CS 510 – ADVANCED INFORMATION RETRIEVAL TECHNIQUES
FINAL PROJECT PROPOSAL

Anirudh Ragavender Ramarathnam Madhavan
(ar80@illinois.edu)

Saairam Venkatesh
(saairam2@illinois.edu)

Sujithra Rajan
(rajan11@illinois.edu)

Krishna Anandan Ganesan
(kag8@illinois.edu)

Saairam Venkatesh is our project coordinator. We are looking to build and develop a novel idea in the ***Development Track*** and following is our project proposal:

SIGNIFICANCE:

- With an information explosion in the digital world where we are exposed to so much information as compared to the maximum level we can consume at a point in time, there is a need for aggregating all the information sources we view daily to go back and view them in the future.
- However, there is a lack of a structured mechanism to store and review such material by clustering them based on their similarities in their information domain.
- As far as college students and working professionals are concerned, they are left with very minimal time in their daily routine to catch up on all the latest development in their domain.
- Such a system would allow for people to aggregate their information based on the domain and review them at any given time.

FUNCTION AND USERS:

- We propose to develop a knowledge management system that takes in a document and classifies it based on the recognized labels.
- The labels can then be used to organize, search and manage the data without any loss in information and relevancy.
- For the moment, we believe the best way to develop this solution as a web-based application where you could save bookmarks based on their primary domain.
- The tool will serve as a knowledge management system which can help to store documents, categorize them based on recognized entities and allow for searching through them using the underlying knowledge graph.
- As we aim to develop this using Open-source software and intend to make it available to one and all. We believe that students and working professionals would benefit heavily from this tool.

APPROACH:

- We intend to leverage full-stack development with frontend being developed with ReactJS and backend being taken care of by Golang and a NoSQL database. The usage of a NoSQL database allows us to organize tags in a structured manner which would be useful to search the documents and future addition of tags would be very convenient in the case of a NoSQL database as compared to a relational database.

- We intend to use open-source large language models (LLMs) for facilitating label classification of identified tags in documents and potential ranking of documents relevant to the tags for the recommender algorithm.
- The risks we can foresee for now are related to the generated tags. There could be potential conflicts in the automatic arrangement of bookmarks within their primary tag and in case we want to diversify the collection of bookmarks under a primary tag, we would have to resolve these conflicts by creating sub-folders within the primary tag and this could deal with a bit of human intervention.
- Secondly, the overall accuracy of the system in classifying the documents to the primary tags will have to be measured and improved over time by fine tuning the model and working on different datasets.
- Finally, addressing the flavor that each user has with respect to arranging their bookmarks is an issue for which we would have to come up with a generic arrangement solution that would encompass the personal preferences of every user.

EVALUATION:

- The key evaluation metric for the system would be to estimate the effectiveness of the system to search, manage and save bookmarks in their primary tags effectively.
- To measure the correctness of the system, we intend to use basic accuracy metrics such as precision, recall and F1-Score.
- Some of the datasets we intend to use to test the system are the following:
    - **NYC Times Corpus.**
    - **Medium Articles Dataset from Hugging Face**

TIMELINE:

- The three major tasks we have identified for the project along with their estimated timeline are as follows:
    - Classifier – April 20, 2024 -> Build the classifier and evaluate the performance with test datasets.
    - Backend – April 30,2024 -> Integrate classifer with Backend system.
    - Frontend – May 05,2024 -> Integrate existing system with Frontend.

TASK DIVISION:

- Anirudh – Anirudh will be taking care of Backend and classifier.
- Saairam – Saairam will be taking care of Classifier.
- Sujithra – Sujithra will help with backend and frontend development.
- Krishna – Krishna will be taking care of Classifier and frontend development.