Advance Practical Computer Concepts for Bioinformatics

Final Project

Chih-Fan Lin

Introduction:

I choose to implement my own version of a RNAi database because I thought that the RNA interference mechanism on gene expression knockdown is very interesting. With RNAi, scientists can inhibit the gene expression of a specific gene. Although there is an existing RNAi database, my goal of this project is not replicating the existing database. Instead, I am trying to provide researchers a simple search tool to rapidly parse through the raw data with a few keywords. Moreover, the users can further narrow down their search within the result table search bar.

Implementation issues:

The first issue that I had was creating a properly working database on MySQL. Since I had the same database schema design for both fly and human data, I thought I could duplicate the fly tables, delete the data and rename the table. However, the first column 'id' is an auto-increment primary number, so it starts auto-incrementing from the previously removed data. This throws off the entire table, and the table was getting a lot more rows than it was supposed to. It took me a day or two to figure out that I should have deleted the first column and add it back after all the 'INSERT INTO' statement had gone through.

Another issue was implementing autocomplete for all search fields. I was trying to figure out a way to fit all the cgi search scripts of each field into a single cgi script. However, I was not able to reference individual 'term' variables for each field in the cgi script. Therefore, I had to split up the cgi scripts with their own SQL query.

Future Improvements and Discussion:

One of the future improvements is to allow users to choose the fields that they want to display in the result table. If users are only interested in 'gene_id' and 'gene_symbol', they should be able to view these two fields only. A possible solution is to implement checkboxes in the search.html page and let the users choose the fields to display. The SQL query will still query all the fields, but a javascript can selectively choose the fields to display on the template.html.

Another potential improvement is to allow users to download the search result as a tsv/csv file. Since a tsv/csv file can be opened with Excel, users can further aggregate the result data.

In my opinion, working on a personal project is quite rewarding. Especially when we are able to connect the backend information with the front end web page. I was constantly trying to improve the project and adding features. Although my web page is far from the standard of a modern web page, I was able to program the homepage to play a background video. In addition, since there are many modern jquery libraries, I was also able to easily implement a dynamic table on the result page. In conclusion, I learned a lot about connecting the backend information to frontend framework with this project.