Athens University of Economics and Business

MSc in Business Analytics

Data Mining – Assignment 1

Deadline: 28/5/2023

Group assignment (groups of up to 2 people).

The assignment corresponds to 25% of the total grade of the course.

Discussions between groups are recommended but collaborating on the actual solutions is considered cheating and will be reported.

There will be no extension of the assignment deadline!

Professor: Y. Kotidis (kotidis@aueb.gr)

Assistant responsible for this assignment: I. Filippidou (filippidoui@aueb.gr)

In this assignment you will explore the use of Jaccard distance, min hashing, and LSH in the context of user similarity in a movie rating dataset. To fulfill this assignment, you will have to perform the following tasks:

1) Import and pre-process the dataset with users

Download the movieLens dataset from moodle. This dataset includes 100.000 ratings (1-5) from 943 users on 1682 movies. Each user has rated at least 20 movies. There are 3 files for the dataset, the users.txt file contains id, age, gender, occupation, and postcode separated by |, the movies.txt file contains id, title (with release year) and some other information not related with the assignment separated by |, the ratings.txt file (tab separated) which contains userid, movieid, rating (1-5) and timestamp. For this assignment you will only use the set of movies that a user has rated and not the ratings. In your report you should describe in detail any processing and conversion you made to the original data and the reasons it was necessary.

2) Compute exact Jaccard similarity of users

To assess the similarity between users you should compute the exact Jaccard Similarity for all pairs of users and only output the pairs of users (unique) that have similarity at least 0,5 (>=50%). For each pair denote their ids and the similarity score.

Also output the movie titles that the most similar pair of users has seen.

3) Compute similarity using Min-hash signatures

In this step you compute min-hash signatures for each user and use them to evaluate their similarity.

Description of hash functions: use the following family of hash functions: $h_{a,b}(x)=(ax+b) \bmod R$, with a,b random integers in the interval (0,R) and R a large enough prime number that you may want to finetune in your initial experimentation. Make sure that each hash function uses different values of a,b pairs.

Evaluation of Min-hashing: Use 50, 100, and 200 hash functions. For each value, output the pair of users that have estimated similarity at least 0,5, and report the number of false positives and false negatives (against the exact Jaccard similarity) that you obtain. For the false positives and negatives, report the averages for 5 different runs using different functions. Comment on how the number of hash functions affects the false positive and negatives figures.

4) Locate similar users using LSH index

Using a set of 200 hash functions break up the signatures into b bands with r hash functions per band (b*r=200) and implement Locality Sensitive Hashing.

Recall that with LSH we first locate users that are similar (have the same mini-signatures) across at least one band and then assess their true similarity using their initial representations. Use the following two instances of LSH:

- LSH instance 1: b = 25, r = 8
- LSH instance 2: b = 40, r = 5

Using each instance find the pair of users with similarity at least 0.5 and report:

- The number of true pairs returned (true positives).
- The number of similarity evaluations performed using the initial representations.

Report the averages for 5 different runs using different functions.

Based on the reported results, what do we gain/loose by using LSH instead of directly comparing users on their true representations?

Assignment handout:

You should turn in your code, the input file that you used, and the output files that you produced for the different runs. Also turn in a file with the average numbers (tasks 3,4), and a report with your observations.