CS 410 Project write-up

The Project Topic is **WebUI for Nutritional Analysis**.

Task 1: Background

In this project, we will generate Knowledge on food consumptions based on examining the metabolite compositions of the human subjects' fetal samples.

I have to read a very well written Research Paper authored by TA Kowshika Sarker plus other co-authors (such as Professor Zhai) titled "**Augmenting Nutritional Metabolomics with a genome-scale metabolic model for assessment of diet intake**."

Human Fetal samples are homogenized and metabolomic analysis was performed before and after they take certain food (Almond, Avocado, Barleys, Broccoli, Oats, Walnut).

For each sample, concentrations for hundreds of metabolites are generated. This allows us to perform feature analysis based on the changes in the metabolite concentration.

Random Forests using the major metabolites are used to predict the original food intake.

Task 2: Python Scripts

We used an abbreviated version of the Python scripts created by the Research paper.

The Github Repository for the Original Scripts is in:

https://github.com/kowshikasarker/gem-met

The paper uses GEM (Genome-Scale Metabolic Model) to analyze metabolite concentration based on Reaction and Subsystem.

We will focus on Reaction-based analysis in the WebUI.

Python scripts in the src directories are:

webUIFlask.py - this is the Server side Python Script

preprocess-metabolome.py - this is the Python script for Preprocessing Metabolome

preprocess-human-gem-all.py - this is the Python script for Preprocessing GEM

preprocess-human-gem-sets.py - this is the Python script for Generating Reaction-Set dat

compute-change-feature.py – this is the Python script for generating the Change Feature.

compute-ratio-feature.py – this is the Python script for generating the Ratio Feature.

compute-prob-feature.py – this is the Python script for generating the Prob Feature.

run-classification.py, classification.py –scripts for generating RandomForest data

summarize-performance.py – this is the Python script for generating performance report.

Task 3: WebUI Design

Metabolite Concentration files Processing (2 input files, 2 mapping files, 2 output files)

Each food study has 2 set of samples, the Treatment set and the Control set.

The original script assumes 2 input files for the food studies.

One file contains the baseline (pre-consumption) metabolite concentrations, and the other file is the end (post-consumption) metabolite concentrations for the studies involved.

Notice that not all the metabolites measurable are the same in each file. We will eliminate metabolites that are present less than 20% amongst all the food studies. For the missing concentration, we will fill in a random value between 0 to 0.5*minimum concentration.

The goal of the Metabolite Concentration files Processing is to generate a concentration Change file that capture the delta (End – Baseline) concentration for all the metabolites that are common. There are total of 114 common metabolites for the paper. There is a mapping file that maps the name of these metabolites into the corresponding HMDB ID.

Given the Research Paper wants to use GEM Reactions, the metabolites extracted need to be present in these Reactions. Hence the 114 metabolites are further reduced to 75. There is a mapping file that maps the HMDB ID into MAM ID and the names used in GEM Reactions.

The WEBUI will provide 4 boxes for the 2 input files and 2 mapping files to be entered.

These 4 files will be copied locally into the right directories expected by the script and the corresponding output files will be generated.

3 output files are generated:

- One contains the study samples for the 75 metabolites with the name used in the original study
- One contains the study samples for the 75 metabolites using HMDB ID.

- One maps name in food study vs HMDB ID vs MAM ID vs name in GEM model.

The WebUI will generate output files needed for subsequent scripts into output directories.

GEM Model Processing (Step 1 – Processing All the Reactions)

The Human-GEM Model is captured in an Excel spreadsheet.

There is a sheet called METS that maps the MAM ID into the name used in the Reaction Equations. The Reaction Equations are listed in the RXNS sheet.

Each Reaction has a MAR ID and the associated equation is represented in textual format using the names listed in the METS sheet. The Reactions are also grouped based on the Subsystem they belong to.

Per the Research Paper, 2 Subsystems' Reactions should be removed. These are Transport reactions and Exchange/demand reactions.

In addition, the Reaction can be Forward only or Reversible.

Metabolites on left side of the Equation is called Substrate.

Metabolites on right side of the Equation is called Product.

First step of GEM Model Processing is to parse the Excel spreadsheet to create a file that generate for each Reaction, how many metabolites it has as Substrate or Product and how many of these metabolites are measured in our food study.

The WebUI generates data for this step without asking for input files from the users.

GEM Model Processing (Step 2 – Reaction Set Generation)

The Reearch Paper create multiple Reaction Sets based on how the Reactions are filtered.

The filters applied are related with how the Reversible reaction are considered and also how many measured metabolites involved.

For a Reaction to be considered, it should have at least one or more of the Metabolites in its equation measured. Hence we have 3 filters on Metabolome Composition:

- Filter-4: Reactions for which at least one metabolite measured
- Filter-5: Reactions for which at least one Substrate and one Product measured
- Filter-6: Reactions for which at least two metabolites measured

The Reversible reaction has "⇔" in the Equation.

When we treat Reversible reaction as 2 irreversible reaction. It will be split into 2 reactions, 1 Forward Only and 1 Backward Only. The Backward Only reaction will further be processed as a Forward Only reaction with the left and right side of its Equation reversed.

e.g. if we have A + B ⇔ C, we will split it as A + B => C and C => A + B

When we treat Reversible reaction as one bidirectional reaction, we will modify the equation to be the aggregation of the left and right hand side.

e.g. if we have A + B ⇔ C, we will look at it as aggregate of A + B ⇔ C AND C ⇔ A + B

Hence A + B + C ⇔ A C + A + B or A + B + C => C + A + B

- Filter-1: This filter excludes all reversible reactions
- Filter-2: It considers each reversible reaction as two irreversible reactions
- Filter-3: This considers each reversible reaction as one bidirectional reaction

Based on these filters, we generate multiple Reaction Sets (Total of 9 per Research Paper)

The WebUI generates data for this step without asking for input files from the users.

Feature Generation (Step 1 – Change Based)

For the different Reaction Sets created, we are going to generate the corresponding Change value for each Reaction. In this case,

Change = Sum of Change in Products – Sum of Change in Substrate

Notice that Reaction Set based on Filter-3 will have Change equal to 0.

Hence Reaction Sets 1, 2, 3, 4, 7, 8 have Change Feature.

The WebUI generates data for this step without asking for input files from the users.

Feature Generation (Step 2 – Ratio Based)

For the different Reaction Sets created, we are going to generate the corresponding Ratio value for each Reaction. In this case,

Ratio = Sum of Change in Products / Sum of Change in Substrate

Notice that only Reaction Set based on Filter-5 makes sense for Ratio-based.

Notice that Reaction Set based on Filter-3 will have Ratio equal to 1.

Hence Reaction Sets 2, 4 have Ratio Feature

The WebUI generates data for this step without asking for input files from the users.

Feature Generation (Step3 – Prob Based)

For the different Reaction Sets created, we are going to generate the corresponding Prob value for each Reaction. In this case, the Research Paper use the NetworkX package in Python to create a Network model for the Reactions.

In this network, we have these as Nodes: Study Samples, Metabolites (once as Product and once as Substrate), Reactions. The Edges between these Nodes are defined as per the equations in the paper.

PageRank algorithm is applied to the Network to generate the Probability for each Study Sample to reach the other Nodes.

All Reaction Sets can have Prob Feature.

The WebUI generates data for this step without asking for input files from the users.

Classification using RandomForest (This step takes a LONG time ~ 3 hours)

We have total of 17 different Features generated from Changes, Ratio and Prob to be used to predict the Food intake. We used RandomForestClassifier based on each of these Features to predict each type of Food Intake and generated Metrics such as Accuracy, Area Under Precision-Recall Curve Score (AU-PRC) and Area Under Receiver Operating Characteristic Curve Score (AU-ROC)

Summary

This aggregates the Metrics across the 17 RandomForestClassifiers for the Selected Food Studies and generate a .png whose content is displayed into the Web Output Canvas.

Task 4: WebUI Demo

WebUI Server using Flask

We use Flask to run the WebServer that manages execution of the Python Scripts on behalf of the Web Brower request.

webUIFlask.py is created that route the Request from WebUI index.html to execute the corresponding Python script.

We assume the following:

- input files are in a local directories called data
- python scripts are in a local directories called src

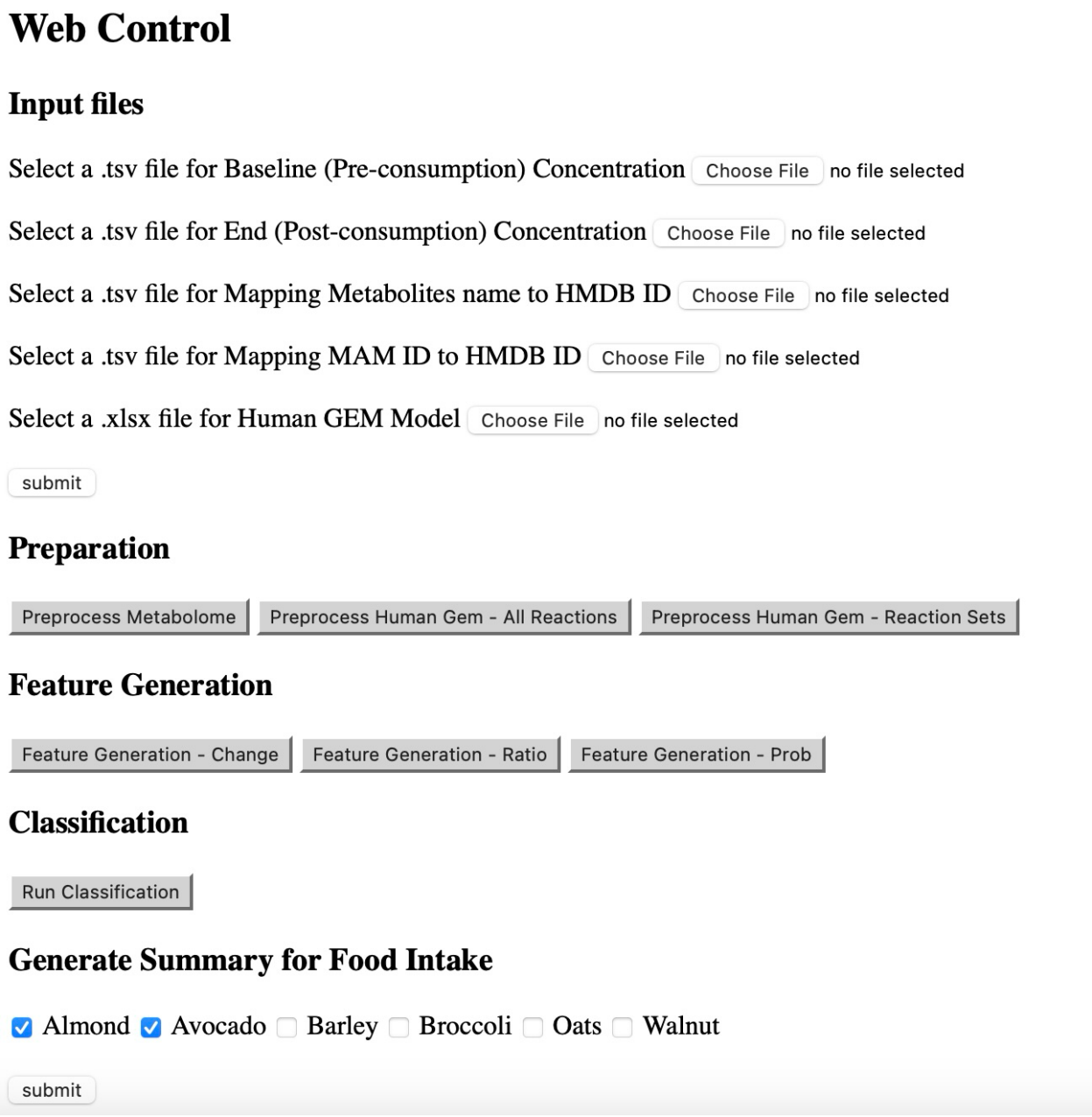
WebUI Client using index.html

Web Control

WebUI Client allows the 5 input files' name to be specified.

Each of the processing steps is enabled by its associated Button.

For Summary Generation, the Check Boxes allow user to specify which of the Food Intake's Performance to be shown in the Web Output section.



Web Output

The output of the Summary .png is displayed in the Canvas.

The snapshot below shows the output when only Almond, Barley and Walnut are selected.

Web Output

Performance Summary of Reaction-based Features

	Treatment	Almond			Barley			Walnut		
	Metric	Accuracy	AUROC	AUPRC	Accuracy	AUROC	AUPRC	Accuracy	AUROC	AUPRC
Reaction Set	Input Features									
Reaction-Set-1	Change	0.68	0.79	0.80	0.54	0.58	0.58	0.92	0.93	0.94
	Prob	0.00	0.00	0.33	0.18	0.04	0.33	0.00	0.00	0.33
Reaction-Set-2	Change	0.71	0.78	0.80	0.39	0.28	0.40	0.75	0.85	0.87
	Prob	0.00	0.00	0.33	0.00	0.00	0.35	0.11	0.03	0.33
	Ratio	0.39	0.32	0.41	0.57	0.57	0.61	0.72	0.77	0.75
Reaction-Set-3	Change	0.71	0.74	0.78	0.68	0.59	0.58	0.92	0.94	0.96
	Prob	0.18	0.06	0.34	0.04	0.02	0.33	0.19	0.09	0.33
Reaction-Set-4	Change	0.68	0.77	0.79	0.75	0.64	0.58	0.86	0.87	0.81
	Prob	0.00	0.00	0.34	0.04	0.00	0.33	0.06	0.04	0.33
	Ratio	0.50	0.43	0.46	0.57	0.61	0.55	0.67	0.75	0.71
Reaction-Set-5	Prob	0.04	0.00	0.32	0.11	0.05	0.33	0.19	0.08	0.33
Reaction-Set-6	Prob	0.00	0.00	0.33	0.00	0.00	0.33	0.00	0.00	0.32
Reaction-Set-7	Change	0.71	0.82	0.88	0.32	0.40	0.52	0.89	0.96	0.96
	Prob	0.07	0.02	0.33	0.07	0.01	0.33	0.19	0.06	0.33