Item-Based and Advanced Collaborative Filtering Topics Quiz

测验, 10 个问题

1 point

1。
Why is item-item more amenable to pre-computation than user-user?

- (●) When there are many more users than items, item similarities are fairly stable, while user similarities can change rapidly as the user interacts with the system.
- ( ) Because items don't really have correlations; all the correlations are made through the users.
- ( ) When there are many fewer items than users that means there are many fewer correlations to pre-compute.
- ( ) Because item-item tends to exhibit lower serendipity, and therefore less popular items don't matter much.

---

1 point

2。
The item-item model is often truncated to only keep M neighbors per item. The scorer uses at most k neighbors to compute each prediction. Why must M be significantly larger than k?

- (●) Because the user we're predicting for may not have rated all M of the neighbor items; we need enough potential neighbors to be able to find k of them among the user's ratings.
- ( ) Because some of the M neighbors we pick will already have been rated by the target user, and we need to find k unrated ones to recommend to that user.

- ○ Because the M neighbors we keep may not be good ones, so we need to find the top-k best neighbors from among those M.
- ○ Because item-item collaborative filtering depends on reduction of the matrix rank to rank-k as a mechanism for smoothing out artifacts in the data set.

---

1 point

3。
Massa and Avesani's trust-aware recommender multiplies user vectors by their trust weight prior to doing item-item collaborative filtering. What does this accomplish?

- ○ It decreases the undue influence of users who rate many items.
- ● It makes high-trust users' ratings more influential in computing item similarities.
- ○ It removes low-trust users from the system.
- ○ It adjusts for the fact that the user vectors were originally divided by trust weight to restore all ratings to equal weight.

---

1 point

4。
When would it not make sense to use item-item collaborative filtering (compared with user-user)?

- ○ When different users have very different tastes.
- ○ When there are lots of similar items.
- ○ When you're more concerned about prediction than about recommendation.

When there are many more items than users.

---

1 point

5。
Which of these best explains how we obtain personalized predictions for a target item i, and a target user u using item-item collaborative filtering?

- (●) We identify which of the close item-neighbors to i have been rated by u, and we compute a weighted average of those ratings, weighted by how similar the items are to i.
- ( ) We compute the normalized rating for item i by subtracting from each rating the mean rating given by that user. Then we personalize that normalized rating into a prediction for user u by adding it to u's mean rating.
- ( ) We average all of the ratings given to item i, but using a weighted average where the weight of each rating is based on the similarity between the user who gave that rating and the target user u.
- ( ) We identify a set of other users who've rated item i, and use item-overlap to determine which of those are closes to target user u. Then we average their ratings for item i, usually using normalized ratings.

---

1 point

6。
MusicFX picked radio stations to play in a shared company gym. Users rated each station (genre). One early feature of the system was that it would avoid playing any station that any single person in the gym marked as hated. What was the problem they experienced that led them to change this feature?

- ( )

They discovered that people didn't want to be seen as imposing their tastes, so they were reluctant to admit that they hated specific station.

- ○ The feature worked well, but they had to discontinue using it because they had a contract that required them to play a wider variety of music.
- ● They discovered that some people would mark a station as "hated" even if they just mildly disliked it, and sometimes just to force the system to change the station.
- ○ They quickly discovered that too often everything was hated by someone, and couldn't find any stations to play.

---

1 point

7。
The Herlocker explanations paper explored a variety of explanation interfaces, but it did have one key mistake. What was that mistake?

- ○ The authors didn't realize that some of the explanations were really just made up data.
- ○ It forgot to use some of the better explanations available.
- ○ The authors didn't recruit enough test users to get any statistically significant results.
- ● The experiment really didn't measure usefulness of explanations; it measured persuasiveness of those explanations instead.

---

1 point

8。

In the mid 1990s, Net Perceptions was struggling to meet performance goals with its user-user collaborative filtering algorithm. What was its solution to this problem?

- ○ Switching from user-user collaborative filtering to a matrix-factorization algorithm.
- ○ Mining through user profiles to find clusters of similar users and combining their records.
- ● Switching from user-user collaborative filtering to an item-item algorithm.
- ○ Limit the number of users handled to no more than 250,000 at a time.

---

1 point

9。
When using item-item CF with unary data, we usually just sum the similarities between the item and its neighbors, rather than computing a weighted average. Why?

- ○ Because summation is significantly faster computationally than computing a weighted average, and a major benefit of item-item is faster performance.
- ● Since there are no ratings, the weighted average is effectively an average of a set of 1s, which is always 1. Summing similarities creates a meaningful score.
- ○ Because sums help adjust for the fact that we don’t really know whether the non-ratings represent items that are disliked or just never consumed.
- ○ Because weighted averages cannot be pre-computed, but sums can be easily cached and reused for future computations.

Item-Based and Advanced Collaborative Filtering Topics Quiz

测验, 10 个问题

1
point

10。
Why might we want to intentionally add noise to ratings or user profiles?

- (●) To protect user privacy from the service provider (or provide deniability of preferences).
- ( ) To improve item-item recommender performance by smoothing over artifacts of individual ratings.
- ( ) To help users discover new items they wouldn’t have discovered with their own ratings.
- ( ) To support a switch from correlation-based to Bayesian probability models.

☑ 我（**伟臣 沈**）了解提交不是我自己完成的作业 将永远不会通过此课程或导致我的 Coursera 帐号被关闭。
了解荣誉准则的更多信息

提交测试