1. INTRODUCTION

ARA (**A**rtificial Intelligence-powered **R**esearch **A**ssistant) represents a cutting-edge application against conventional note-taking applications. ARA is designed to enhance the research capabilities of the user by leveraging advanced AI technologies to revamp the way information is gathered, organized, and synthesized. ARA transcends notetaking, transforming your research process into a dynamic, interconnected web that can be understood and processed by Artificial Intelligence models.

2. BRIEF LITERATURE SURVEY

The growing volume of research information presents a significant challenge. To navigate this complex landscape, there has been a surge in the development of AI-powered research assistants, some of which are mentioned below.

2.1 Notion [1]

Notion is a collaboration platform with Markdown[2] and inducing kanban boards[3], tasks, wikis[4] and databases. It is a workspace for notetaking, knowledge and data management and project and task management.

Notion only provides rudimentary AI features which enable text prediction, generation and solving of simple arithmetic expressions.

2.2 Lex[5]

Lex learns from the text the user adds and uses it to generate new content. Academic and technical writing can be mentally draining, and Lex can take some of the grunt work. Lex helps the user focus on sharing their research findings rather than stressing about technical and scientific writing.

Lex only summarizes the content for the purposes of sharing and does not aid in the process of researching.

3. PROBLEM FORMULATION

Academic research involves overwhelming amounts of information, leading to:

3.1 Information Overload

Researchers struggle to manage and create vast amounts of notes, references, and articles, wasting time on organization and retrieval.

3.2 Lost connections

Disconnected notes and fragmented understanding hinder the identification of key insights and cross-disciplinary connections.

3.3 Missed Opportunities

Researchers overlook relevant advancements and potential collaborations due to inefficient searching and knowledge navigation.

3.4 Limited writing and presentation skills

Ineffective information processing hinders the ability to distill research into clear and compelling arguments.

3.5 Boosting research productivity

Efficient information retrieval, synthesis, and organization save researchers valuable time and effort.

3.6 Unveiling hidden insights

AI analysis reveals key connections and themes across data, fostering deeper understanding and innovation.

3.7 Expanding research horizons

ARA identifies relevant advancements and potential collaborations, propelling research towards groundbreaking discoveries.

3.8 Sharpening research skills

AI guidance improves critical thinking, information management, and communication skills, leading to better research outcomes.

4. OBJECTIVES

- Create a robust note-taking application tailored towards professionals, able to layout information in a presentable and easy-to-understand manner.
- Enhance information retrieval and analysis by using Public Domain, Creative Commons Licensed Content and Open-Source Information available on the internet.
- Search and summarize content in a way that is relevant to the context of the research work being undertaken.

5. METHODOLOGY

The application will be developed according to the Waterfall Model[15] of SDLC.

5.1 Backend

The backend portion of the project encapsulates the data storage, machine learning models, and authentication/authorization flows. Data storage is achieved using PostgreSQL[6], a REST[20] wrapper on top of it and a set of triggers that create a real-time notification system.

The machine learning models are stored and run with Tensorflow[8], Keras[9] and Python[7]. Large Language Models[10] will be used for content summarization and text generation.

Authentication and Authorization is handled using OAuth2[11].

5.2 Frontend

The front end will be built using Svelte[12] and a meta-framework called SvelteKit[13], the code will be written in such a way that the project is compatible with a wide range of systems and is installable under the Progressive Web Application[14] Framework.

6. DELIVERABLES

The deliverables for ARA are summarized in the following components.

6.1 ARA Progressive Web Application[14]

A fully functional installable application that embodies the note-taking application and allows for interaction with the AI models.

6.2 Backend System

The backend system is responsible for processing data, running machine learning models, and managing communication between the user interface and various components.

6.3 Data Storage and Management System

An organized and secure data storage system to manage research data efficiently. Data storage will involve a database or a combination of cloud-based storage solutions.

6.4 Security Features

ARA contains measures to protect sensitive data and ensure compliance with privacy regulations while safeguarding against potential cyber threats.

7. PROPOSED TIMELINE

The application will be developed according to the Waterfall Model[15] of SDLC. The proposed timeline is as follows.

Feasibility Study & Requirement Analysis: 2 weeks (Week 1 to Week 2)

Design Phase: 2 weeks (Week 3 to Week 4)

Coding Phase: 6 weeks (Week 5 to Week 10)

Testing Phase: 2 weeks (Week 10 to Week 12)

Maintenance: 13th week onwards

8. BUDGET

ARA will be built using a collection of open-source technologies which are free to use.

A sum of ₹5000/- is required to allow for utilization of AWS[16] services.

9. FACILITIES REQUIRED

The following are the proposed requirements for ARA.

9.1 Computational Power

High-performance computing resources, possibly with GPU acceleration, for training complex models and handling large datasets.

9.2 Data Storage and Management

A secure and scalable data storage system for managing and storing research data.

9.3 Data Privacy and Security Measures

Implementation of robust security protocols to ensure the privacy and protection of sensitive research data, especially if dealing with confidential or proprietary information.

9.4 AI Frameworks and Libraries

Utilization of popular AI frameworks and libraries such as TensorFlow[8] or PyTorch[19] for developing and deploying machine learning models.

9.5 Semantic Web Technologies[17]

Semantic Web Technologies[17] like RDF[18] and OWL[17] to allow easy access to online resources.

10. PROPOSED OUTCOME

The proposed outcome for ARA is to provide researchers with an advanced and efficient tool that enhances their capabilities, streamlines their workflows, and contributes to the overall productivity of research activities.

10.1 Increased Research Efficiency

Researchers should experience a significant improvement in efficiency, with the AI-powered Research Assistant automating routine tasks, accelerating data analysis, and providing quick access to relevant information.

10.2 Effective Information Retrieval

The system should excel in retrieving relevant information from diverse sources, leveraging knowledge graphs and semantic web technologies to provide contextual and comprehensive results.

10.3 Adaptability and Continuous Learning

The system should adapt to evolving research needs through continuous learning mechanisms, improving its performance over time based on user feedback and changing requirements.

10.4 Enhanced Collaboration

The Research Assistant should facilitate collaboration among researchers by enabling easy sharing of insights, documents, and findings within the platform, fostering a collaborative research environment.

Mini-Project Incharge	Project Guide
Chandrajeet Borkar	Prof. D. J. Chaudhari

REFERENCES

[1] Notion Labs, Inc. "Your connected workspace for wiki, docs & projects" Retrieved January 10, 2024, from *https://www.notion.so*

[2] The Daring Fireball Company LLC. "Markdown Syntax Documentation". Retrieved January 10, 2024, from *https://daringfireball.net/projects/markdown/syntax#philosophy*

[3] Atlassian "What is a Kanban Board?". Retrieved January 10, 2024, from *https://www.atlassian.com/agile/kanban/boards*

[4] Bo Leuf and Ward Cunningham. 2008. *The wiki way: Quick collaboration on the web*, Boston, Massachusetts: Addison-Wesley.

[5] Lex Inc. Retrieved January 10, 2024, from *https://lex.page/*

[6] The PostgreSQL Global Development Group "PostgreSQL". Retrieved January 10, 2024, from *https://www.postgresql.org/*

[7] Guido van Rossum and Python Development Team. 2018. *The Python Library Reference: Release 3.6.4*.

[8] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. 2015. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. Retrieved January 10, 2024, from *https://www.tensorflow.org/*

[9] François Chollet and others. 2015. Keras. https://keras.io. Retrieved January 10, 2024, from *https://keras.io*

[10] OpenAI "Better language models and their implications". Retrieved January 10, 2024, from *https://openai.com/research/better-language-models*

[11] OAuth "OAuth 2.0". Retrieved January 10, 2024, from *https://oauth.net/2/*

[12] Svelte "Svelte • Cybernetically enhanced web apps". Retrieved January 10, 2024, from *https://svelte.dev/*

[13] Svelte "SvelteKit • Web development, streamlined". Retrieved January 10, 2024, from *https://kit.svelte.dev/*

[14] Mozilla Corporation "Progressive web apps | MDN". Retrieved January 10, 2024, from *https://developer.mozilla.org/en-US/docs/Web/Progressive_web_apps*

[15] Kai Petersen, Claes Wohlin, and Dejan Baca. 2009. *The Waterfall Model in Large-Scale Development*. In Product-Focused Software Process Improvement, Springer Berlin Heidelberg, Berlin, Heidelberg, 386–400.

[16] Amazon Web Services, Inc. "Train Machine Learning Models - Amazon SageMaker Model Training - AWS". Retrieved January 10, 2024, from *https://aws.amazon.com/sagemaker/train/*

[17] W3C "Semantic Web FAQ". Retrieved January 10, 2024, from *https://www.w3.org/RDF/FAQ*

[18] W3C "RDF 1.1 Primer". Retrieved January 10, 2024, from *https://www.w3.org/TR/rdf11-primer/*

[19] The Linux Foundation "PyTorch". Retrieved January 10, 2024, from *https://pytorch.org/*

[20] Fielding, Roy Thomas. 2000. *Architectural Styles and the Design of Network-based Software Architectures*. Doctoral dissertation, University of California, Irvine.