Prototype Recommender System

Project Report

CS609

Spring 2021

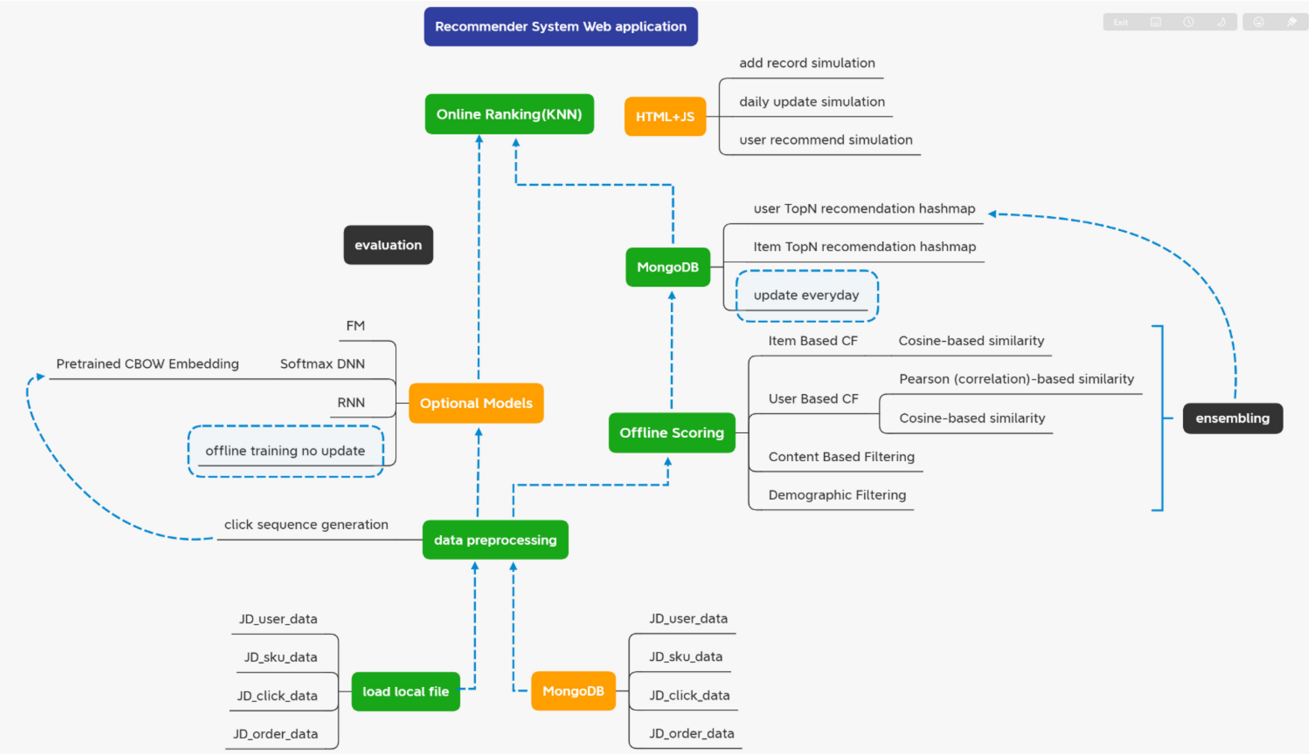
(Implementation track)

Team member(s): Wei Yang 10448858

1. Introduction

- **Problems**: Not all data is using yet since matrix is too large to load into memory.
- **Scenario**: A start-up company needs to build a new recommender system with exist users, items and events related data.
- **Goal**: Familiar with algorithms in recommender system and implement part of them as practicing; Build a web application.

2. System architecture



3. Progress so far

- **Progress:** As the figure above, the green modules completed.
- **Resource:** MongoDB, Node.Js, Python packages (Tensorflow, Keras, DeepCTR)
    - For factorization-machines: https://github.com/shenweichen/DeepCTR

4. Remaining work of the project

- **Progress:** As the figure above, the yellow modules are on the process.
- **Evaluation dataset**: This project will use the data on the last week to evaluate the system. The ID is encoded, which means we cannot know the name of them in life. It will probably make the recommendations outcome not that intuitive.
- **Preprocessing:** For the pre-processing part, the users without user information are removed from samples. Further, considering limited 32 GB RAM space and python thread lock on windows system, I only choose about 10% of user ID (40000) and 40% item ID (10000) yet. After everything is prepared, implementation will be tested on Linux system in AWS.
- **Evaluation measures**: Since this project will not have a truly online environment to make a A/B test, the system could only evaluate on existed data, which are not influenced by such system. Therefore, the CTR (Click-Through Rate) is not that proper for this system. The precision and recall will be used in this case.
- **Evaluation details:** For one model, it will provide 10 similar items that it thought (retrieved items).
    - For user-based models, we will calculate (***Precision***) the proportion of retrieved items(model prediction) that are clicked by user (ground truth) on next day. And we will take average of the result for the last week, as well as (***Recall***) the proportion clicked items (ground-truth) in the retrieved items(model prediction)

5. Conclusion

- The schedule is delay.
- There are two directions to dive. One is to build the website and make the system runnable. Another is to integrate more models.
- Although exploring more models in recommender system field is interesting, more time should be spent on learning the web programming techniques and make the system runnable. Integrating all the code still need to spend time as well.