Homework 2

Data Mining Technology for Business and Society

Deadline: **24 May 2021 23:59 (Rome Time Zone)**

Having **EXACTLY TWO** students per group is **RECOMMENDED**.

The total length of the report cannot exceed 6 pages.
It is forbidden to print or store this document, you can only read this document online.
It is forbidden to submit software written with Python-Notebook.
Only ".py" software is considered as a valid solution.

The software must be commented.

Data and software are available at:
http://www.diag.uniroma1.it/~fazzone/Teaching/Data_Mining_Technology_for_Business_and_Society_2020_2021/DMT4BaS_2020_2021.html

The homework is composed of two parts: "Recommendation-System" and "Team-Formation + Local Community Detection with PageRank".

Part 1

In this part of the homework, you have to improve the performance of two recommendation-systems by using non-trivial algorithms and also by performing the tuning of the hyper-parameters.

Part 1.1

Using the two datasets available in "DMT_2021__HW_2/Part_1/dataset/", you must apply all algorithms for recommendation made available by "Surprise" libraries, according to their default configuration.
WARNING: Ratings for "DMT_2021__HW_2/Part_1/dataset/ratings_1.csv" are integers in [1, 5], instead the ratings for "DMT_2021__HW_2/Part_1/dataset/ratings_2.csv" are integers in [1, 10].
For this part of the homework, and also for the next one, it is mandatory to use all CPU-cores available on your computer, by specifying the value in an explicit way with an integer number greater than 1.

Results for 1.1

For both provided datasets, you have to "copy-paste" in the final report all the "TABLES" in output from the execution of the "cross_validate" command on all algorithms: the number of folds to use is equal to 5.
Moreover, for both provided datasets, you must rank all recommendation algorithms you tested according to the MEAN_RMSE metric value: from the best to the worst algorithm.
Finally, you have to explain, by writing exactly one sentence, how you exploited all CPU-cores available on your machine.

Part 1.2

In this part of the homework, you have to improve the quality of both KNNBaseline and SVD algorithms, by performing hyper-parameters tuning always over **five-folds**. Even for this part of the homework, it is mandatory to use all CPU-cores available on your computer, and you have to use, again, the two datasets available in "/DMT_2021__HW_2/Part_1/dataset/".
For "DMT_2021__HW_2/Part_1/dataset/ratings_1.csv", only configurations with an average RMSE over all five folds **less than 0.89** will be accepted.
For "DMT_2021__HW_2/Part_1/dataset/ratings_2.csv", only configurations with an average RMSE over all five folds **less than 1.845** will be accepted.
In particular, you must perform a **Random**-Search-Cross-Validation process for tuning the hyper-parameter of the KNNBaseline algorithm. Instead, for tuning the hyper parameter of the SVD algorithm, you must use a **Grid**-Search-Cross-Validation approach.

Results for 1.2

By using **at most four** pages of the report, you must:
.) put in the report the complete "Grid-of-Parameters" you used to increase the performances for each method, for both "ratings_1.csv" and "ratings_2.csv".
.) put in the report the best configuration you found for each method, for both "ratings_1.csv" and "ratings_2.csv".
.) put in the report the two average-RMSE associated with the two best estimators you tuned, for both "ratings_1.csv" and "ratings_2.csv".
.) put in the report the total time required to select the best estimators, for both "ratings_1.csv" and "ratings_2.csv".
.) put in the report the number of CPU-cores you used.
.) put in the report, by writing exactly one line, an explanation on how you exploited all CPU-cores available on your machine.

Part 2

In this part of the homework, you have to implement a simple Team-Formation method based on Topic-Specific-PageRank and perform an analysis on the Team-Mate Pokemon Network. The dataset for this part (”DMT_2021__HW_2/Part_2/dataset/pkmn_graph_data.tsv”) is an unweighted and undirected graph where nodes represent Pokemon and an edge represents the fact that two Pokemon have a high level of affinity in battle.

Part 2.1

In this subpart of the homework, you have to use Topic-Specific-PageRank to assemble a team of exactly 6 different Pokemon starting from some already selected members in input. For each assigned starting set in input, you have to create a team by mining from the graph the remaining members with more synergy with the input starting set according to the Topic-Specific-PageRank score. You **MUST** consider as “Topic” the input set of the already selected team members. It is mandatory to use a damping factor of **0.33**.

For the following three input starting sets of Pokemon

```
Set_A = set(["Pikachu"])
Set_B = set(["Venusaur", "Charizard", "Blastoise"])
Set_C = set(["Excadrill", "Dracovish", "Whimsicott", "Milotic"])
```

, you must mine the best team of 6 Pokemon containing them using the Topic-Specific-PageRank procedure explained above. A team MUST be represented as a Python SET of Pokemon names of size 6.

To show that this procedure builds teams not simply by aggregating teams generated from individual nodes, you must perform the following experiment:

.1.) Create the best team using as input a single Pokemon: `"Charizard"`
.2.) Create the best team using as input a single Pokemon: `"Venusaur"`
.3.) Create the best team using as input a single Pokemon: `"Kingdra"`

.4.) Create the best team using as input a pair of Pokemon: `set(["Charizard", "Venusaur"])`
.5.) Create the best team using as input a pair of Pokemon: `set(["Charizard", "Kingdra"])`
.6.) Create the best team using as input a pair of Pokemon: `set(["Venusaur", "Kingdra"])`

.7.) Compute the number of team members inside the Team(Charizard, Venusaur) that are neither in Team(Charizard) nor in Team(Venusaur)
.8.) Compute the number of team members inside the Team(Charizard, Kingdra) that are neither in Team(Charizard) nor in Team(Kingdra)
.9.) Compute the number of team members inside the Team(Venusaur, Kingdra) that are neither in Team(Venusaur) nor in Team(Kingdra)
.10.) Write down all these results in the final report.

Results for 2.1

For the first part, you must provide the following:
.) The **set** of Pokemon provided in output by the Topic-Specific-PageRank procedure as the best team of 6 Pokemon using Set_A as input.
.) The **set** of Pokemon provided in output by the Topic-Specific-PageRank procedure as the best team of 6 Pokemon using Set_B as input.
.) The **set** of Pokemon provided in output by the Topic-Specific-PageRank procedure as the best team of 6 Pokemon using Set_C as input.

For the second part, provide the following:

.) The **set** of Pokemon provided in output by the Topic-Specific-PageRank procedure as the best team of 6 Pokemon using `"Charizard"` as input.
.) The **set** of Pokemon provided in output by the Topic-Specific-PageRank procedure as the best team of 6 Pokemon using `"Venusaur"` as input.
.) The **set** of Pokemon provided in output by the Topic-Specific-PageRank procedure as the best team of 6 Pokemon using `"Kingdra"` as input.
.) The **set** of Pokemon provided in output by the Topic-Specific-PageRank procedure as the best team of 6 Pokemon using `set(["Charizard", "Venusaur"])` as input.
.) The **set** of Pokemon provided in output by the Topic-Specific-PageRank procedure as the best team of 6 Pokemon using `set(["Charizard", "Kingdra"])` as input.
.) The **set** of Pokemon provided in output by the Topic-Specific-PageRank procedure as the best team of 6 Pokemon using `set(["Venusaur", "Kingdra"])` as input.
.) The **number** of team members inside the Team(Charizard, Venusaur) that are neither in Team(Charizard) nor in Team(Venusaur)
.) The **number** of team members inside the Team(Charizard, Kingdra) that are neither in Team(Charizard) nor in Team(Kingdra)
.) The **number** of team members inside the Team(Venusaur, Kingdra) that are neither in Team(Venusaur) nor in Team(Kingdra)

Part 2.2

In this part of the homework, it is requested to discover the social communities around Pokemon and also the Pokemon that are most and least present inside communities.
Interactions among Pokemon are collected inside the tsv files stored in the directory "DMT_2021__HW_2/Part_2/dataset/pkmn_graph_data.tsv". Each row in the tsv file represents the fact that two Pokemon are frequently members of the same team.
What is requested by the homework is to discover, for each Pokemon in the dataset, the local community centered on it. For discovering these local communities you must create an unweighted and undirected graph where nodes are Pokemon and where edges represent the interactions reported in the input tsv file: "DMT_2021__HW_2/Part_2/dataset/pkmn_graph_data.tsv".
The technique to use for discovering local communities must be the one explained in the lecture "Lab 3 part 2" of the course, but with the following change: instead of using a single fixed value for the PageRank damping factor, for finding a good local community, you have to try all the following values: [0.95, 0.9, 0.85, 0.8, 0.75, 0.7, 0.65, 0.6, 0.55, 0.5, 0.45, 0.4, 0.35, 0.3, 0.25, 0.2, 0.15, 0.1, 0.05].

It is clear now that, for finding a local community with a good conductance value for a given Pokemon inside the network, you **must** run the modified local community detection method for each of the possible 19 configurations given by all "PageRank damping factor" values.
WARNING: Communities with a conductance value of 0 or 1 are not considered as valid communities.
WARNING: Communities with more than 140 nodes (Pokemons) are not considered as valid communities.

It is important to remark that **it is not** requested to find a unique "PageRank damping factor" value that is good for all nodes (Pokemons), but, what is requested, is to find a good ad-hoc "PageRank damping factor" value for each single node (single Pokemon).

Once you mined for each Pokemon its local community, by using **at most one** page of the report, you have to report in a table the **five most frequent Pokemon** in local communities, and also the **five least frequent Pokemon** in local communities.
These two tables must be sorted in descending order of the number of local-communities a Pokemon belongs to… a.k.a. "*community frequency*".

Results for 2.2

By using **at most one** page of the report, you must report the two sorted tables containing the five most/least frequent Pokemon in local communities.
You must also submit a ".tsv" file containing a record for each Pokemon with the following fields: **pokemon_name**, **number_of_nodes_in_the_local_comunity**, **conductance_value_of_the_local_comunity**. The records in the file must be sorted in alphabetical order of the Pokemon name.

Where/What To Send

At the end of the process, you have to create a **zip** file with **ONLY** the following data:

1. The software for addressing Part_1: /DMT_2020/HW_2/part_1/sw/ (.py files).
2. The software for addressing Part_2: /DMT_2020/HW_2/part_2/sw/ (.py files).
3. The Part_2 tsv output file : /DMT_2020/HW_2/part_2/output.tsv (.tsv file).
4. The final report in **PDF**: /DMT_2020/HW_2/report.pdf .
5. **PLEASE, DO NOT PUT THE INPUT DATASETS IN THE ZIP FILE.**

The name of the zip file must have this format:

DMT_2021__HW_2__StudentID_StudentName_StudentSurname_StudentID_StudentName_StudentSurname.zip

Finally you must send the ".zip" file to fazzone@diag.uniroma1.it with the following email subject:

DMT_2021__HW_2__StudentID_StudentName_StudentSurname_StudentID_StudentName_StudentSurname.

p.s.

For any problem, doubt or consideration, please send a single email to both fazzone@diag.uniroma1.it and siciliano@diag.uniroma1.it.