CS410 Text Info System - Course Project Proposal

Fall 2023

Qinxi Wang

qinxiw2@illinois.edu

1. *What are the names and NetIDs of all your team members? Who is the captain? The captain will have more administrative duties than team members.*

   Team member: Qinxi Wang   Captain: Qinxi Wang

   Member NerdID: qinxiw2   Member email: qinxiw2@illinois.edu

   Since the team size is one, the captain will be the same member.

2. *What is your free topic? Please give a detailed description. What is the task? Why is it important or interesting? What is your planned approach? What tools, systems or datasets are involved? What is the expected outcome? How are you going to evaluate your work?*

My free topic will be search and recommendations around wine based on their attributes and reviews. The task is to build a program that support the following functionalities: 1) Search wine by description using free text; 2) Categorize reviews by positive, neutral, or critica sentiment; 3) Retrieve relevant wine search by variety, country, designation, etc; 3) Select a wine and click recommend similar wines via content based and collaborative filtering, and tree-based models to learn bias / weights on various discriminative features.

The main selling point of the project is to combine and utilize various things we have learned this semester, including search, retrieval and ranking, NLP techniques, text classification/categorization, and recommendations. This is attractive and unique because there is no existing system that applies all these on the dataset and provides all the functionalities listed above.

The planned approach at the moment is search by creating index and matching after text cleaning, topic identification (i.e is the user asking for a variety, winery, reviewer, or something else), sentiment analysis (for review), recs using content based, collaborative filtering, and another discriminative classification model. If the target is available in the dataset, I will split the data into train and test set; or more collective intelligence tasks such as sentiment analysis, the plan is to find any pre-trained one that pertain to our task and apply fine tune and transfer learning on top using the dataset we have.

The tools and system during the initial exploratory phase will be interactive python, jupyter notebook. For the final polished submission, the code and function will be cleaned into regular python classes and executable programs. If time permits I will look into Flask with HTML or Django for some lightweight but UI/UX friendly components. The dataset I plan to base the project on is https://www.kaggle.com/datasets/zynicide/wine-reviews/data.

The expected outcome is an interactive program that allows search and other exploratory activities with the wine and their attributes(variety, region, etc) and reviews. The program will provide all the functionalities listed in the first paragraph here. The evaluation will consist of two parts, offline and online. The offline evaluation we will use the classific machine learning metrics for each task, such as NDCG and MAP for search and ranking, precision and recall for recommendation tasks, etc. The online evaluation will instrument user interaction and their implicit feedback in real time, and provide that back to the system for later pipeline improvements, model retrain, and overall product improvements in search, retrieval and recommendations.

3. *Which programming language do you plan to use?*

   I plan to use Python for this project.

4. *Please justify that the workload of your topic is at least 20*N hours, N being the total number of students in your team. You may list the main tasks to be completed, and the estimated time cost for each task.*

   N = 1. The project will consist of the following main components:

   1. Data cleaning and exploratory - 2-3 hr
   2. Build search index and allow for search across text columns - 3-4 hr
   3. Sentiment analysis and opinion mining on the review text - 5-7hr
   4. Build similar wine recommendations using content based and collaborative filtering - 4-5 hr
   5. Implement functionality where a user can search, select, and retrieve similar / recommended wines - 3-5 hr
   6. Documentation of code, how to use, contribute, and adapt - 1 hr
   7. Run model eval for offline experiments, and add instrumentation of online metrics - 1-2 hr
   8. Produce a demo and write up for the final submission - 1-2 hr

   This adds to 20 - 28 hours total given the rough estimates.