Final Project

Due December 18th

Build a production-grade recommendation system.

The outcome of this project should be able to serve as a public project in your portfolio to advertise your skill set. Most importantly, it should showcase how you think about typical business problems through the lens of a data scientist. This includes both designing a technical solution and communicating your results.

Deliverable

The main deliverable is a GitHub repository with the following:

- A README file outlining the repository contents
- A requirements file with all software/package requirements to run your code
- A top level directory with all code
    - Include any support modules that you write
    - Include a notebook or markdown with your approach and basic results

Projects should be completed in Python (preferred) or R.

Data Scientists are more than statisticians and more than engineers. They solve real-life business problems by leveraging data and are usually brought in to brainstorm and frame a problem from day one. You will probably be expected to interface directly with other departments in the business, where your trust will be gained by sharing your insights into the core business problem at hand, and by translating these problems into a technical solution. This also means that you will be required to *communicate* your solution to key stake holders that are committed to solving the problem, but may not have your technical background.

Imagine a future employer looking at this repository. They would evaluate your project based on the solution's accuracy and the quality of your code, but they would also look for coherence and creativity. An employer would look for thoughtfulness and thoroughness; for example, how does the solution scale, were the hyper parameters tuned, will this solution discover novel recommendations, etc. Are the major contributions of your work clear? Did you call out important caveats?

These are the same standards on which you will be graded for these projects in this class.

Instructions for groups

Each team member has their own strengths and their own opportunities for development. It is ok for teams to divide and conquer for divisible tasks, but every team member should have a complete understanding of the entire solution, end to end. The entire team should brainstorm possible approaches and the deliverable before implementing a solution.

Data set

Use the Yelp Dataset Challenge from 2019:
https://www.yelp.com/dataset/challenge

You can create a ratings matrix and find active users with Python code similar to this:

```
import pandas as pd
import json
from tqdm import tqdm

line_count = len(open("review.json").readlines())

user_ids, business_ids, stars, dates = [], [], [], []

with open("review.json") as f:
   for line in tqdm(f, total=line_count):
        blob = json.loads(line)
        user_ids += [blob["user_id"]]
        business_ids += [blob["business_id"]]
        stars += [blob["stars"]]
        dates += [blob["date"]]

ratings = pd.DataFrame(
    {"user_id": user_ids, "business_id": business_ids, "rating": stars, "date": dates}
)

user_counts = ratings["user_id"].value_counts()
active_users = user_counts.loc[user_counts >= 5].index.tolist()
```

There is other information that you may find useful in the reviews file. Additional information in other files may also help, including metadata on the businesses (`business.json`), metadata on users (`user.json`), and tips (`tip.json`). You may even find the photos data set useful.

Instructions

Your goal is to predict the last rating for each active user. That is, for users's with 5 or more reviews, hold out their final review (by date) and make a prediction on the star rating of this final review. This can be viewed as a recommendation task. You can evaluate your recommendation performance with an error metric (i.e. how close your prediction is to the actual rating) and/or a ranking metric (i.e. for positive reviews, how highly do you rank the actual next-reviewed business? What about for negative reviews?). Carefully document your choice of performance criteria.

You can and should attempt this task with Collaborative Filtering (memory-based or model-based). This will serve as one of your baselines. Feel free to reuse code from your first project. Next, try to beat this baseline with techniques that you learned in class. You don't have to try every technique, but some possibilities include:

- Collective factorization techniques
- Content based models (text metadata, review text, or photos)

- Location-aware or time-aware models
- Deep learning models

You will not be graded on how well your models perform compared to baseline. You *will* be graded for how you apply the ideas from class to this problem, and how well you explain your results. For example, if you find that you cannot beat a baseline model, explain your hypotheses as to why, and perform an analysis to back your hypotheses. This project is an opportunity to take risks and apply yourself to new ideas - accuracy by itself does not matter, but make sure that you properly document your reasoning and results.

Put yourself in the role of “Senior Data Scientist” at a company that recommends local businesses. Your company evaluates itself on the ability to recommend the next best business for each user. The report that you deliver should be suitable for a technical manager that is well versed in both data science and business objectives. You should advocate for or against your solution - that is, do you think that a simple solution is enough for your use case, or do you think the company should invest time and money to implement a more sophisticated, and complicated, solution.

Report

Along with the code, document your project with a notebook or markdown.

1. Clearly outline your objectives
2. Provide benchmarks. These should include a baseline bias model and Collaborative Filtering at the very least. The results for every model and every metric that you use should be collected into a single table at the end.

linear regression parameters

3. Explore your model(s). Plot performance curves for important parameters. Do your models work equally well for all users and items? What about for less popular items or less prolific users? Think carefully about a business or technical framework to segment your data for users, items, or more, and test accuracy separately for these segments.
4. Devise methods to test for quality beyond just accuracy metrics. Consider testing coverage, novelty, serendipity etc
5. Can you think of any way to make your model better, like changing the objective, or adding in side information? It’s ok if you can’t make it better, but document your efforts and try to explain the results.

Evaluation

[30%] Technical correctness
[30%] Creativity

- Defining your model
- Optimizing your model
- Exploring your model

[40%] Presentation (see *Report* above)

- This is from the perspective of your technical manager. Help your manager solve their business problem. Does this project demonstrate that you can frame and solve a sophisticated personalization problem?