

Personal Contribution

Our team focuses on creating a digital menu with built in recommendation systems that is suitable for restaurants that provide dine-in options. Our core focus of the application is on the 3 recommenders, namely Popular Items Recommendation, Similar Items Recommendation and Complementary Items Recommendation.

As the overall team lead for the project, I am tasked to oversee the progress of this project and assign tasks that suit the strength of each individual member.

Here is a break down of tasks that I have worked on:

Pre-development

- Project timeline and scope planning with team members
- Project research and idealisation
 - Includes manually synthesising the relationship between menu items and their related ingredients

Development

- Backend server implementation and deployment using Flask
- Knowledge graph construction using Neo4J
- Logic creation of similar items, complementary items and popular items within our knowledge graph (backend)
- Cypher queries development that include retrieval of popular items and similar items (backend)
- User interface design and user journey mapping, including prototyping

Post-development

- Project report writing
- Business case explanation video creation

Useful lessons Learnt

I have learnt a few useful techniques that allow me to represent knowledge, create reasoning models and deploy these models into an actual application.

Knowledge Representation

I have to use a variety of tools to represent the knowledge for this project. They include:

1. Rule based representation - a simple yet effective way to capture knowledge
2. Flow chart and graph - an efficient method to convey ideas in pictorial form
3. Tables - a concise way to represent groups of ideas

Reasoning Model Creation

I mainly learned how to represent knowledge using a graph (Neo4j) and I have the most fun developing the structure and retrieving certain knowledge from the graph. Infact, I learnt that most recommendations can be done efficiently and effectively through a well designed graph.

Reasoning Model Deployment

Lastly, I learnt how to apply our reasoning models onto the software development phase so that we can use them in our daily life.

Finally, I realised that an effective machine learning and reasoning model can be created using a hybrid of different reasoning models and they need not be complex to implement and understand.

Applying Knowledge and Skills in Other Situation or Workplaces

I will use the 3 steps approach described earlier (representing knowledge, creating reasoning models and deploying models) to create a learning and reasoning solution to problems that appear frequently in other situations or in my workplace.

For example, when it comes to taking care of my 3 months old daughter, my goal is to **optimize** for less crying situations and more sleeping time. My wife and I observe that her wake window is about 90 minutes and we represent this knowledge in the form of a timetable.

We then formulate a simple rule (communicated verbally) to know if she is hungry or sleepy. The formula is that if she cries within her wake window, she is very likely to be hungry and if it is beyond the window, she is likely to be sleepy.

Lastly, we deploy our model in her daily routine and record down the number of times she is crying or sleeping so that we know how accurate our knowledge is.