Exercise - Data Scientist

Problem: Book recommendations - „I like Lord of the Rings, what else should I read?“

Goal:

- take some data, try some approaches, produce some code, get some results
- then come and show us your solution and have a chat around it - show how you think about a specific problem, how you are able to explain what approach you used and why, think about the limitations of the approach and how things could be improved if there was more time, what you think of the results and if they make sense, etc.

Philosophy:

- the actual result and code are not that important – the journey there and potential future paths are more important
- if you are able to follow-through with some ideas then great, if you just start something and have a clear idea on how to proceed that is also useful
- the expectation is that you will spend an evening or two with the task (but there are no bounds to proactivity if you enjoy playing with the problem)

Data:

- available open dataset: https://www.kaggle.com/datasets/arashnic/book-recommendation-dataset?select=Ratings.csv
- alternatively feel free to grab any other relevant data set

Tools:

- use whatever you are comfortable in (R, Python, Matlab, Java, SQL,…) or feel free to use it as an opportunity to try out a new language
- it is not a contest in finding the best black-box library and blindly using it – own solutions are preferred even if they are simple
- how to present: up to you – slides, walking through code, drawing, …

BONUS: Exercise - Machine Learning Engineer

Problem: Productionalize the book recommender

Description:

- Imagine you have the recommendation model from the previous task. Now - you would like to build an application, where everyone can paste his favourite book and the application returns a list of recommended books - using the model.

Goal:

- come up with the application's architecture - which parts should the application have, how they should be connected etc.
- result should be ideally some reasonably detailed scheme (in PowerPoint/Lucidchart/draw.io/paper/…). If you trust your architecture (and want), you may actually build a lightweight incarnation of it. Something like a mini API that takes in the book name and returns the recommendations would be cool. But you are definitely not expected to materialize the entire architecture (you don't have to actually spin up a VM with a database; or code a website), still it's mainly about the concept.

Comments/Hints:

- there could be a lot of moving parts (new books/ratings could come => what to do; we create a better recommendation model => what to do; and many more)
- it would be interesting to think about particular technologies/frameworks that could be used (AWS for this, Oracle database for that, Spark for …) and what are the benefits of doing so;
- feel free to compare various architectures/technologies.
- there is no limit to the creativity and scope - the idea may grow well beyond the one app