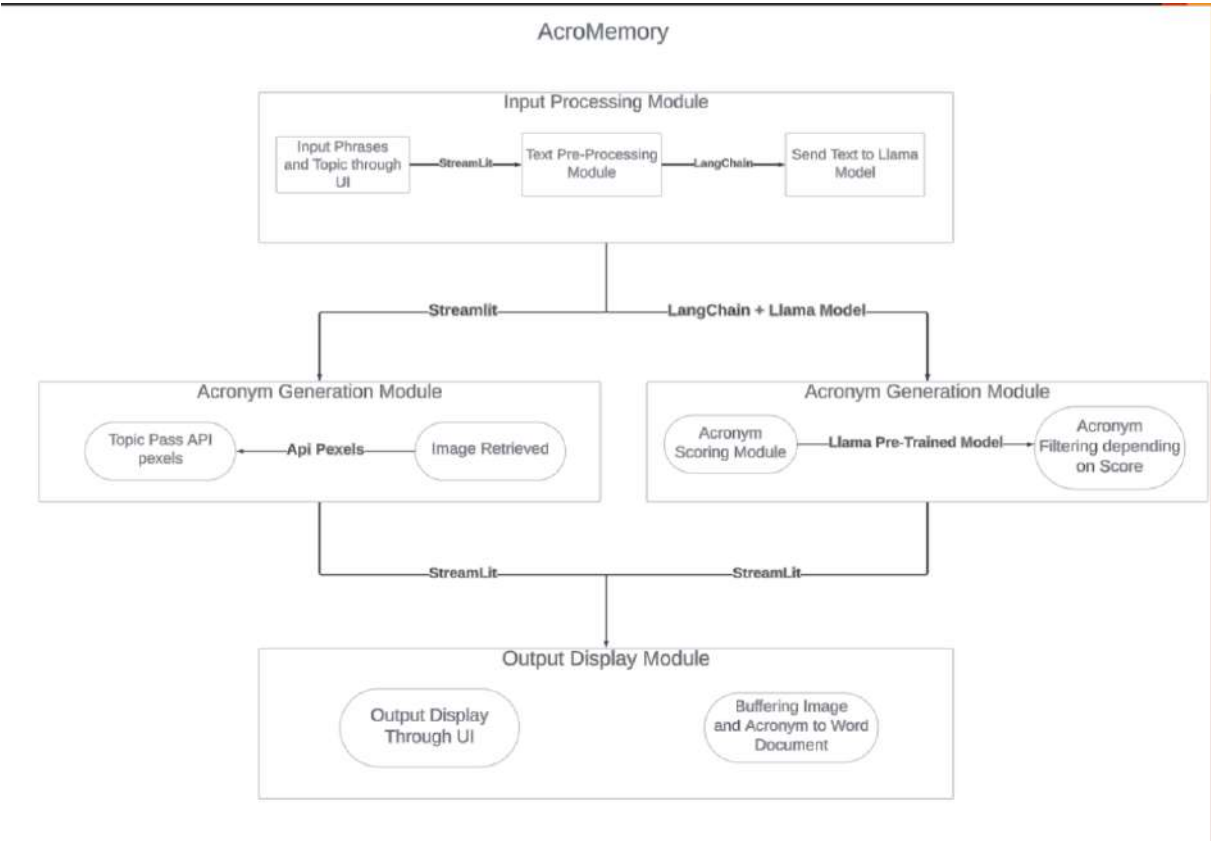


Figure 3.2: Architectural Diagram

3.3 Dataset identified

The dataset used comes from the pre-trained data model of Llama which consists of 13 billion parameters. It uses Natural Language Processing to form coherent sentences and use proper grammar.

3.4 Proposed Methodology/Algorithms

Function: getLLamaResponse(inputtext, acrostyle)

Inputs

1. inputtext: String containing the sentence to generate an acronym from. 2. acrostyle: String representing the desired output format ("Sentence" or "Acronym").

Outputs: 1. String containing the generated response from the Llama model.

Steps:

1. Create a 'llama' object using CTransformers to access the Llama model.
 - 1.1 Use the CTransformers library to instantiate the Llama model object.
2. Define a 'PromptTemplate' object named 'template' with placeholders for style and text.
 - 2.1 Instantiate a PromptTemplate object with placeholders for the style and input text.
3. Set the prompt template string instructing the model on generating the desired output (sentence or acronym) based on the input and style.
 - 3.1 Define a string template that guides the model on generating the output format based on the provided style and input text.
4. Format the template using 'prompt.format' with the provided 'acrostyle' and 'input-text'.
 - 4.1 Use the format method to fill in the placeholders in the template with the provided acrostyle and inputtext.
5. Call the 'llama' object with the formatted prompt to generate a response.
 - 5.1 Invoke the Llama model with the formatted prompt to generate the desired response.
6. (Optional): Print the response for debugging purposes.
 - 6.2 Print the generated response if necessary for debugging or logging purposes.
7. Return the generated response.
 - 7.1 Return the generated response from the Llama model.

Main Script

Steps:

1. Configure the Streamlit page title, icon, layout, and initial sidebar state.
 - 1.1 Set the configuration parameters for the Streamlit application, including the page title, icon, layout settings, and initial sidebar state.
2. Display a heading "Generate Acronyms" using 'st.header'.
 - 2.1 Render a header in the Streamlit application with the text "Generate Acronyms" to

provide a visual indicator of the application's purpose.

3. Create a text input field named 'inputtext' for users to enter their sentence.

3.1 Add a text input widget to the Streamlit application interface, allowing users to input their desired sentence.

4. Create a selectbox named 'acrostyle' with options "Sentence" and "Acronym" for users to choose the desired output format.

4.1 Include a selectbox widget in the Streamlit application interface, providing users with the option to select the desired output format.

5. Create a button named 'submit' for users to trigger the generation process.

5.1 Add a button widget to the Streamlit application interface, enabling users to initiate the acronym generation process.

6. If the 'submit' button is clicked:

6.1 Call the 'getLLamaResponse' function with the user-provided 'inputtext' and 'acrostyle'.

6.1.1 Invoke the 'getLLamaResponse' function with the user-provided inputtext and acrostyle parameters to generate the desired response.

6.2 Display the returned response from the Llama model using 'st.write'.

6.2.1 Render the generated response from the Llama model on the Streamlit application interface using the 'st.write' function.

3.5 User Interface Design

.



Figure 3.3: User Interface

3.6 Database Design

There is no such database that is used except the one that was used to originally train the pre-existing Llama model. The system makes use of this already trained Llama model and does not require a database.

3.7 Description of Implementation Strategies

The provided code sets up a Streamlit application for the AcroMemory app, allowing users to generate acronyms from input sentences using the LLama 2 language model. It leverages several libraries and modules:

1. Streamlit: Streamlit is a Python library used for building interactive web applications for machine learning and data science projects. It simplifies the development process by enabling developers to create UI components using familiar Python syntax.
2. langchain.prompts.PromptTemplate: This custom module likely provides functionality for defining templates with placeholders for generating prompts for language models. It allows for dynamic generation of prompts based on specified variables.
3. langchain.llms.CTransformers: Another custom module, this likely provides an interface for interacting with language models, specifically the LLama 2 model in this case. It handles model initialization, input formatting, and response retrieval.

The code itself can be summarized as follows:

- It defines a function 'getLLamaResponse(inputtext, acrostyle)' responsible for interfacing with the LLama 2 model to generate responses based on user input.

- The Streamlit page configuration is set up using `'st.setpageconfig'`, specifying parameters such as title, icon, layout, and initial sidebar state.
- A Streamlit header titled "Generate Acronyms" is displayed using `'st.header'`.
- A text input field is provided for users to enter their sentences via `'st.textinput'`.
- A selectbox is presented for users to choose between generating a full sentence or just an acronym using `'st.selectbox'`.
- A "Generate" button is included to initiate the acronym generation process with `'st.button'`.
- Upon clicking the "Generate" button, the `'getLLamaResponse'` function is called with the user-provided inputtext and acrostyle, and the generated response is displayed using `'st.write'`.

Overall, this code constructs an interactive interface for users to input sentences and choose whether to generate full sentences or acronyms, providing a seamless experience for utilizing the AcroMemory app.

3.8 Module Division

Modules used from Langchain

Llama Wrapper:

Langchain provides wrappers for various LLMs, including a potential wrapper for the Llama model.

This wrapper simplifies interacting with Llama, allowing the script to send text prompts and receive generated outputs.

Text Pre-Processing Module:

Tokenization: Breaking down the user input sentence into individual words.

Lowercasing: Converting all characters to lowercase for consistency.

Punctuation Removal: Removing punctuation marks that might confuse the model.

Text Cleaning: Handling special characters, typos, or other noise in the input

Acronym Scoring Module

Length Score: Shorter acronyms generally receive higher scores.

Pronounceability Score: This could involve checking if the acronym is easy to say out

loud using external libraries.

Relevance Score: You can define rules to check if the generated acronym reflects the meaning of the original phrase.

Filtering Module

Langchain’s filtering functionalities can be used to select the top-scoring acronyms based on user-defined settings.

The script can define a threshold score or a fixed number of acronyms to return.

Fitering acronym using scoring module

- Input and Output Parse: Madhav M. Menon and Jerin Varghese
- Drop Down Box for Formatting Words Within Input: Madhav M. Menon and Jerin Varghese
- Multiple acronym generation using Llama: Issac Mathew Jaimon and Jerin Varghese
- Tom
- Fiterting Acronym Using Scoring Module: Issac and Jerin Varghese
- Tom
- Text Pre Processing Module: Issac and Jerin Varghese
- Tom

3.9 Work Schedule - Gantt Chart

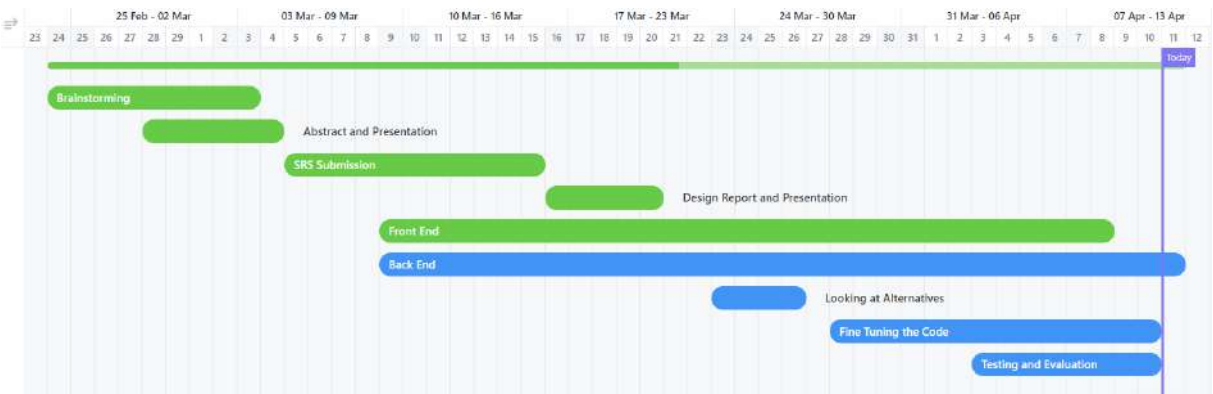


Figure 3.4: Gantt Chart

Chapter 4

Results and Discussions

4.1 Overview

The software provides a solution for users to generate acronyms and corresponding images based on input text and its related topic. This innovative tool streamlines the memorization process by condensing lengthy information into memorable acronyms while enhancing comprehension through visual aids. While the overall results demonstrate significant utility and creativity, some occasional inaccuracies are observed due to constraints such as model complexity and computational resources. Despite these limitations, the software remains a valuable asset for mnemonic learning and visual reinforcement, catering to diverse educational and professional needs.

4.2 Testing

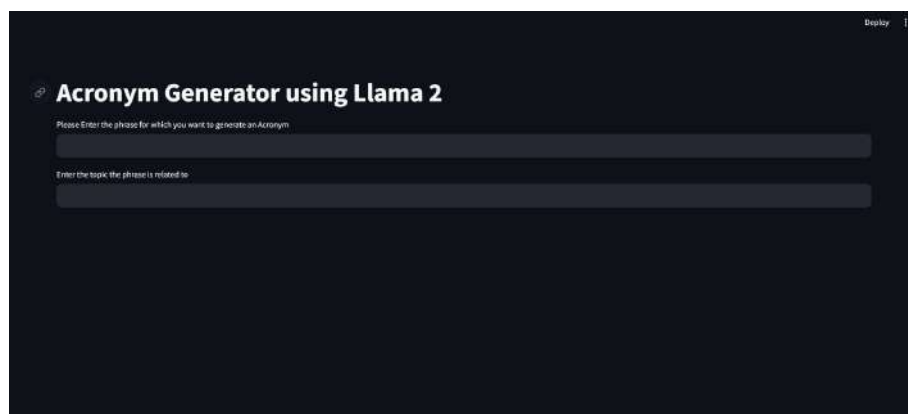


Figure 4.1: Interface to input text



Figure 4.2: Output generation



Figure 4.3: Text document

4.3 Quantitative Results

As the sentence generated always has semantic understanding, we are measuring the accuracy of our results through the keywords successfully identified. Testing accuracy based on 5 random test cases gives the following accuracy (where numerator represents the number of keywords successfully identified and the denominator represents the total number of keywords)

On taking an average of these results, the accuracy of our model comes out to be 59.5 per cent.

$$\frac{6}{9} \times 100 = 66.66\%$$

$$\frac{5}{7} \times 100 = 71.42\%$$

$$\frac{2}{5} \times 100 = 40\%$$

$$\frac{6}{8} \times 100 = 75\%$$

$$\frac{4}{9} \times 100 = 44.44\%$$

Figure 4.4: Accuracy of Random Cases

4.4 Graphical Analysis

This graph shows the accuracy of our model through week 1 till week 6. This was done by fine tuning the code and improving the prompts.

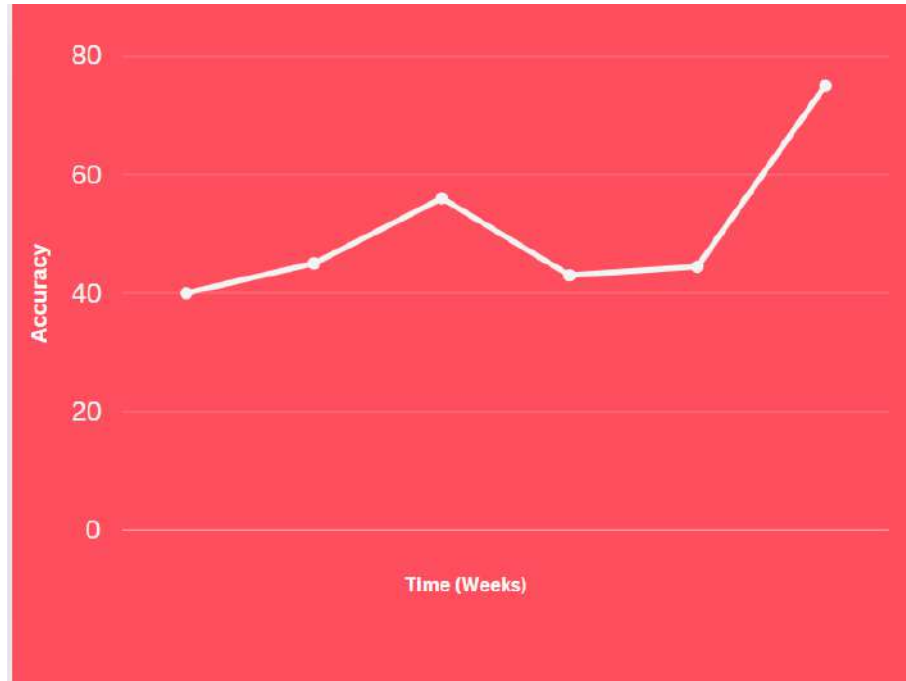


Figure 4.5: Graph Showing Accuracy of Model

4.5 Discussion

The results attained after using the acronym generator are somewhat accurate. The sentences are coherent but do not always extract the correct letters. There are small instances of words repeating. The picture generated along with the acronyms are accurate and of good quality. These visual cues also help in remembering the acronym.

Chapter 5

Conclusion

5.1 Conclusion

In conclusion, the development of AcroMemory marks a significant milestone in the realm of content summarization, offering a novel approach to aiding comprehension and retention of complex topics. Throughout the project lifecycle, our team has dedicated efforts to distill intricate information into easily digestible acronyms. By streamlining the summarization process, AcroMemory has proven instrumental in facilitating our understanding of challenging subjects, particularly in preparation for exams.

The deployment of AcroMemory has not only provided us with a powerful tool for personal learning but has also underscored the importance of leveraging technology to enhance educational outcomes.

5.2 Future Scope

As we reflect on the project's outcomes, we recognize the broader implications of AcroMemory in promoting efficient study habits and fostering deeper understanding across various academic disciplines. Moving forward, we are committed to further refining AcroMemory's capabilities, ensuring its accessibility and effectiveness for learners of all backgrounds. We will also be doing our best to add user customizations and refining the acronyms generated for better understanding.

References

- 1)mdpi.com (article on Llama and how to implement it well and use them as a guide)
- 2)researchgate.net (Use the resources and papers present there to better understand LLM and its workings)
- 3)Getting Started wioth StreamLit for Data Science(book)

Appendix A: Presentation

AcroMemory

DESIGN PRESENTATION

1

Anu Maria Joykutty

Issac Mathew Jaimon
Madhav M Menon
Jerin Varghese Tom
Jerin Varghese

AcroMemory

5/15/2024

Contents

2

- Introduction
- Problem Definition
- Objectives
- System Architecture
- Datasets
- UI Design
- Work Division - Gantt Chart
- Software/Hardware Requirements
- Conclusion
- References

AcroMemory

5/15/2024

Introduction

3

- AcroMemory is an application that generates acronyms depending on user input.
- Automated Process
- Generate original and Creative Acronyms.
- Acronyms can be changed according to user preferences.
- Comprehensive and summarisable approach to large pieces of text.

AcroMemory

5/15/2024

Problem Definition

4

- Challenge lies in summarizing complex Information to enhance comprehension and retention.
- Distill Key concepts from text and knowledge dissemination.

AcroMemory

5/15/2024

Objectives

5

1. Inputting the text to convert
2. Input topic related to text
3. Llama Text Processing
4. Acronym Output Generated
5. Topic Related Picture Generation

AcroMemory

5/15/2024

Scope and Relevance

6

- Generate meaningful and sensible acronyms.
- Word limit to inputted text. (Limit of: 25 words).
- Most useful for students for memorable and enhanced learning.
- Help in naming Projects and Products.
- Browser and Extension Support.

<Project Title>

5/15/2024

System Design

7

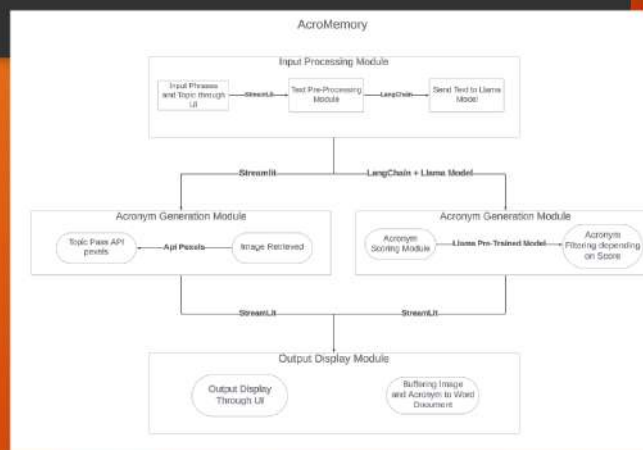
- User Interface (UI)
- Backend Processing
- Langchain Module
- Document Generation
- Scalability and Performance

<Project Title>

5/15/2024

System Design: Architectural Design

8



AcroMemory

5/15/2024

System Design: Module Division

9

Input Processing Module:

- Llama Wrapper
 - Langchain provides wrappers for various LLMs, including a potential wrapper for the Llama model.
 - This wrapper simplifies interacting with Llama, allowing the script to send text prompts and receive generated outputs.
- Text Pre-Processing Module
 - Tokenization: Breaking down the user input sentence into individual words.
 - Lowercasing: Converting all characters to lowercase for consistency.
 - Punctuation Removal: Removing punctuation marks that might confuse the model.
 - Text Cleaning: Handling special characters, typos, or other noise in the input.

AcroMemory

5/15/2024

System Design: Module Division

10

Acronym Generation Module:

Output Acronyms generated by pre-trained Llama model through the use of in-built NLP algorithms and techniques. These techniques are used on the pre-processed text and generates multiple Acronyms.

- Acronym Scoring Module
 - Length Score: Shorter acronyms generally receive higher scores.
 - Pronounceability Score: This could involve checking if the acronym is easy to say out loud using external libraries.
 - Relevance Score: You can define rules to check if the generated acronym reflects the meaning of the original phrase.

AcroMemory

5/15/2024

System Design: Module Division

11

- Picture Generation Module:

Pexels API

Topic-Based API Request: The module takes a given topic, constructs a search query, and sends it to the Pexels API. It includes necessary parameters such as the API key and search keyword to retrieve relevant images.

Image Retrieval and Delivery: The module processes the API response, selects the most relevant image, downloads it, and sends the image to the software, ensuring it receives a topic-related picture for use.

AcroMemory

5/15/2024

System Design

12

Output Generation Module

- Acronym Display

- Text and Image Integration: The module displays the output in a clear, text format accompanied by a relevant image on the side. This combination ensures that users can easily read and visually contextualize the information presented.
- Download Option: Users have the option to download the document, which is pre-buffered to ensure quick and seamless access. This feature allows users to save and share the content conveniently.

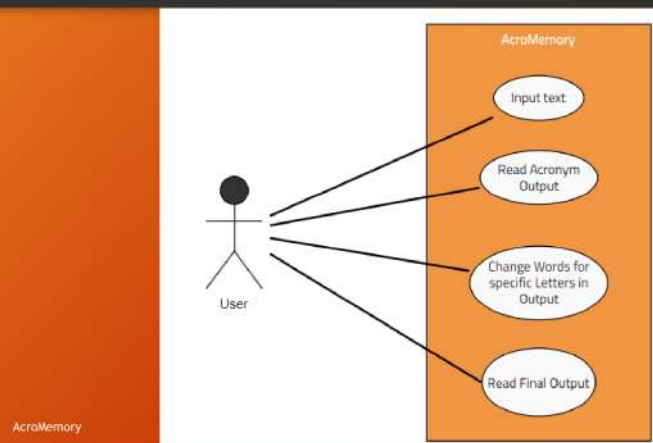
AcroMemory

5/15/2024

System Design

Use Case Diagram

13



Datasets

• Sample dataset and source

14

RedPajama V2 is a dataset with 30 T tokens that is used to train and fine tune the Llama 2 Model.

Phrase	Acronym
Profession and Ethical Duty	Pushing Endless Drawbacks
Light is a Refractive Model	Let's illuminate, refract, and magnify
Mitochondria is the powerhouse of a cell.	Many Intelligent People Chatting
Black Holes Bend Light.	Brilliant Hunters Brave the Latter.

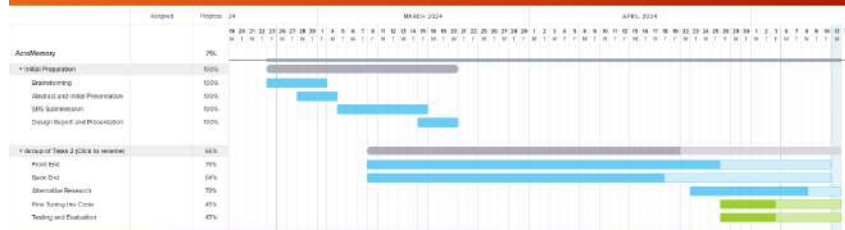
AcroMemory

5/15/2024

Work Division

15

• Gantt Chart



AcroMemory

5/15/2024

Software/ Hardware Requirements

16

- Software specifications
 - LangChain toolkit
 - StreamLit
 - Llama-2 pretrained model(13b parameters)
- Hardware specifications
 - 12GB GPU Nvidia GTX 1660 Ti
 - Monitor
 - Keyboard

AcroMemory

5/15/2024

Results

17

AcroMemory

Please Enter the phrase for which you want to generate an Acronym

Enter the topic the phrase is related to

Generate

AcroMemory

5/15/2024

Results

18

Generated Content by Llama 2

Your acronym has been generated successfully!

```
{
  "user_input":
  "Apple Banana Orange Strawberry
  Pineapple Mango Grape Watermelon
  Kiwi Pear",
  "text":
  "Apple Banana Orange
  Strawberry Pineapple Mango Grape
  Watermelon Kiwi Pear
  "A Banana-eating Bear once stole a
  pear from a kiwi farm."
}
```

Fetches Image



Final Doc to Download

Download Word Document

AcroMemory

5/15/2024

Future Enhancements

19

- Multiple Acronym generation with user customization
- Adding subsequently generated acronyms to the same document
- Increasing accuracy of generated acronym

<Project Title>

5/15/2024

Conclusion

20

- Sensible and Meaningful Acronym Generation from given Text.
- Automated Process.
- User Customization for Outputted Acronyms.

AcroMemory

5/15/2024

References

21

- <https://www.mdpi.com/2504-2289/7/4/182>
- https://www.researchgate.net/profile/Oguzhan-Topsakal/publication/372669736_Creating_Large_Language_Model_Applications_Utilizing_LangChain_A_Primer_on_Developing_LLM_Apps_Fast/links/64d114a840a524707ba4a419/Creating-Large-Language-Model-Applications-Utilizing-LangChain-A-Primer-on-Developing-LLM-Apps-Fast.pdf
- https://books.google.co.in/books?hl=en&lr=&id=9804EAAQBAJ&oi=fnd&pg=PP1&dq=streamlit+python&ots=qngveKNJ8M&sig=7DbhN47se9hEHhZnA57nP2x9304&redir_esc=y#v=onepage&q=streamlit%20python&f=false

Appendix B: Vision, Mission, Programme Outcomes and Course Outcomes

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
RAJAGIRI SCHOOL OF ENGINEERING & TECHNOLOGY (AUTONOMOUS)
RAJAGIRI VALLEY, KAKKANAD, KOCHI, 682039
(Affiliated to APJ Abdul Kalam Technological University)



Vision, Mission, Programme Outcomes and Course Outcomes

Institute Vision

To evolve into a premier technological institution, moulding eminent professionals with creative minds, innovative ideas and sound practical skill, and to shape a future where technology works for the enrichment of mankind.

Institute Mission

To impart state-of-the-art knowledge to individuals in various technological disciplines and to inculcate in them a high degree of social consciousness and human values, thereby enabling them to face the challenges of life with courage and conviction.

Department Vision

To become a centre of excellence in Computer Science and Engineering, moulding professionals catering to the research and professional needs of national and international organizations.

Department Mission

To inspire and nurture students, with up-to-date knowledge in Computer Science and Engineering, ethics, team spirit, leadership abilities, innovation and creativity to come out with solutions meeting societal needs.

Programme Outcomes (PO)

Engineering Graduates will be able to:

1. Engineering Knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

2. Problem analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

3. Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

4. Conduct investigations of complex problems: Use research-based knowledge including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

5. Modern Tool Usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

6. The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal, and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

7. Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

8. Ethics: Apply ethical principles and commit to professional ethics and responsi-

bilities and norms of the engineering practice.

9. Individual and Team work: Function effectively as an individual, and as a member or leader in teams, and in multidisciplinary settings.

10. Communication: Communicate effectively with the engineering community and with society at large. Be able to comprehend and write effective reports documentation. Make effective presentations, and give and receive clear instructions.

11. Project management and finance: Demonstrate knowledge and understanding of engineering and management principles and apply these to one's own work, as a member and leader in a team. Manage projects in multidisciplinary environments.

12. Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change.

Programme Specific Outcomes (PSO)

A graduate of the Computer Science and Engineering Program will demonstrate:

PSO1: Computer Science Specific Skills

The ability to identify, analyze and design solutions for complex engineering problems in multidisciplinary areas by understanding the core principles and concepts of computer science and thereby engage in national grand challenges.

PSO2: Programming and Software Development Skills

The ability to acquire programming efficiency by designing algorithms and applying standard practices in software project development to deliver quality software products meeting the demands of the industry.

PSO3: Professional Skills

The ability to apply the fundamentals of computer science in competitive research and to develop innovative products to meet the societal needs thereby evolving as an eminent researcher and entrepreneur.

Appendix C: CO-PO-PSO Mapping

COURSE OUTCOMES:

After completion of the course the student will be able to

SL. NO	DESCRIPTION	Blooms' Taxonomy Level
CO1	Identify technically and economically feasible problems (Cognitive Knowledge Level: Apply)	Level 3: Apply
CO2	Identify and survey the relevant literature for getting exposed to related solutions and get familiarized with software development processes (Cognitive Knowledge Level: Apply)	Level 3: Apply
CO3	Perform requirement analysis, identify design methodologies and develop adaptable & reusable solutions of minimal complexity by using modern tools & advanced programming techniques (Cognitive Knowledge Level: Apply)	Level 3: Apply
CO4	Prepare technical report and deliver presentation (Cognitive Knowledge Level: Apply)	Level 3: Apply
CO5	Apply engineering and management principles to achieve the goal of the project (Cognitive Knowledge Level: Apply)	Level 3: Apply

CO-PO AND CO-PSO MAPPING

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2	PSO 3
CO1	3	3	3	3		2	2	3	2	2	2	3	2	2	2
CO2	3	3	3	3	3	2		3	2	3	2	3	2	2	2
CO3	3	3	3	3	3	2	2	3	2	2	2	3			2
CO4	2	3	2	2	2			3	3	3	2	3	2	2	2
CO5	3	3	3	2	2	2	2	3	2		2	3	2	2	2

3/2/1: high/medium/low

JUSTIFICATIONS FOR CO-PO MAPPING

MAPPING	LOW/ MEDIUM/ HIGH	JUSTIFICATION
101003/CS6 22T.1-PO1	HIGH	Identify technically and economically feasible problems by applying the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
101003/CS6 22T.1-PO2	HIGH	Identify technically and economically feasible problems by analysing complex engineering problems reaching substantiated conclusions using first principles of mathematics.
101003/CS6 22T.1-PO3	HIGH	Design solutions for complex engineering problems by identifying technically and economically feasible problems.
101003/CS6 22T.1-PO4	HIGH	Identify technically and economically feasible problems by analysis and interpretation of data.
101003/CS6 22T.1-PO6	MEDIUM	Responsibilities relevant to the professional engineering practice by identifying the problem.
101003/CS6 22T.1-PO7	MEDIUM	Identify technically and economically feasible problems by understanding the impact of the professional engineering solutions.
101003/CS6 22T.1-PO8	HIGH	Apply ethical principles and commit to professional ethics to identify technically and economically feasible problems.
101003/CS6 22T.1-PO9	MEDIUM	Identify technically and economically feasible problems by working as a team.
101003/CS6 22T.1-PO10	MEDIUM	Communicate effectively with the engineering community by identifying technically and economically feasible problems.
101003/CS6 22T.1-PO11	MEDIUM	Demonstrate knowledge and understanding of engineering and management principles by selecting the technically and economically feasible problems.
101003/CS6 22T.1-PO12	HIGH	Identify technically and economically feasible problems for long term learning.
101003/CS6 22T.1-PSO1	MEDIUM	Ability to identify, analyze and design solutions to identify technically and economically feasible problems.
101003/CS6 22T.1-PSO2	MEDIUM	By designing algorithms and applying standard practices in software project development and Identifying technically and economically feasible problems.
101003/CS6 22T.1-PSO3	MEDIUM	Fundamentals of computer science in competitive research can be applied to Identify technically and economically feasible problems.
101003/CS6 22T.2-PO1	HIGH	Identify and survey the relevant by applying the knowledge of mathematics, science, engineering fundamentals.

101003/CS6 22T.2-PO2	HIGH	Identify, formulate, review research literature, and analyze complex engineering problems get familiarized with software development processes.
101003/CS6 22T.2-PO3	HIGH	Design solutions for complex engineering problems and design based on the relevant literature.
101003/CS6 22T.2-PO4	HIGH	Use research-based knowledge including design of experiments based on relevant literature.
101003/CS6 22T.2-PO5	HIGH	Identify and survey the relevant literature for getting exposed to related solutions and get familiarized with software development processes by using modern tools.
101003/CS6 22T.2-PO6	MEDIUM	Create, select, and apply appropriate techniques, resources, by identifying and surveying the relevant literature.
101003/CS6 22T.2-PO8	HIGH	Apply ethical principles and commit to professional ethics based on the relevant literature.
101003/CS6 22T.2-PO9	MEDIUM	Identify and survey the relevant literature as a team.
101003/CS6 22T.2-PO10	HIGH	Identify and survey the relevant literature for a good communication to the engineering fraternity.
101003/CS6 22T.2-PO11	MEDIUM	Identify and survey the relevant literature to demonstrate knowledge and understanding of engineering and management principles.
101003/CS6 22T.2-PO12	HIGH	Identify and survey the relevant literature for independent and lifelong learning.
101003/CS6 22T.2-PSO1	MEDIUM	Design solutions for complex engineering problems by Identifying and survey the relevant literature.
101003/CS6 22T.2-PSO2	MEDIUM	Identify and survey the relevant literature for acquiring programming efficiency by designing algorithms and applying standard practices.
101003/CS6 22T.2-PSO3	MEDIUM	Identify and survey the relevant literature to apply the fundamentals of computer science in competitive research.
101003/CS6 22T.3-PO1	HIGH	Perform requirement analysis, identify design methodologies by using modern tools & advanced programming techniques and by applying the knowledge of mathematics, science, engineering fundamentals.
101003/CS6 22T.3-PO2	HIGH	Identify, formulate, review research literature for requirement analysis, identify design methodologies and develop adaptable & reusable solutions.

101003/CS6 22T.3-PO3	HIGH	Design solutions for complex engineering problems and perform requirement analysis, identify design methodologies.
101003/CS6 22T.3-PO4	HIGH	Use research-based knowledge including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
101003/CS6 22T.3-PO5	HIGH	Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools.
101003/CS6 22T.3-PO6	MEDIUM	Perform requirement analysis, identify design methodologies and assess societal, health, safety, legal, and cultural issues.
101003/CS6 22T.3-PO7	MEDIUM	Understand the impact of the professional engineering solutions in societal and environmental contexts and Perform requirement analysis, identify design methodologies and develop adaptable & reusable solutions.
101003/CS6 22T.3-PO8	HIGH	Perform requirement analysis, identify design methodologies and develop adaptable & reusable solutions by applying ethical principles and commit to professional ethics.
101003/CS6 22T.3-PO9	MEDIUM	Function effectively as an individual, and as a member or leader in teams, and in multidisciplinary settings.
101003/CS6 22T.3-PO10	MEDIUM	Communicate effectively with the engineering community and with society at large to perform requirement analysis, identify design methodologies.
101003/CS6 22T.3-PO11	MEDIUM	Demonstrate knowledge and understanding of engineering requirement analysis by identifying design methodologies.
101003/CS6 22T.3-PO12	HIGH	Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change by analysis, identify design methodologies and develop adaptable & reusable solutions.
101003/CS6 22T.3-PSO3	MEDIUM	The ability to apply the fundamentals of computer science in competitive research and prior to that perform requirement analysis, identify design methodologies.
101003/CS6 22T.4-PO1	MEDIUM	Prepare technical report and deliver presentation by applying the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
101003/CS6 22T.4-PO2	HIGH	Identify, formulate, review research literature, and analyze complex engineering problems by preparing technical report and deliver presentation.

101003/CS6 22T.4-PO3	MEDIUM	Prepare Design solutions for complex engineering problems and create technical report and deliver presentation.
101003/CS6 22T.4-PO4	MEDIUM	Use research-based knowledge including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions and prepare technical report and deliver presentation.
101003/CS6 22T.4-PO5	MEDIUM	Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools and Prepare technical report and deliver presentation.
101003/CS6 22T.4-PO8	HIGH	Prepare technical report and deliver presentation by applying ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
101003/CS6 22T.4-PO9	HIGH	Prepare technical report and deliver presentation effectively as an individual, and as a member or leader in teams, and in multidisciplinary settings.
101003/CS6 22T.4-PO10	HIGH	Communicate effectively with the engineering community and with society at large by prepare technical report and deliver presentation.
101003/CS6 22T.4-PO11	MEDIUM	Demonstrate knowledge and understanding of engineering and management principles and apply these to one's own work by prepare technical report and deliver presentation.
101003/CS6 22T.4-PO12	HIGH	Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change by prepare technical report and deliver presentation.
101003/CS6 22T.4-PSO1	MEDIUM	Prepare a technical report and deliver presentation to identify, analyze and design solutions for complex engineering problems in multidisciplinary areas.
101003/CS6 22T.4-PSO2	MEDIUM	To acquire programming efficiency by designing algorithms and applying standard practices in software project development and to prepare technical report and deliver presentation.
101003/CS6 22T.4-PSO3	MEDIUM	To apply the fundamentals of computer science in competitive research and to develop innovative products to meet the societal needs by preparing technical report and deliver presentation.
101003/CS6 22T.5-PO1	HIGH	Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
101003/CS6 22T.5-PO2	HIGH	Identify, formulate, review research literature, and analyze complex engineering problems by applying engineering and management principles to achieve the goal of the project.

101003/CS6 22T.5-PO3	HIGH	Apply engineering and management principles to achieve the goal of the project and to design solutions for complex engineering problems and design system components or processes that meet the specified needs.
101003/CS6 22T.5-PO4	MEDIUM	Apply engineering and management principles to achieve the goal of the project and use research-based knowledge including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
101003/CS6 22T.5-PO5	MEDIUM	Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools and to apply engineering and management principles to achieve the goal of the project.
101003/CS6 22T.5-PO6	MEDIUM	Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal, and cultural issues and the consequent responsibilities by applying engineering and management principles to achieve the goal of the project.
101003/CS6 22T.5-PO7	MEDIUM	Understand the impact of the professional engineering solutions in societal and environmental contexts, and apply engineering and management principles to achieve the goal of the project.
101003/CS6 22T.5-PO8	HIGH	Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice and to use the engineering and management principles to achieve the goal of the project.
101003/CS6 22T.5-PO9	MEDIUM	Function effectively as an individual, and as a member or leader in teams, and in multidisciplinary settings and to apply engineering and management principles to achieve the goal of the project.
101003/CS6 22T.5-PO11	MEDIUM	Demonstrate knowledge and understanding of engineering and management principles and apply these to one's own work, as a member and leader in a team. Manage projects in multidisciplinary environments and to apply engineering and management principles to achieve the goal of the project.
101003/CS6 22T.5-PO12	HIGH	Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change and to apply engineering and management principles to achieve the goal of the project.
101003/CS6 22T.5-PSO1	MEDIUM	The ability to identify, analyze and design solutions for complex engineering problems in multidisciplinary areas. Apply engineering and management principles to achieve the goal of the project.

101003/CS6 22T.5-PSO2	MEDIUM	The ability to acquire programming efficiency by designing algorithms and applying standard practices in software project development to deliver quality software products meeting the demands of the industry and to apply engineering and management principles to achieve the goal of the project.
101003/CS6 22T.5-PSO3	MEDIUM	The ability to apply the fundamentals of computer science in competitive research and to develop innovative products to meet the societal needs thereby evolving as an eminent researcher and entrepreneur and apply engineering and management principles to achieve the goal of the project.