Social Graph

Kristina Markina, Kyle Lawrence, Kristen Glielmi, Lilach Hacohen

This project uses the graph data structure to create a way to track social relationships. The main class of the project 'SocialGraph' has several methods to make this possible. You can add a person into the graph with the method *add_user*, you can remove a person from the graph with *remove_user*, you can create relationships/ connections between people using *add_relationship*, and finally you can remove a connection between people using *remove_relationship*. There are a few more methods such as *clear*, *is_empty*, *__len__* and *__eq__*, as well as *to_json*– a method to convert the graph into the json format to make the transfer of this data to other applications much simpler.

The last important method of this class is *generate_matrix*. Calling this method generates and returns an adjacency matrix in the form of a pandas DataFrame. The matrix can be returned in string form as well by passing in any value to the str_ parameter.

To make creating and editing the graph quicker we have created a simple flask application to host a small series of webpages. The layout of all the pages is identical; a form for changing graph data on the left, and the current graph and adjacency matrix on the right. Upon submitting data in any form option, you will be forwarded to a new URL containing the same page layout but with an updated version of the graph and adjacency matrix. This is done by sending the data to the flask app using the POST method, processing the data/editing the graph, then calling methods to update the graph display on the new page.