

Dear Reviewers,

thank you for your valuable feedback. Please find below our responses to the reviews:

- Related work was moved to Section 2.

Reviewer 2:

Weak points:

W1: Please notice that one of our goals is to make it simple to derive existent (or new) recommendation algorithms through our framework. For each different algorithm, the adaptation is as simple as changing the edge weights. In all, the algorithms without the framework would perform the same as with the framework in terms of precision and recall, but the effort to implement each algorithm separately (without our framework) would be much greater.

W2: We disagree with the reviewer. Please notice that the algorithms were evaluated following a holdout methodology (i.e., in our case 5 random splits of training and test datasets were generated), which is a evaluation protocol widely used in the machine learning/recommender systems community.

W3: Please notice that in section 2 we describe the recommendation scenario formally and in section 3.1 we give an example illustrating this scenario in terms of our relational approach. Moreover, the ultimate goal is to deploy this framework in real world recommender systems.

Concerning the question on whether the cold-start problem was solved, we believe, based on the results, that the problem was alleviated. Please recall that edges containing cold-start users are weighted differently based on the preferences expressed in higher levels of the hierarchy of regions considered, while non-cold start users are weighted with the in-context cosine similarity (preferences within the city of interest). If we have used only the flat cosine similarity, the algorithm would not have being able to compute any recommendations for the cold-start users at all since their preference vectors would be empty. The possibility of weighting each case differently in the relational graph, enabled us to make recommendations for cold-start and non cold-start users using the same recommendation algorithm.

The question on why to use cosine was also addressed.