Capstone Project Overview | Coursera

Project Overview

In this capstone project you will explore social network data, completing a project where you work with graphs and other data structures you've learned in previous courses in this specialization, while applying at least one new algorithm to solve a problem of your choosing.

Social network data is all around us, and is easily represented as a graph. For example, if you think about your friend network on Facebook, each person can be represented as a node in the graph, and each friendship connection can be represented as an undirected edge between the nodes containing the people involved in the friendship. Twitter can be represented similarly, but with directed edges since following someone is a one-way relationship (you can follow someone without them following you back).

Using this graph abstraction, we can answer many interesting questions about the data underlying the graph. For example, we can automatically discover communities in a social network; we can recommend that people connect as friends based on the structure of their mutual acquaintances in the graph; we can figure out who the "influencers" are; and, we can look at how information might travel within and across groups.

In this capstone project, you will explore some of these issues. Which questions you answer and which properties you consider will largely be up to you: we're looking for creativity and personal expression. We'll provide you with some support, but where you go with it is up to you and we look forward to seeing what you uncover in the world of social network data!

Project Components and Checkpoints

Throughout this course we have both graded and ungraded assignments to help guide you through your project. We recommend that you complete all of these assignments (and if you do you will certainly stay on track to meet the requirements of the project), but you only have to complete the required assignments.

Because this project relies heavily on peer review, we and your peers appreciate you staying on schedule with the deadlines. This is the best way to ensure that you and your peers get the feedback they need to pass this course.

Here is an overview of the assignments in each week to help you get a sense of the project and plan your time:

- **Week 1: Warm up assignment (required, auto-graded)** In this first assignment you will start to work with some real-world social network data and implement some graph algorithms to help you warm up

and dive back into data structures and Java programming.

- **Week 2: Project scope and definition (required, peer-graded)** In this second week you will define the scope of your project. This assignment will involve selecting a data set, determining what questions you will ask about your data, finding or developing the data structures and algorithms to answer those questions, and doing a preliminary analysis of the running time of these algorithms.
- **Week 3: Mini-project and report (optional, peer graded)** One of the questions you will pick in Week 2 to ask about your data set is required to be relatively easy to answer. This will allow you to go through the "full cycle" of the capstone project from question formulation to implementation to report and get some early feedback on each of these components, before you repeat the process on a more challenging question and algorithm. In this week, you'll do this first "full cycle".
- **Week 4: Main project checkpoint (optional, peer graded)** In week 4 you should be well into the investigations of the questions you laid out in week 2, and you should have already gotten some feedback from your peers about the scope of your projects which you will be using to guide and refine your work. This week you have the option to report on your progress and again get guidance from your peers to ensure that you will pass the main report assignment in week 5.
- **Week 5: Main project report (required, peer graded)** In week 5 you will complete the implementation of your project and submit your code and a final written report (including an analysis of your algorithm(s)) for peer review. You will also perform peer review on your peers' submitted project reports.
- **Week 6: Oral project report (required, peer graded)** In week 6 you will complete an assignment very similar to the assignment you completed in week 3 of our course on Mastering the Software Engineering Interview. We will ask you to submit a video in which you give an oral presentation addressing specific aspects of your project. You will also perform peer review on your peers' submitted oral presentations.

Project Requirements

After a short warm-up assignment, the scope and direction of this project is very much up to you, and we look forward to seeing a lot of creativity in your projects! However, there are a few basic requirements for your work and some things that you should know to help you select the direction of your project.

1. You must work with real-world social network data. It does not have to be extremely large, but it must have enough data to make it interesting (usually at least a hundred nodes or more). You may use any of the data sources we provide with the starter code, or you may use any datasource you legally obtain online. This github “awesome list” is a great place to look: https://github.com/caesar0301/awesome-public-datasets#social-networks
2. You must propose at least two questions that you will explore via the code you write. One of these questions must be relatively easy to answer (e.g. you should be able to write the code to answer it in less than a week) while the other should be more complex. Examples of the types of questions we expect will be given in week 2.
3. You must implement code to explore the questions you propose on your data set. It is not enough to simply explore the data in a program like excel. We are looking for a sophisticated program that

implements at least a graph and at least one (though probably many more) graph algorithms (see next requirement).

4. To answer your "harder" question, you must implement at least one new graph algorithm that you have not previously implemented in this specialization. As a reminder, the algorithms you have previously implemented include breadth-first search, Dijkstra's Algorithm, and A* search, though for the purpose of the capstone, DFS doesn't count as a new graph algorithm either.
5. Your overall grade on the capstone project will be based on a combination of: (1) the scope of your project--the more ambitious the project, the more points you can earn here, (2) the quality of your code and implementation--your peers will be reviewing your code, (3) how well you were able to answer the questions you posed, or how well your program allows someone to explore the answers, including the appropriateness of your algorithm and data structures that you chose to implement (4) the correctness of your analysis of the running time of your algorithm, (5) your oral presentation in week 6.
6. Your grade on this project will be used, along with your performance in the rest of the specialization, when we select learners who will get invited to "perks" associated with the class, for example invitations to be community mentors and invitations to participate in sessions with Google engineers.