**Conversational Recommender System Project Instruction**

*All the previous capstone* ***weekly meeting slides*** *can be found in this folder:* [*https://drive.google.com/open?id=13HpdSRvGXE2o65obFEO4XRMotlKH9g5E*](https://drive.google.com/open?id=13HpdSRvGXE2o65obFEO4XRMotlKH9g5E)

*All the* ***documentations*** *for the project can be found in this folder:*

[*https://drive.google.com/drive/folders/1Z7IQ37hP4zbmX1uilBK-VBJrxUnq7dJP?usp=sharing*](https://drive.google.com/drive/folders/1Z7IQ37hP4zbmX1uilBK-VBJrxUnq7dJP?usp=sharing)

*The* ***Final Design Specification*** *document can be found using this link:* <https://docs.google.com/document/d/1IqQp5szMM--H6LMn1vEntEDKenbILv2izSJUxu0kGqk/edit?usp=sharing>

**Project code**

*\*All data files will be located at ../data/*

1. **Data preprocessing & Generation**

*This file processes the Toronto businesses data and the review data, filtered out businesses that are non-restaurants by identifying the selected keywords contained in the restaurant categories.*

**Data preprocessing code:**

* data\_processing.ipynb

**Imported Data files:**

* toronto\_reviews.csv
* businesses\_final\_toronto.csv

**Cleaned exported data files:**

* Cleaned\_Toronto\_Reviews.json
* Cleaned\_Toronto\_Business.json

1. **Project data generation (for Recommender System)**

*Run following code for data files generation:*

*python projectData\_generation.py –data\_dir ../data/ --data\_name Cleaned\_Toronto\_Reviews.json*

*Current application for the recommender system setting is item-based to make user-item preference predictions. The code currently leverages item-keyword information stored in vector space using TF-IDF to compute item-item similarities matrix. Therefore, the current item similarity matrix is generated by using the item-keyword matrix.*

**Data file generations code:**

* projectData\_generation.py

**Imported data files:**

* Cleaned\_Toronto\_Reviews.json

**Generated data files:**

* Dictionaries that maps item id to their attributes:
  + icDictionary.json
  + ipDictionary.json
  + isDictionary.json
* Dictionaries that maps items id to their distance to each intersection:
  + idDictionary\_yongefinch.json
  + idDictionary\_bloorbathurst.json
  + idDictionary\_spadinadundas.json
  + idDictionary\_queenspadina.json
  + idDictionary\_blooryonge.json
  + idDictionary\_dundasyonge.json
* Other relevant matrices
  + rtrain.npz
  + icmatrix.npz
  + IKbased\_II\_similarity.npy
  + UI\_prediction\_matrix.npy

1. **Explanation generation**

*This code reads in the Toronto business data file and generate 3 explanation phrases for each business. Detailed logic behind the generation of the explanation phrases can be found in the* [*Explanation documentation*](https://docs.google.com/document/d/11zMtXgZHFCdUvCfdODTGetrjGVim34eSZbKDg8M0bOM/edit?usp=sharing)*.*

**Data preprocessing code:**

* data\_processing.ipynb

**Imported data files:**

* Export\_TorontoData.json

**Exported explanation data file:**

* Toronto\_explanation.json

A screenshot of a cell phone

Description automatically generated

1. **Conversational Recommender System API**

*The file reads in all the generated data files, run the recommender system and connects it with an API. Different functions are enabled through different endpoints. Details for the design of these endpoints and user-system interaction(critiquing) logic can be found in the* [*API documentation*](https://docs.google.com/document/d/11gWgs_M_lJ__KCjHC2Ph4mFObI7yiAmg8s7AF3V9O0A/edit?usp=sharing) *&* [*Critiquing documentation*](https://docs.google.com/document/d/1C5m5kJXA5kzlxlxr3VzOXZYRzyuyD1kjfDSejaKupiI/edit?usp=sharing)*.*

**API code:**

* convSys\_API.ipynb

**Imported Data files:**

* Toronto\_explanation.json
* All the data files generated from *Section 2*

*Currently, the project uses 3 endpoints to manage the user-system interactions*

**A screenshot of a cell phone

Description automatically generated**

1. **Conversational Recommender System Middle tier (fulfillment agent)**

*This part of the code handles the middle tier of the project, which matches the user’s intents to the correct response, and send retrieve the corresponding responses for the intent.*

A screenshot of a cell phone

Description automatically generated

**Fulfillment code:**

* index.js

**Using ngrok for connecting the API as webhook for the conversational system:**

Initial Setup:

In order to let our local API to be connected to external network, the team used [ngrok](https://ngrok.com/docs) for this connection. The circled section needs to be changed for, as it is the ngrok deployed webhook address used by the group.

Please follow the ngrok [documentation](https://dashboard.ngrok.com/get-started) to set up your personal account and token.

A close up of a logo

Description automatically generated

Local API connection:

* *Run the API on local host (5002 as indicated in the screenshot below)*

**A screenshot of a cell phone

Description automatically generated**

* *After the API has been running at local host, run ngrok at your folder*

A screenshot of a cell phone

Description automatically generated

* *Ngrok will connect the API to the external network, select the following address to put into index.js file*

A screenshot of a cell phone screen with text

Description automatically generated

* *There are functions that maps the google dialog flow intents to the external API we created, this section in the file indicates the mapping of each intent to the function*

A screenshot of a cell phone

Description automatically generated

1. **Conversational Recommender System front end (Google Dialog Flow agent)**

*Use the following steps to import the capstone Google Dialog Flow project.*

**Dialogflow project to import:**

* Inago-Capstone-Recsys.zip

1. *To import an existing Google Dialogflow project, create a new project in Actions Console*

A screenshot of a cell phone

Description automatically generated

1. *Open project in Dialogflow, navigate to settings, Export and Import section and select Import from Zip. Then import the Inago-Capstone-Recsys.zip*

A screenshot of a cell phone

Description automatically generated

1. *Follow* [Google Assistant](https://developers.google.com/assistant/conversational/overview) *documentation to learn the basic functions and for the dialog flow project.*
2. **Additional files**

*These files contain the research code for the project but do not directly contribute to the operation of the conversational recommender system.*

**allMethods\_CrossValidation.ipynb:**

This research code compares all different algorithms the group considered for the recommender system. The cross-validation process was performed. All the functions were not formatted, the computation time is long. However, it gives a brief overview of all the different attempts the group used for the recommender system.

**algorithmSelection\_userStudy.ipynb**

This research code was used for computing the first user study operated by the group. Where the users will be exposed with 5 different algorithms used for making restaurant recommendations.

**similarityCalculation\_analysis.ipynb**

This file contains the research code for analysing different ways of computing the similarity matrices for user-based and item-based collaborative filtering methods for restaurant recommendations.