



Trinity College Dublin
Coláiste na Tríonóide, Baile Átha Cliath
The University of Dublin

MSc COMPUTER SCIENCE – DATA SCIENCE – RESEARCH PROPOSAL

A Recommender system for Scratch TEL platform

Roman Shaikh

18300989

shaikhr@tcd.ie

Project Proposal submitted in partial fulfillment

Of the requirements for the module of

Research Methods (CS7CS1)

Author Declaration

Student Name: Roman Shaikh

Student Number: 18300989

Email: shaikhr@tcd.ie

Proposed supervisor: Glenn Strong

Assignment Number: 1

Module Number: CS7CS1

Title of Assignment: Research Proposal for A Recommender system for Scratch TEL platform.

Word Count: 2098

I have read, and I understand the plagiarism provisions in the General Regulations of the University Calendar for the current year, found at <http://www.tcd.ie/calendar>

I have also completed the Online Tutorial on avoiding plagiarism 'Ready, Steady, Write', located at <http://tcd-ie.libguides.com/plagiarism/ready-steady-write>

I declare that this assignment, together with any supporting artifact is offered for assessment as my original and unaided work, except in so far as any advice and/or assistance from any other named person in preparing it and any reference material used are duly and appropriately acknowledged.

Signed:

Date:

Contents

1.	Introduction	4
2.	Research Question	4
3.	Research aims	4
4.	Motivation.....	5
5.	Potential benefits of the research	5
6.	Literature search strategy	5
7.	Sources of literature	6
8.	Proposed methodology.....	6
9.	Research Dissemination Plan	6
10.	Description of the research area	7
11.	Evaluation criteria and method	7
12.	Ethics	7
13.	Research Milestones	7
14.	Proposed Table of contents	8
15.	Skills audit	8
16.	Anticipated problem	8
17.	References	8

1. Introduction

As computers and programs are becoming an important part of our daily lives, the demand for programmers and people who develop and improve software systems is growing even more. For this reason, more and more people are taking interest in learning how to write code. People normally turn towards online and open programming courses. This has given popularity to a more informal mode of education, in which a student learns and develops without a mentor/teacher. This type of self-paced and self-directed learning is one of the most important parts of Technology Enhanced Learning (TEL) model which covers technology in all forms of teaching and learning activities [10]. TEL platforms which support self-directed learning are a useful way for students to practice and master essential concepts and techniques presented in introductory programming courses. Although these systems are used widely across the academic landscape, we have a limited understanding of how a user uses these systems and what can be learned from the data collected [7].

Scratch is one such TEL platform developed by Lifelong Kindergarten Group at the MIT Media Labs. Scratch helps young and inexperienced students to develop programming skills and think creatively. With Scratch, one can program interactive stories, games, and animation. It also provides a collaborative platform through which students can share their own code and develop on others work. Scratch is also being used in more formal education by teachers to introduce programming concepts to inexperienced users [11]. It is often been observed that some users get demotivated easily because of either they are unsure of where to go further or the programming exercises are not up to their individual expectations [12]. Thus, the concern arises how do we keep students motivated in Scratch and improve the learning experience.

An effective solution is to recommend students with exercises according to their level of knowledge and previous experience. This intuitively is known as a Recommender Systems (RSs), a system that recommends users with exercise based upon their previous activities. Recommender system in an educational environment is proven to be significantly beneficial ((e.g., reviewed in [13] and [14])).

Based on my initial survey there has been very limited research done on Recommender systems in Scratch. In this research, I propose to build an effective Recommender system which improves the learning experience of Scratch users. Based on the assessment of the previous exercise in scratch and finding correlations the system would recommend a set of exercise that student could follow to complete his goals.

2. Research Question

Can we improve the learning experience of users on Scratch by recommending them exercises to follow based on their Scratch history?

3. Research aims

The research aims to improve the learning experience of Scratch users by suggesting and recommending them the exercises and projects. Most users who start learning with Scratch typically follow different

areas of interest like Storytelling, animations, and games. Each set of these areas require different types of skills which a user develops by playing around.

Since in Scratch the student usually does not have any mentor to guide, some students simply lose interest because they are not able to follow through the entire learning exercises. This research intends to address this issue by building a recommender system that could guide a student to follow a learning exercise based on his history and area of choice.

4. Motivation

There are many platforms that help novice programmers learn to code in various programming languages. Scratch developed at MIT in 2007 is one such platform mainly targeted towards young people and building their appetite to code. It also provides a collaborative platform to share, mix and develop code in the community. Scratch has seen a major growth in its user base over time. Massive Open Online Courses (MOOC) platforms like Coursera and Edx are most important "novelty" in the field of e-learning in this decade [1]. These platforms offer their own recommendation to users for the next best course available.

Scratch encourages the sharing, reuse, and combination of code, as indicated by their slogan, "Imagine, Program, Share" [15]. Users can make their own projects, or they may choose to "remix" someone else's project. It is part of a research to design new technologies to enhance learning in after-school centers and other informal education settings and broaden opportunities for youth who can possibly become designers and inventors.

Scratch was popular in the United Kingdom through Code Clubs. These code clubs usually consist of volunteering mentors who guide young people with the learning process. One of the major tasks that this helps the student with is suggesting them new exercises to follow. Most TEL platforms have a system in place wherein they suggest users with appropriate courses and exercises. However, Scratch has limited functionality when it comes to recommending exercise and projects. This motivated us to build a recommender system which would help budding programmers in their learning journey.

5. Potential benefits of the research

This research will help in building a recommender system for Scratch programming environment which will help young and novice Scratch users to better build up their skills and develop future applications. The research can also be used as a basis for understanding programming patterns of the user and how they develop over time.

6. Literature search strategy

The literature search strategy consists of finding and studying existing research papers, tractions, and conferences on the recommender system for MOOC platforms. Various research publications like *jstor*, *IEEE* explore will be used for finding appropriate material. Information and resources found using Google scholars search engine will also be used in studying the existing work on Scratch data analysis. Apart from this various source like blog posts and training material for studying methods to build a

recommender system will be used. Also, papers related to identifying managing patterns of human behavior to recommender system will be useful

All these resources will be referenced and used to better understand the algorithms and methods used to design recommender systems in on TEL platforms.

7. Sources of literature

IEEE Explorer, JSTOR digital library digital library, TARA, GitHub, Kaggle blogs.

8. Proposed methodology

The methodology I propose to use is a Build form of methodology [9], as the research focuses on building a recommender system.

Building a good recommender system requires both a good quality and quantity of data to study. For this to be successful we propose the following methodology.

Planning a modular approach will be used while designing the system. Small interfaces will help in further testing of the software system.

A suitable language would be the one which provides the most flexibility, support, and packages which support data analysis and machine training. Based on our findings either Python or Java would be best suited for this task.

A dataset will be scraped from scratch API (or the publicly available one will be used) which contains information like *user code samples, the kind of project user is most interacting with, the projects a user follows and builds upon* etc. We hope this will satisfy most of our requirements.

Based on the dataset, a pool of exercise will be identified that most users follow. This can be either concept based (like the sequence in code, iterations, loops, Boolean logics etc.) or a pre-defined set of exercise created by instructors. This is yet to be decided.

Different types of recommendation systems like content-based and collaborative filtering will be chosen to best fit the dataset/model we are trying to build.

The recommender system would then be trained based on the training dataset and the results will be tested on the test datasets.

9. Research Dissemination Plan

The dissemination will take place using different venues like conferences, blogs, and journals. All this will be identified based on the relevant areas of the research topic e.g. education, data analytics, and a recommender system.

10. Description of the research area

The area which is most related to this research is education and online education. With the increasing popularity of online and web-based programming courses, it also becomes essential to guide the learners in the right direction to develop the right skills. This research will help educators and learners who use Web-based programming exercises in Scratch with tools which can be beneficial to them. This research can be further expanded to other online learning platforms.

11.Evaluation criteria and method

The evaluation criteria and methods roughly will include validating the recommendation system against test dataset. A good recommender system is characterized by how much the user has learned by following through the system recommended exercises [14] [5]. A feedback mechanism will be in place to take the user's feedback on the completed exercise and their usefulness. This will give us a sense of how and how much the users' knowledge has improved.

12.Ethics

The dataset used may contain user sensitive information like username and project IDs. Although this may indicate a requirement for a formal ethics approval since all the dataset is made available by MIT in public it will not be a problem. Moreover, the user sensitive information will not be used in any form in this research. If at all any user-related information is deemed to be required, an ethics approval will be applied for before using it.

13. Research Milestones

The research milestones will tentatively follow below timelines. This could change depending on the ongoing work and accomplished task.

[illegible]

14. Proposed Table of contents

The proposed table of contents is as follows:

1. Abstract
2. Introduction
 - 2.1 Motivation
 - 2.2 Objectives
3. Literature Review and Previous work
4. Description of Dataset
5. Design
6. Implementation
7. Evaluation
8. Insights
9. Conclusion

15. Skills audit

This research deals with building a recommender system, in a traditional sense which requires a good knowledge of data analytics and machine learning. As a part of my module, I have studied and familiarized myself with machine learning and data analytics. However, an in-depth knowledge of recommender systems and their working is required for this study. With this research, I hope to develop a complete understanding of recommender systems. Also, a good understanding of the education sector and TEL platforms.

I feel my knowledge and experience in data warehousing and mining would help me with the preliminary task of the research.

16. Anticipated problem

Various anticipated problems can arise while pursuing this research, such stated below:

1. The dataset obtained could lack the necessary information required for building a reliable system.
2. Cold Start Problem - This problem occurs when new users enter the system and no information is available.

17. References

1. Jaime Spacco, Paul Denny, Brad Richards, David Babcock, Robert Duvall: Analyzing Student Work Patterns Using Programming Exercise Data
2. J. Bennedsen and M. E. Caspersen. Failure rates in introductory programming. SIGCSE Bull
3. Efthimia Aivaloglou, Felienne Hermans, Jesús Moreno-León, Gregorio Robles, A dataset of scratch programs: scraped, shaped and scored, Proceedings of the 14th International Conference on Mining Software Repositories, May 20-28, 2017, Buenos Aires, Argentina
4. Deborah A. Fields, Yasmin B. Kafai, Michael T. Giang, Youth Computational Participation in the Wild: Understanding Experience and Equity in Participating and Programming in the Online

Scratch Community, ACM Transactions on Computing Education (TOCE), v.17 n.3, p.1-22, August 2017

5. Leonel Morales Diaz, Luis Felipe Ayala Lopez, A Classification of Programming Styles in Scratch, Proceedings of the 8th Latin American Conference on Human-Computer Interaction Article No. 9 Antigua Guatemala, Guatemala.
6. Filiz KALELIOĞLU , Yasemin GÜLBAHAR : The Effects of Teaching Programming via Scratch on Problem Solving Skills: A Discussion from Learners' Perspective
7. Fatiha Bousbahia, Henda Chorfia : MOOC-Rec: A Case Based Recommender System for MOOCs
8. Jesennia Cardenas, Pavel Novoa-Hernández, Amilkar Puris and David Benavides Recommending Exercises in Scratch: An Integrated Approach for Enhancing the Learning of Computer Programming
9. Jos´e Nelson Amaral: About Computing Science Research Methodology
10. Manouselis, N., Drachsler, H., Vuorikari, R., Hummel, H., & Koper, R. (2011). Recommender systems in technology enhanced learning.
11. Malan DJ, Leitner HH (2007) Scratch for budding computer scientists.
12. Tanrikulu E, Schaefer BC (2011) The users who touched the ceiling of scratch.
13. Manouselis N, Drachsler H, Verbert K, Duval E (2013) Survey and Analysis of TEL Recommender Systems, S
14. Klasnja-Mili ´cevi ´c A, Ivanovi ´c M, Nanopoulos A (2015) Recommender systems in ´e-learning environments: a survey of the state-of-the-art and possible extensions.
15. "Scratch - Imagine, Program, Share". scratch.mit.edu.