The diagram of my pipeline description:

Step 1: Data Processing

Processing the input data from the file "soc-LiveJournal1Adj.txt".

Output format:
[
    (3, [0,12,41,55,1532,12636,13185,27552,38737]),
    ...
]

Step 2: Generate relationship pairs (U, (X, Y))

Generate a defined relation pair (U, (X, Y)) for a given user id(U).
    (U, (X, Y)): Here user U and user X are mutual friends, user X and user Y are mutual friends, but user U and user Y are not mutual friends.

Output format:
    For example (1, (0, 2)) means user 1 and user 0 are mutual friends, user 2 and user 0 are mutual friends, but user 1 and user 2 are not mutual friends.

Step 3: Get recommends N = 10 users using pairs (U, (X, Y))

Aggregate the count of mutual friends for each user_id

Output format:
[
    (user_id, recommendation]), number of mutual friends),
    ...
]

Step 4: Fetch final results

Write top_recommendations results to "output.txt"
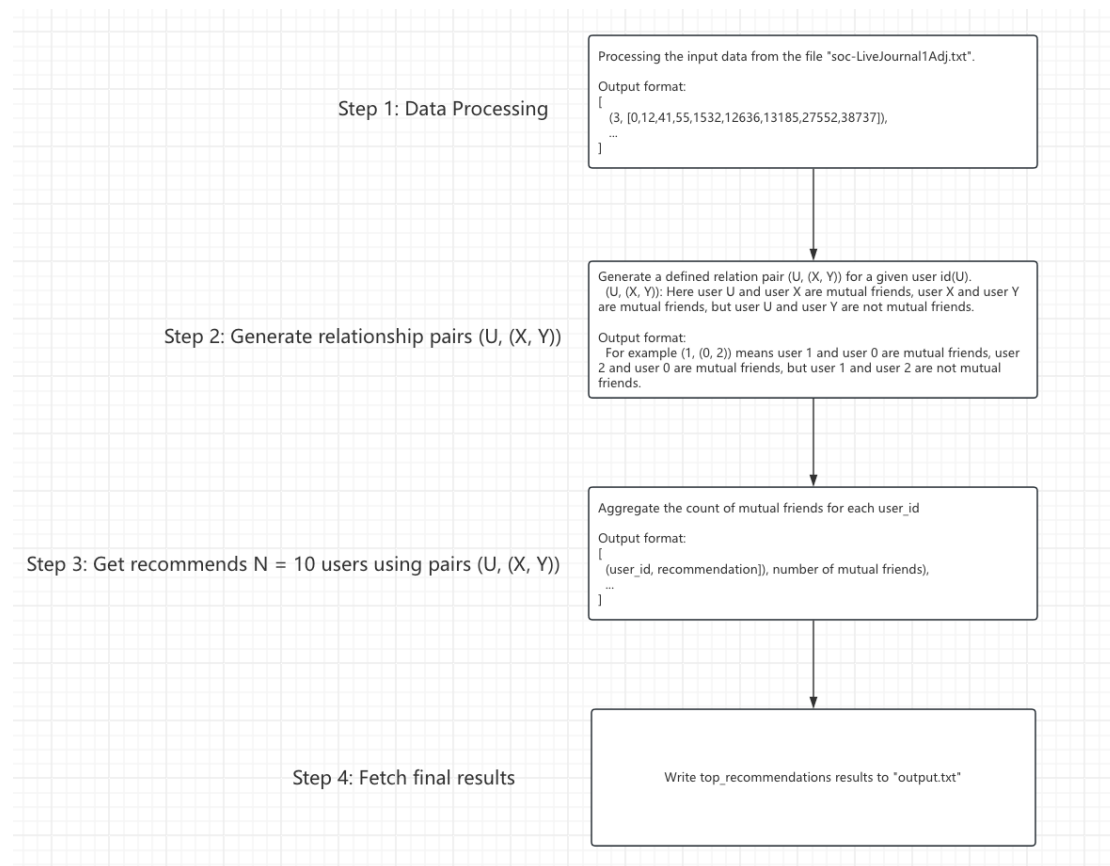
The recommendations for the users with following user IDs

924: [439, 2409, 6995, 11860, 15416, 43748, 45881]

8941: [8943, 8944, 8940]

8942: [8939, 8940, 8943, 8944]

9019: [9022, 317, 9023]

9020: [9021, 9016, 9017, 9022, 317, 9023]

9021: [9020, 9016, 9017, 9022, 317, 9023]

9022: [9019, 9020, 9021, 317, 9016, 9017, 9023]

9990: [13134, 13478, 13877, 34299, 34485, 34642, 37941]

9992: [9987, 9989, 35667, 9991]

9993: [9991, 13134, 13478, 13877, 34299, 34485, 34642, 37941]