

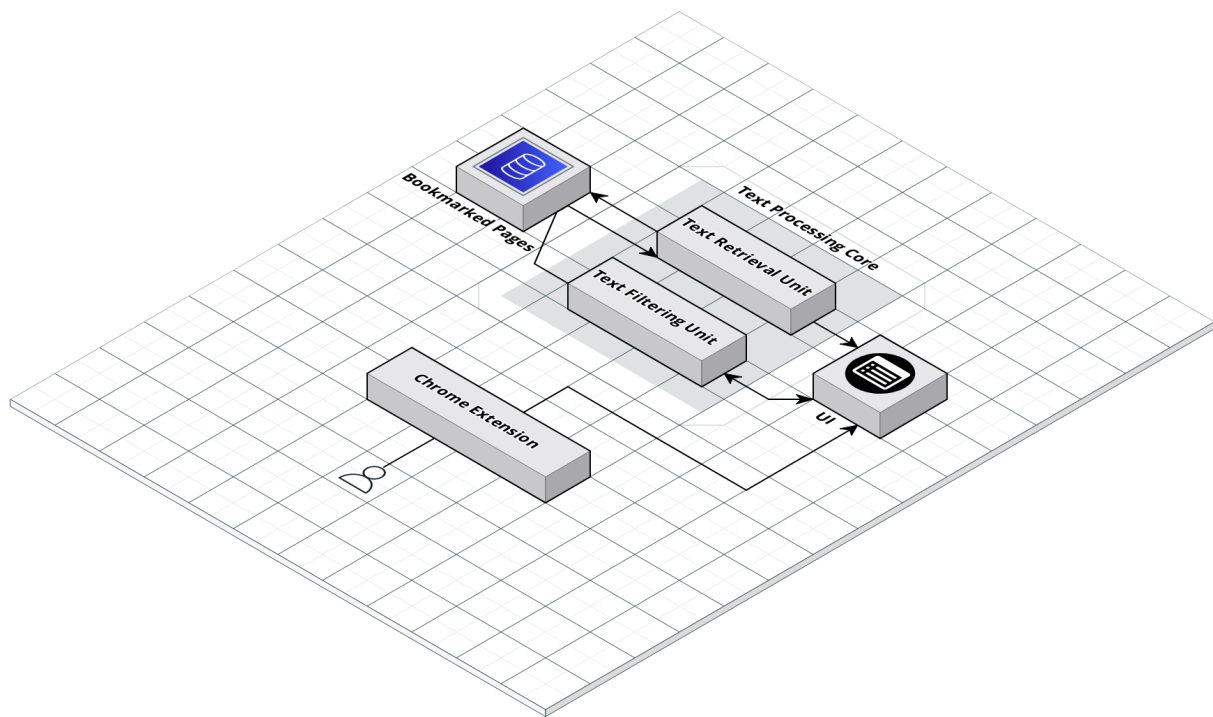
Project Progress Report

Xiaohan Tian (xtian13@illinois.edu)

1. Progress Made

So far, I've finished the task of finding and evaluating a JavaScript-based text processing toolkit for my course project as the base framework; I have chosen Lunr.js. I have completed a full analysis of the code implementation, performance evaluation, and different ways of customization.

Additionally, I have successfully embedded Lunr.js into a Chrome Extension and also wrapped it with a series of interface methods that I designed for my projects. Additionally, utility methods have been implemented as well.



(Figure 1. High-Level System Architecture)

Moving to the next step, I started working on the Chrome Extension UI a few days ago; so far, I have only implemented the prototype; there is still a lot of work that needs to be done, such as binding the backend logic, CSS decorations, etc.

Regarding the text processing core, I have finished the text retrieval unit; it has been tested and proven to work as expected. But I haven't started coding the text filtering unit yet; this feature will be implemented after UI is done since I need to make sure the text retrieval works end-to-end before starting working on the filtering part.

2. Pending Workload Analysis

Task	Original Est.	Pending Workload
Research suitable JS-based Text Processing Toolkit	2 hours.	0 hours
Embedding and Enhancing the chosen toolkit. Or, write one from scratch.	3-10 hours	1-2 hours
Build extension UI.	5 hours	4 hours
Implement the text retrieval unit.	7 hours	1 hour
Implement the text filtering unit.	7 hours	7 hours
Testing and evaluation.	3 hours	3 hours

3. Challenges

The main challenge I'm facing right now is to find a suitable way to capture the page content of existing bookmarks due to various very restricted limitations of the Chrome Extension system; more research needs to be done here. Since this is not a part of the core feature of text retrieval/text mining, the worst-case scenario is to build a small server application running on the host machine to perform the job.

The other challenge is to optimize the text filtering unit. I've done some early-stage implementation of it, but the performance is not very ideal. More research is needed in this area in order to make the recommendation system can work faster.