Interactive, web-based visualization of scientific collaboration networks

Ben Leone, Ryuichi Ohhata, Lukas Rosario

Jinyu Tian, Angela Vellante

Mentors:
- Eric Kolaczyk (Hariri Institute, Director)
- Arezoo Sadeghi (SAIL)
- Margot Menestrot from Red Hat UX team
- Joe Farmer (BU Office of Research, Program Manager & Data Analyst)
- Jonathan Chamberlin (Teaching Fellow)

BOSTON UNIVERSITY

1

Hariri Institute for Computing and Computational Science and Engineering

→ Initiates, catalyzes, propels collaborative research and training initiatives

Problem
- https://www.bu.edu/hic/people-2/faculty-affiliates/
- 250 faculty affiliates
- Represented in **table** format with limited search capabilities
- Difficult to see everyone involved and the collaborations between them

Solution
- Interactive visualization of a network collaboration

Project Goals

1 Create a cleaner view of the affiliates than the current spreadsheet design which uses only raw HTML code

2 Provide an intuitive user experience that allows for easier navigation of the connections between collaborators using a network visualization

3 Provide researchers with tools to find similar collaborators based on fields of interest, past work and articles, etc.

4 May serve as a base prototype for other departments to implement a visualization of academic collaboration

BOSTON UNIVERSITY

Users

- Faculty affiliates of the Hariri Institute for Computing
- Potential collaborators and researchers looking for assistance in their work
- BU faculty members and non-members who are conducting teaching/training initiatives in computing or computational science
- Anyone interested in how the BU departments and colleges work together and what work is done in the fields of computing and computational science

MVP

- A clean and intuitive visualization of the collaboration network between affiliates within the Hariri Institute
    - Nodes in the visualization are the affiliates
    - Edges that connect the nodes are primarily the number of collaborations
    - A search/filter function to allow more granularity
    - The network visualization will be dynamic (i.e. clicking on a node displays its connections clearly to the user)
    - Links to academic works will be included, when possible
- The size of the network will be ~250 affiliates
    - Not all affiliates will be presented at the initial landing on the visualization
    - Possible breakdown by department to then show affiliates within selected department

BOSTON
UNIVERSITY

5

Setbacks

- Creating a persistent storage database in case it goes down
- Connecting the PosgreSQL database to the hosted API on OpenShift
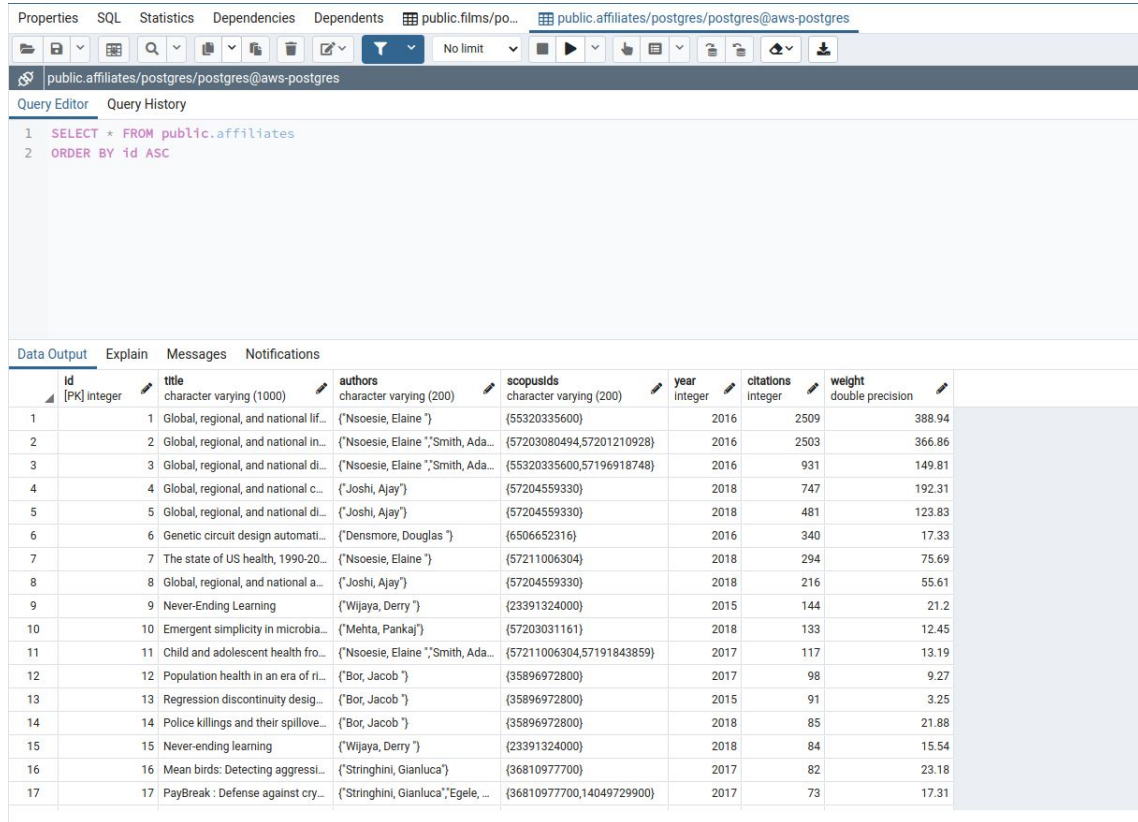
Setbacks

- Connecting to database using GUI (i.e. pgAdmin) to upload the data to be used by the API
- [Port forwarding](#) did not expose the database to the service

Current Obstacles

- Using AWS RDS free tier as interim solution

- Slow for data uploads (db.t2.micro)
  - 1 vCPU
  - 1 GiB RAM
  - 20 GiB Disk

Database

- PostgreSQL
- Python scripts that scrape Affiliate list and SciVal papers and upload to Database
- Name matching algorithm
- Easy to maintain and change

BOSTON
UNIVERSITY

Database

Three tables:

- affiliate_info – each Hariri affiliate's information
- affiliate – papers the affiliates published
- relation – how many papers an affiliate collaborated with other affiliates

Database

API

- Python Flask, Blueprint, and RESTful to construct API
- Applies credentials to DB and retrieves data

Three endpoints:

- /api/v1/info – information about each affiliates
- /api/v1/members – affiliates' published papers
- /api/v1/relations – Lists affiliates under "nodes" and their co-authorships under "links"

BOSTON
UNIVERSITY

API

- Dockerfile to containerize the API application
- Hosted on OpenShift
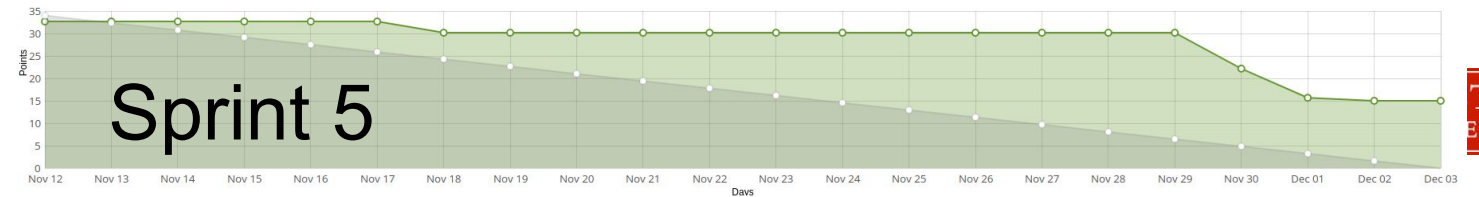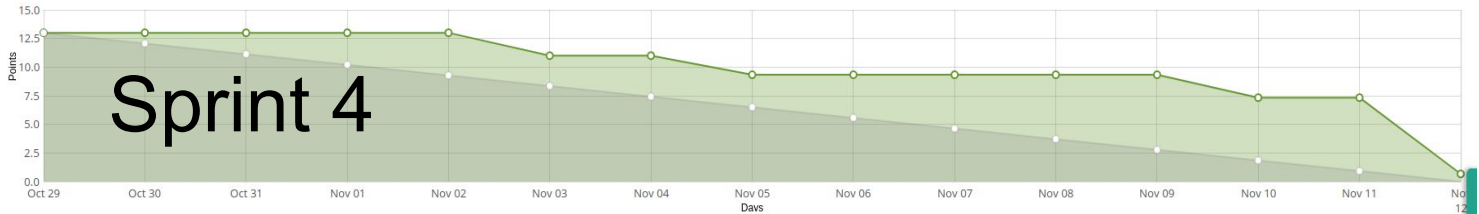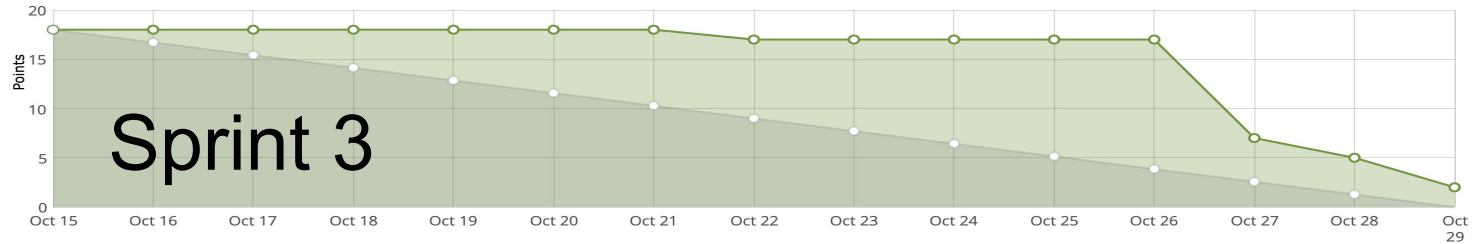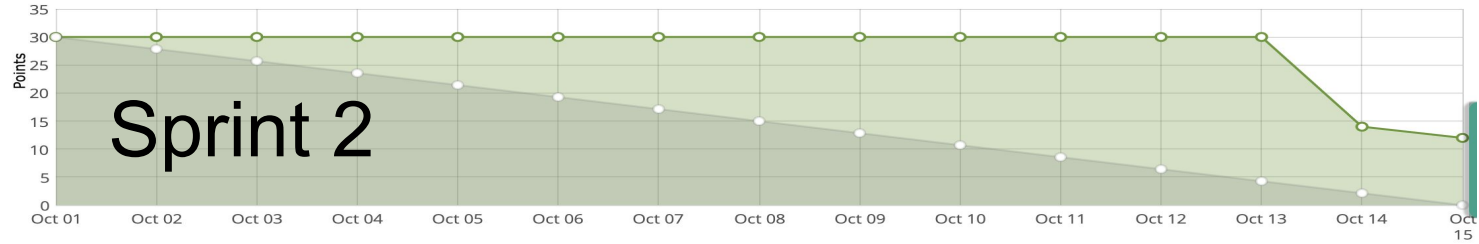- http://api-flask-ece-528-interactive-web-visualization.k-apps.osh.massopen.cloud/api/v1/relations

API

Source: Affiliate 1
Target: Affiliate 2
Value: How many collaborated papers

Sprint 2

Sprint 3

Sprint 4

Sprint 5

Lessons Learned

- WordPress is an unfriendly environment to do web development outside of basic features

- OpenShift databases only directly communicate with other resources inside the same container

- Working in an agile way with Taiga was difficult at first, but proved to be useful for keeping us on track

Project Limitations and Future Improvements

- Displaying name of publication when link is clicked on
- Functional
    - Search Bar
    - Isolated Button
        - Option for displaying only the nodes without links
    - Extended Button
        - Option for displaying only nodes with links
- UI/UX clean up
- Key/Legend
- Zoom Scale

BOSTON UNIVERSITY

Thank you!
Questions?