**Rev 2 Algorithm**

**Algorithm** -It defines three mutually-recursive metrics - fairness of users, goodness of products & reliability of ratings. It uses both network & behavioural properties to output these metrics, also addresses cold start problem.

It works on bipartite weighted rating graph of users giving ratings to products. Rev2 calculates three intrinsic “quality” scores -

F(u) - Fairness score for each user user

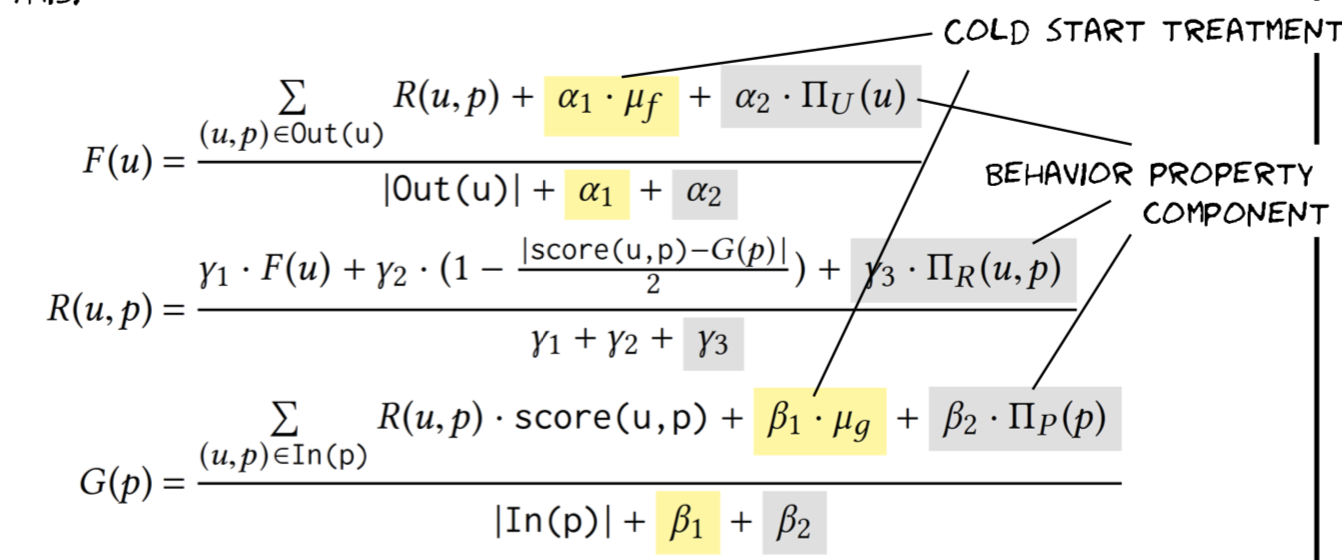
R(u,p) - Reliability score for each rating

G(p) - Goodness score for each product

**Reference Documents** - <https://cs.stanford.edu/~srijan/rev2/>

**Our Objective** - To generate fairness score for each score

**Rev 2 Formulation -**



**To generate network pkl file from network csv file**

Code - Graph.py

Input - network.csv file

Format is: source id, destination id, edge weight, timestamp

Output - network.pkl

Note - change path in code to generate for any other dataset.

**Unsupervised Scenario** - We are not having ground truth , whether the user is fake or genuine.

The algorithm is run for several combinations of α1,α2,β1,β2,γ1,γ2 and γ3 as in- puts, and the final scores of a user across all these runs are averaged to get the final Rev2 score of the user.

0 ≤ **α1,α2,β1, β2,γ1,γ2,γ3** ≤ 2

**ΠU , ΠR ,and ΠP** - behavioural score, using BIRDNEST score, but when behavior scores are not present (e.g., when behavior properties are unknown), then these scores are initialized to highest value 1.

Steps to run the code -

**1-** Download Rev2 folder

* Python 2 is required to run the code
* You may need to install required packages to import following libraries -

random, subprocess, numpy, time, collections, defaultdict, csv, networkx, datetime, math, unicodecsv, csv, detect, cPickle

**2 -** Go to command window/terminal & change directory to Rev2 (downloaded folder)

**3-** Run following command to generate user fairness score for different combination of

0 ≤ **α1,α2,β1, β2,γ1,γ2,γ3** ≤ 2

./run-rev2-all-params.sh amazon

This will run rev2code.py for all the combinations & output all files in “results” folder with user fairness score.

Note : “amazon” is network name passed as parameter in above command, it can be different based on your naming convention

Input - All files in “data’ folder

Output - All files in “results” folder

**4 -** Run following command to combine result from all the files

python evaluate-combined.py amazon

This will average all the score generated in Step 3 for each user, output file will be in “results-combined” folder. It will run “evaluate-combined.py” file.

Note : “amazon” is again the network name passed as parameter

Input - All files in “results” folder

Output - One file in “evaluate-combined.py” folder

**Note –**

We first initialize all scores using their respective behavior scores ΠU , ΠR ,and ΠP .

BIRDNEST algorithm is used to derive behavioural scores for users, product & edges(ratings).

If behavior scores are not present (e.g., when behavior properties are unknown), then these scores are initialized to highest value 1.