Paper Assignment for Project 4

In this project, you need to perform different collaborative filtering algorithms on two data sets.

1. Data Set

The first data set is Anonymous Microsoft Web Data (https://archive.ics.uci.edu/ml/datasets/Anonymous+Microsoft+Web+Data). It's an example of implicit voting data, with each vroot characterized as being visited (vote of one) or not (no vote).

The second data set is EachMovie (http://www.gatsby.ucl.ac.uk/~chuwei/data/EachMovie/eachmovie.html). This is an explicit voting example using data, with votes ranging in value from 0 to 5.

The two data sets are both evaluated in Paper 1. For the first one, the train-test split was already done by the author. But for the second one, you need to randomly select 20% from each user's ratings as test set.

2. Implementation

Working in teams, you need to evaluate and compare a pair of algorithms for collaborative filtering (CF). The assignment details can be seen in the Table.

The first category you need to implement is memory-based algorithms, and the framework you will use is from Paper 2. Each team is assigned with 4 different similarity measure, including SimRank (Paper 3), and other different combinations of prediction algorithm components.

The second category of CF techniques is model-based algorithms. You are only required to perform the cluster model introduced in Paper 1. And for simplicity, you are allowed to choose the number of classes by cross-validation, instead of marginal likelihood approximation.

3. Evaluation

You need to compare the performance for these different algorithms and component combinations. For the first data set, you can use ranked scoring for evaluation, as introduced in Paper 1. And for the second data set, you can use mean absolute error (MAE) and ROC sensitivity, as introduced in Paper 2. Also, other different evaluation criteria are encouraged.

* Assignment Table

Algorith	Component	Variants	grp1	grp2	grp3	grp4	grp5	grp6	grp7	grp8	grp9
Memory-based Algorithm	Similarity Weight	Pearson Correlation	1, 2			1, 2	1, 2		1, 2		1, 2
		Spearman Correlation	1, 2	1, 2			1, 2	1, 2		1, 2	
		Vector Similarity	1, 2	1, 2	1, 2			1, 2	1, 2		1, 2
		Entropy		1, 2	1, 2	1, 2			1, 2	1, 2	
		Mean-square-difference			1, 2	1, 2	1, 2	1, 2		1, 2	1, 2
		SimRank*	2	1	2	1	2	1	2	1	2
	Significance Weighting	No	1, 2	1, 2	1, 2	1, 2	1, 2	1, 2	1, 2	1, 2	1, 2
		Yes	1, 2			1, 2	1, 2		1, 2	1, 2	
	Variance Weighting	No	1, 2	1, 2	1, 2	1, 2	1, 2	1, 2	1, 2	1, 2	1, 2
		Yes	1, 2	1, 2			1, 2	1, 2		1, 2	1, 2
	Selecting Neighbours	Weight Threshold	1, 2	1, 2	1, 2	1, 2		1, 2			
		Best-n-estimator		1, 2	1, 2	1, 2	1, 2	1, 2	1, 2		1, 2
		Combined		1, 2	1, 2	1, 2		1, 2		1, 2	
	Rating Normalization	Deviation for Mean	1, 2	1, 2	1, 2	1, 2			1, 2		1, 2
		Z-score			1, 2		1, 2	1, 2	1, 2	1, 2	1, 2
Model-based Algorithm	Cluster Models		1	2	1	2	1	2	1	2	1

Comments:

1. Each table cell shows which data set you need to implement on for this variant, where “1, 2” means you need to implement both.
2. When comparing performance of different variants for a component, e.g. Similarity Weight, you can just choose one variant for each other component, instead of trying all possible combinations.