Remix Networks in Scratch

Yue Han, Jeffrey V. Nickerson {yhan4, jnickerson}@stevens.edu
Stevens Institute of Technology Howe School of Technology Management

Explorations

The remix tree is defined as a tree formed by a project and all of its associated remixes. The nodes of the trees are projects, and each project has associated with it a user and a creation time, as well as parent and child links.

The following shows four typical remixing tree patterns on Scratch. In these graphs, the blue node represents the original project and the red nodes represent all remixes.

In many trees, each user only contributes once. But in some trees, some users contribute multiple times. In these trees, reciprocity can occur. Amy modifies Beth’s design and then Beth modifies Amy’s design, and so on. Thus, we can classify the remix trees into three types according to their user networks: uniform, weighted, and reciprocal. In uniform trees, each user contributes exactly once. In weighted trees, some users contribute more. In reciprocal trees, pairs of users alternate in modifying each other’s work.

The graph on the right is the third type of remix graph: both the original project and one of the remixes have generated many remixes.

The last graph is the fourth type of remix tree, in which a chain of remixes occur.

In many trees, each user only contributes once. But in some trees, some users contribute multiple times. In these trees, reciprocity can occur. Amy modifies Beth’s design and then Beth modifies Amy’s design, and so on. Thus, we can classify the remix trees into three types according to their user networks: uniform, weighted, and reciprocal. In uniform trees, each user contributes exactly once. In weighted trees, some users contribute more. In reciprocal trees, pairs of users alternate in modifying each other’s work.

The above three graphs are different in their remix depth. However, they are similar, in that most of the remixes are created based on the original project.

Project popularity can be measured in three different ways. Users can click the love button on the projects they admire. Or they can add a project to their favorite projects gallery. Or they can remix the project. The three different types of trees exhibit significantly different degrees of popularity, with the reciprocal trees by far the most popular.

The study is advised by Dr. Jeffrey V. Nickerson from Stevens Institute of Technology. This material is based upon work supported by the National Science Foundation under Grants IIS- 0968561 and IIS-1211084.

We propose a recommender system for the Scratch online community. The proposed recommender system utilizes project tag information to determine similarities between various users and then uses these relationships to identify the optimal set of items to be recommended. Through a calculated combination of relevancy and diversity (LSI and MMR), the recommender system is aimed at leading users to explore further into the Scratch community and improving the productivity of “passive producers” by using the output of “active consumers”.

Our aim is twofold:
- Create and evaluate recommendations based on two different types of input tags: user shared tags and user favorite tags.
- Evaluate recommendations based on relevancy and diversity.

The algorithm:
- Using latent semantic indexing (LSI), we constructed a collaborative view of user profiles and created user-tag and user-user similarity matrices in a reduced dimensional space.
- Using MMR algorithm, MMR(Q,Di) = λSim(Q, Di)−(1−λ) max Sim(Dj, Di), we iteratively selected the users with highest similarity to our query and then updated the remaining user similarity scores by computing the degree of dissimilarity between each user and the previously selected ones.

Controlled Studies

To test the quality of the final recommendations, a set of HITs were posted on Amazon Mechanical Turk. Each HIT was opened to 20 Amazon Mechanical Turk workers. Workers were presented with the project tags of our users and recommended users’ project tags. They were asked to review the recommendations and indicate their likelihood of viewing recommended projects.

- C1. Evaluating the shared tags of randomly selected users and λ values of 0, 0.5 and 1.
- C2. Evaluating the favorite tags.
- C3. Randomly selected users from user pool. 10 were assigned to the shared tags group and 10 were assigned to the favorite tags group. Turkers were asked to evaluate these users similar to conditions 1 & 2.

Results

Our initial experimental results show that recommendations that balance the notions of relevancy and diversity (λ = 0.5) perform better.

Our results also show that using favorite tags instead of shared tags may result in better recommendations. The act of “favoriting” might be a better indication of user interest and show a clearer direction to what future projects users would like to create next. In that sense, a favorites list could be used as a predictor and a repository for future project ideas.

Notes and Credits

- The study is conducted using de-identified data from Scratch, a project of the Lifelong Kindergarten Group at the MIT Media Lab.
- The study is advised by Dr. Jeffrey V. Nickerson from Stevens Institute of Technology.
- This material is based upon work supported by the National Science Foundation under Grants IIS- 0968561 and IIS-1211084.