TextData Visual Search

General Information

Project Track: Development
Team Member Names & Emails: Dhruv Saligram (dhruvks2@illinois.edu)
Project Coordinator: Dhruv Saligram (dhruvks2@illinois.edu)

Functions and Users

The software tool I plan to implement is a visual search functionality within the TextData website. This project would be an extension of the existing TextData website, with the aim to hopefully have the final project fully integrated into the TextData platform by the end of the semester.

When users click on a community, they are currently met with a list of all the existing submissions. Even with less than 100 documents in our CS510 community, it is extremely difficult to meaningfully navigate the submissions. While there is a “Visualize” feature currently implemented that offers a visual map of all the submissions (as opposed to a list), the map seems to have no structure or organization. This again makes meaningful navigation difficult.

For my project, I envision users clicking the “Visualize” button and instead being met with a desktop-like folder organization. Each folder would have clear labels that describe the documents contained within it. The submissions within a community would essentially be organized in a tree – they would be split into broad categories, and broad categories would have more specific categories within them that would then contain individual submissions. There certainly exists a balance between how deep the folders should go versus how many individual documents can be contained within a single folder, which will be an integral part of the project’s development. The project’s key functionality is offering TextData users a way to easily navigate their communities based on the semantic themes of each individual submission.

The envisioned users of this tool would be all TextData users.

Significance

I believe that the TextData Visual Search tool is needed to help students engage more with other submissions, all while being able to do it in one centralized platform.

The inspiration for this project came with CS510 students in mind. With the task of upvoting certain vision essays, I found myself scrolling through every single submission, trying to find ones that resonated with me. Additionally, in order to help increase engagement with the vision essays, 2 separate websites were created by the course staff that students had to navigate. With this project, I hope to address these existing pain points – students can interact more meaningfully and without needing to check different sites, while course staff can avoid having to create new, but isolated, methods for engagement.

While this project would not address a major societal need or change the world, I believe it would make a significant impact in the classroom. Students would be more engaged in the class, they would be able to find vision essays that inspire them more easily, and the platform of TextData could become a central hub. Instead of posting project ideas on Campuswire or in the slide deck, students could put them on TextData and other students could easily find project ideas that interest them. Lecture transcripts from Instinfo and links to recordings could be uploaded to the community, allowing students to find lectures that covered specific topics they want to review, as opposed to cycling through every slide deck to find the relevant class. Ultimately, I believe that a better search functionality built into TextData would yield significant benefits.

Approach

Since TextData is open-source and has a publicly available GitHub, my plan to contribute to the platform is by eventually making a pull request that includes my finished feature. The main languages that are used in the repository are JavaScript and Python – while I am not extremely proficient with JavaScript, I do have a lot of experience with website development. Additionally, the server runs with Flask, which I do have a lot of familiarity with.

In terms of existing resources that I can leverage, there are a few that come to mind. First, the existing TextData code would help me generally get a sense of how to route data between the backend and frontend. Second, the code behind our TA Dean Alvarez's website based on hyperedge navigation between vision essays could also prove helpful, as a large component of my project would be making these hyperedges and then organizing them. Third, the neural generation tool already built into TextData might prove useful for creating topics for submissions. Finally, existing Hierarchical Multi-Label Classification Networks (HMCN) could potentially be utilized to help in the organization process.

In terms of risks and barriers, I do believe that setup could pose a challenge given the large-scale nature of TextData and the large number of components that make up the system. However, given TextData's connection to our course, I believe I could initially somewhat rely on course staff to help set up my environment and make local changes to the platform. The other main barrier I see is with actually integrating my project into TextData effectively, which again, I believe could be mitigated by communicating with the original creators of TextData.

Evaluation

I plan to demonstrate the correctness of my tool by designing my own curated set of submissions to a community and observing how well my algorithm's output compares to my manual organization. After this general testing, I hope to apply the backend to our current CS510 community and qualitatively observe how well the submissions are organized. This could potentially be bolstered by having other students in the course also test the tool and provide feedback in an extremely informal user feedback study. This student feedback would also help contribute to proving the usefulness of my tool.

Timeline

Week 1 (3/30 - 4/5)
- Get familiarized with the TextData codebase, set it up locally, and successfully make minor local changes to the platform

Week 2 (4/6 - 4/12)
- Design the backend of the feature
    - Determine a system to extract keywords / topics from submissions
    - Develop the splitting / organization algorithm

Week 3 (4/13 - 4/19)
- Create a sample dataset of submissions to test the backend system on
- Continue to refine & test the backend
- Route dummy data to a dummy frontend

Week 4 (4/20 - 4/26)
- Develop the frontend of the feature and visualize results

Week 5 (4/27 - 5/3)
- Refine code and potentially implement stretch goals

Week 6 (5/4 - 5/10)
- Write project report and work on final presentation

Task Division

I would be undertaking this project by myself.

Algos:

- Topic modeling
    - Hierarchical
- Hyperedge / hypergraph
- Clustering techniques

Hierarchical Multi-Label Classification Networks (HMCN)

Can have qualitative analysis of how deep folders should go versus how many documents max can be in one folder

Evaluation can be qualitative, but some quantitative metrics would be nice as well

To submit a proposal, follow the following steps:
1) Create a PDF version of your proposal and make it publicly available on github.
2) Add a submission to the CS510 Community on TextData, with 1) the title of your project , 2) the URL of your proposal on github, and 3) the hashtag "#proposal".

You may find it most convenient to make submission by using the Chrome Extension of TextData (i.e., while visiting your proposal github page, open the Extension, describe the submission with your project title, add the hashtag "#proposal" any where, and save to the CS510 Community. Of course, you can also do it by using the TextData website.

We want you to submit to TextData so that you can easily learn about ideas proposed by your peers as well as provide feedback to your peers. You will be asked to find three most interesting proposals and provide comments/feedback using TextData later.

The grading of your proposal will be based on the completion of the submission, i.e., if your team has submitted a proposal that has explicitly addressed all the aspects that your proposal should cover as explained above, you will receive full points. You will need to form a group and submit a link to the TextData submission here on the course webpage.