Show Me Related Work

Group 6

Abhishek Dutta, Apoorv Khairnar, Chiraag Subramanian
Pranjal Karankar, Pushkar Ladhe, Rajat Sinha,
Sachin Magar

Sponsor: Dr. Milos Gligoric
Course Instructor: Mr. Robert Reimar Heinrichs

Department of Software Engineering
Arizona State University

May 2018

Preface

This document is designed for helping software engineers who are studying this application or working on it for further development. It describes the motivation, features and the steps required to install, deploy, setup and run this project from scratch. Each chapter tells about the functionality of this application and understand the purpose of it. If you are a software engineer, this document will help you understand the project, the features that have been implemented, and the steps to setup and deploy this project from scratch. If you are a researcher or a student, then this document will explain how this product will help you in your research work, express your views on different papers and benefit from other user’s ratings and comments.

Table of Contents

1a Introduction

1a.1 Project description:

Show me project is intended to help the researchers in the research community to find relation between research papers and help each other in this process. Objective of this project is to display the relation between research papers in such a way that most information is displayed graphically which helps the user to discern it with minimal efforts. As an effort to give back to the research community, researchers can provide feedback on the research papers which can be useful for others.

1a.2 Motivation:

It is hard to find the strength of relation between research papers. There are existing applications like google scholar which provide the information such as citations for research papers. These applications, however do not provide the information about the strength/relevance of paper citations. If a user wants to choose the next paper to continue their research, there is no way other than going through all references. This process is time consuming process. Making the process of determining the relevance of the relation between papers, a community driven activity, not only saves the time of the user but also increase the reliability with feedback from diverse set of people.

1a.3 Our approach of addressing this problem:

We are displaying the relation between research papers in the form of a graph with nodes as research papers and edges as the relation between these papers. We are providing users the option to upvote or downvote a relation. The number of upvotes and downvotes together is considered as weight for the relationship. This weight determines the strength of the relations as in the more number of users upvote a relation, more closely are the papers related. Sometimes, it's hard to provide the feedback just in the form of upvote and downvote so we are also providing option of commenting on a relation. Along with the information about relation between papers, we are also providing all the required attributes of the research papers. By providing this information, we are making sure to provide users all the necessary information at single place.

1a.4 Issues/Risks and Resolutions:

Being a crowd-sourced application, the biggest risk is to deal with fake upvotes / downvotes. Providing feedback through bots can defeat the purpose of the application. To address this issue, we implemented authentication to the application. A visiting user can see all the information about papers and their relationships, however user need to login if they want to provide feedback in the the form of upvote / downvote or comment. For authorization, we used Auth0 service which allows user to login using Facebook or LinkedIn (More application like Google, Yahoo etc. can be added).

This application is heavily dependent on data about research papers and this data is not easily available. Most online resources either do not provide information about all the attributes that we needed or do not allow crawling. To address this issue, we identified APIs provided by IEEE which provide information about the required attributes. We further developed crawlers on the APIs to fetch the information.

1b Product Features

1b.1 Problem Statement:

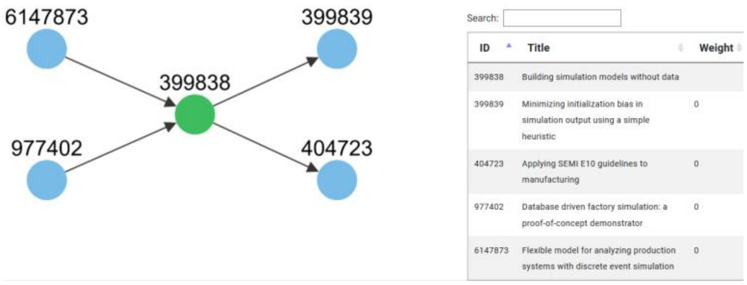
Since the goal of this project has been to help the research community to find and understand related work as well as help each other in the process, the base requirement of this project was to visualize relations among papers as graphs. Each node is a paper and there should be a direct edge between two nodes if one paper cites another. A click on the node in the graph should redirect to the paper pdf (if available), and a click on the edge of the graph should show the way one paper cites another. The users should also have the option to mark an edge as strong (i.e. paper is closely related) or weak. This data could help other users to navigate through the graph more efficiently.

1b.2 Requirements:

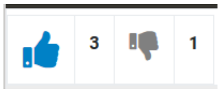
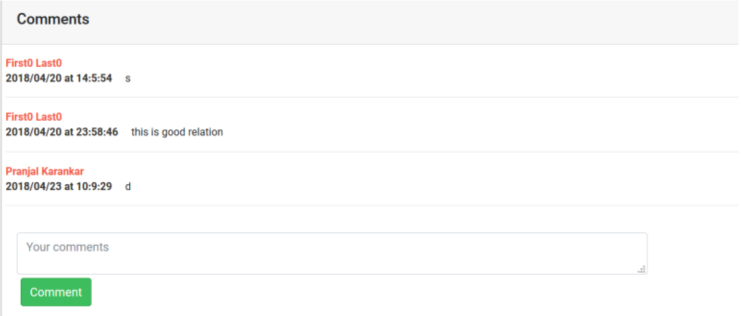
The following table contains a list of features and description about how to use them. This would allow you to understand the feature as well as its high-level implementation.

<table>
<tr><th>SNo.</th><th>Feature</th><th>Description</th></tr>
<tr><td>1.</td><td>Graphical visualization
<b>Sponsor Requirement</b></td><td>We have used cytoscape, a library for graph visualization, to display papers and the relation between them, as a graph. The nodes of the graph represent the papers and the edges represent the relationship between the papers. Incoming edges to a selected paper represent the papers that have referenced the selected paper whereas the outgoing edges represent the papers referenced by the selected paper.

You can click on the arrows(relations) to get further details about them. The green node represents the current node wherein the blue nodes are the immediate neighbours of the green node.

You can click the blue nodes to navigate to that node. This will make that node green (current node) and display all its neighbours.
You can use the legend (table) to the right in the image below to see details of each node in the graph.


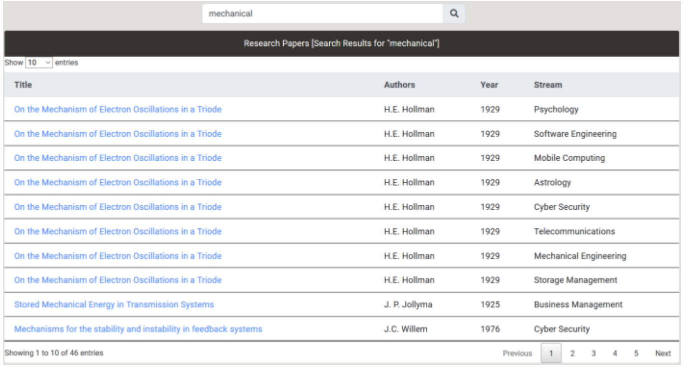
<table><tr><th>ID</th><th>Title</th><th>Weight</th></tr><tr><td>399838</td><td>Building simulation models without data</td><td></td></tr><tr><td>399839</td><td>Minimizing initialization bias in simulation output using a simple heuristic</td><td>0</td></tr><tr><td>404723</td><td>Applying SEMI E10 guidelines to manufacturing</td><td>0</td></tr><tr><td>977402</td><td>Database driven factory simulation: a proof-of-concept demonstrator</td><td>0</td></tr><tr><td>6147873</td><td>Flexible model for analyzing production systems with discrete event simulation</td><td>0</td></tr></table></td></tr>
</table>

2.	Upvote - Downvote **Sponsor Requirement**	This feature provides weightage to the relationship between two papers. If a user finds the relationship between two papers relevant, they can click on the edge between two papers and upvote the relation by clicking the 'thumb up' button, else if they do not feel that the relationship is strong enough, they can downvote the relationship by clicking on the 'thumb down' button. This feature is limited to users who have logged in to the ShowMe application to avoid bot attacks. 
3.	Comments **Sponsor Requirement**	Users can provide comments to the relationship between any two papers hence providing their feedback regarding the relevance of the relationship between the papers. This can be done by clicking on the edge between two paper nodes and then typing a comment in the displayed comment section. This feature is limited to users who have logged in to the ShowMe application to avoid bot attacks.  Comments First0 Last0 2018/04/20 at 14:5:54 s First0 Last0 2018/04/20 at 23:58:46 this is good relation Pranjal Karankar 2018/04/23 at 10:9:29 d
4.	Crawler **Sponsor Requirement**	A crawler was constructed to continuously crawl research paper data directly from IEEE. The crawler extracts information regarding the paper along with the information about the referenced papers. The data extracted by the crawler is directly pushed into the databases without any manual intervention. The crawler can be left running in the background but if it is stopped, on restarting it will resume crawling from the point it stopped. You can find detailed set up steps for the crawler in the installation/setup section.

<table>
<tr>
<td>5.</td>
<td>Search
<strong>Enhancement Feature</strong></td>
<td>

Users can search for any paper by typing in the author’s name, year of publication, or a keyword present in the title of the paper.
On searching, a paginated table is displayed with a list of papers matching the searched keyword.
On selecting a paper from the table, the graphical visualization for the selected paper is displayed.

This feature internally uses MongoDB’s full text search using mongo indexing for fast and efficient text matched search.



mechanical

Research Papers [Search Results for "mechanical"]

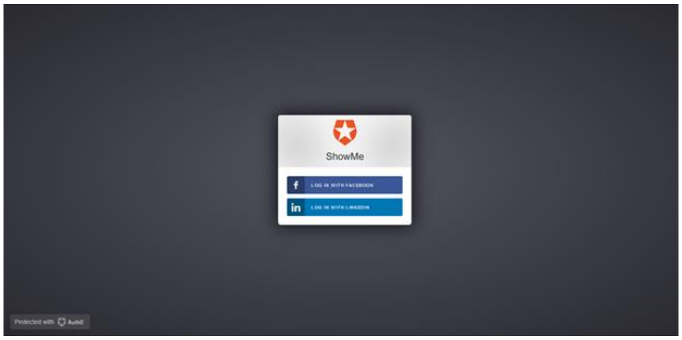
Show 10 entries

<table>
<tr><th>Title</th><th>Authors</th><th>Year</th><th>Stream</th></tr>
<tr><td>On the Mechanism of Electron Oscillations in a Triode</td><td>H.E. Hollman</td><td>1929</td><td>Psychology</td></tr>
<tr><td>On the Mechanism of Electron Oscillations in a Triode</td><td>H.E. Hollman</td><td>1929</td><td>Software Engineering</td></tr>
<tr><td>On the Mechanism of Electron Oscillations in a Triode</td><td>H.E. Hollman</td><td>1929</td><td>Mobile Computing</td></tr>
<tr><td>On the Mechanism of Electron Oscillations in a Triode</td><td>H.E. Hollman</td><td>1929</td><td>Astrology</td></tr>
<tr><td>On the Mechanism of Electron Oscillations in a Triode</td><td>H.E. Hollman</td><td>1929</td><td>Cyber Security</td></tr>
<tr><td>On the Mechanism of Electron Oscillations in a Triode</td><td>H.E. Hollman</td><td>1929</td><td>Telecommunications</td></tr>
<tr><td>On the Mechanism of Electron Oscillations in a Triode</td><td>H.E. Hollman</td><td>1929</td><td>Mechanical Engineering</td></tr>
<tr><td>On the Mechanism of Electron Oscillations in a Triode</td><td>H.E. Hollman</td><td>1929</td><td>Storage Management</td></tr>
<tr><td>Stored Mechanical Energy in Transmission Systems</td><td>J. P. Jollyma</td><td>1925</td><td>Business Management</td></tr>
<tr><td>Mechanisms for the stability and instability in feedback systems</td><td>J.C. Willem</td><td>1976</td><td>Cyber Security</td></tr>
</table>

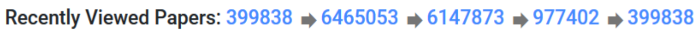
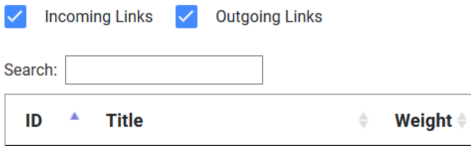
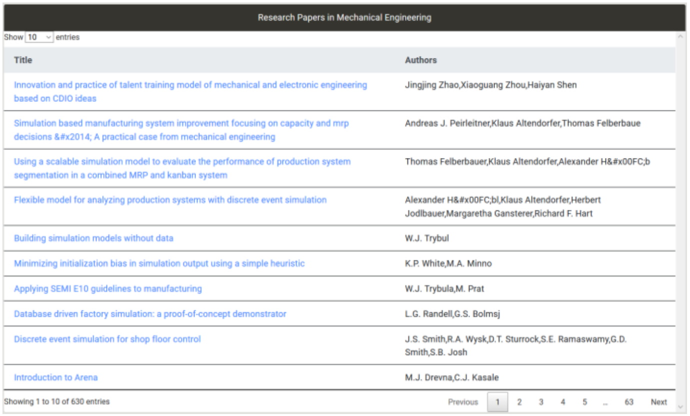
Showing 1 to 10 of 46 entries

</td>
</tr>
<tr>
<td>6.</td>
<td>User Login (Auth0)
<strong>Enhancement Feature</strong></td>
<td>

User authentication has been carried out with the help of Auth0. Currently the users can login using either their LinkedIn or Facebook accounts.
Auth0 is a third-party authentication manager that allows easy management of social sign ons.
The feature allows us to avoid bot attacks on ShowMe application’s upvote, downvote and comment feature which accounts for weightage of the graph, and hence is very critical to the application.
Once logged in, users can upvote or downvote the relationship between two papers and provide comments to the relationship.



</td>
</tr>
</table>

<table>
<tr><td>7.</td><td>Breadcrumbs
<b>Enhancement Feature</b></td><td>Breadcrumbs feature stores the last 5 papers traversed by the user. This helps the user to see the navigation path they took to reach their current paper.
This information is stored as session information and hence exists for the current session only.

Recently Viewed Papers: 399838 ➡ 6465053 ➡ 6147873 ➡ 977402 ➡ 399838</td></tr>
<tr><td>8.</td><td>Graphical visualization filters
<b>Enhancement Feature</b></td><td>Graph visualization filters are useful when the graph is dense and/or user needs to concentrate on either the incoming or outgoing nodes.
Additionally, this feature also allows us to sort the legend using Paper id, title or relation weight. The search bar can be used to search within the legend in case the graph is too dense and user wants to find a specific paper in the graph.

☑ Incoming Links ☑ Outgoing Links
Search:
ID ▲ Title ⬍ Weight ⬍</td></tr>
<tr><td>9.</td><td>Pagination
<b>Enhancement Feature</b></td><td>Paginated tables have been implemented to display a list of papers.
A paginated table is displayed either when a user searches for a paper (resulting papers containing the searched keyword) or when a domain is selected (papers available in the selected domain).
A filter is also available allowing the user to view 10, 25, 50, or 100 papers at once in the paginated table.
By default, 10 papers are displayed.

Research Papers in Mechanical Engineering
Show 10 entries
<table>
<tr><th>Title</th><th>Authors</th></tr>
<tr><td>Innovation and practice of talent training model of mechanical and electronic engineering based on CDIO ideas</td><td>Jingjing Zhao,Xiaoguang Zhou,Haiyan Shen</td></tr>
<tr><td>Simulation based manufacturing system improvement focusing on capacity and mrp decisions &#x2014; A practical case from mechanical engineering</td><td>Andreas J. Peirleitner,Klaus Altendorfer,Thomas Felberbaue</td></tr>
<tr><td>Using a scalable simulation model to evaluate the performance of production system segmentation in a combined MRP and kanban system</td><td>Thomas Felberbauer,Klaus Altendorfer,Alexander H&#x00FC;b</td></tr>
<tr><td>Flexible model for analyzing production systems with discrete event simulation</td><td>Alexander H&#x00FC;bl,Klaus Altendorfer,Herbert Jodlbauer,Margaretha Gansterer,Richard F. Hart</td></tr>
<tr><td>Building simulation models without data</td><td>W.J. Trybul</td></tr>
<tr><td>Minimizing initialization bias in simulation output using a simple heuristic</td><td>K.P. White,M.A. Minno</td></tr>
<tr><td>Applying SEMI E10 guidelines to manufacturing</td><td>W.J. Trybula,M. Prat</td></tr>
<tr><td>Database driven factory simulation: a proof-of-concept demonstrator</td><td>L.G. Randell,G.S. Bolmsj</td></tr>
<tr><td>Discrete event simulation for shop floor control</td><td>J.S. Smith,R.A. Wysk,D.T. Sturrock,S.E. Ramaswamy,G.D. Smith,S.B. Josh</td></tr>
<tr><td>Introduction to Arena</td><td>M.J. Drevna,C.J. Kasale</td></tr>
</table>
Showing 1 to 10 of 630 entries
 </td></tr>
</table>

1c Installation & Setup

1c.1 Prerequisites:

- MongoDB
- Neo4j
- Node Server
- Node JS
- Clone the git repo provided in the link section

1c.2 MongoDB:

1c.2.1 Download and Installation

For Windows: https://www.mongodb.com/download-center#enterprise

For Linux: https://docs.mongodb.com/v3.0/tutorial/install-mongodb-on-ubuntu/

Run the executable and follow the on screen instructions.

1c.2.2 Setup and run DB

Create a new database named - showMe

1c.2.3 Load data in DB

Data gets loaded directly by the crawler.

1c.3 Neo4J:

1c.3.1 Download and Installation

Follow the following links for downloading and installation:

For Windows: https://neo4j.com/download/

For Linux: https://neo4j.com/docs/operations-anual/current/installation/linux/debian/

Run the executable and follow the on screen instructions.

1c.3.2 Setup and run DB

- Run Neo4J Community edition and click on Start
- Now browse to the URL - http://127.0.0.1:7474/browser/
- Run the query -: server connect
- Enter “neo4j” as the username and password
- Set the new password

1c.3.3 Load data in DB

Data gets loaded directly by the crawler.

1c.4 Node Server:

1c.4.1 Prerequisites

Install Node using the following link: https://nodejs.org/en/download/

Run the executable and follow the on screen instructions.

1c.4.2 Installation and Configuration of server

Follow the given instructions below (Presuming that you have cloned the Git repo provided in section 2.1):

- Open terminal and go to the path of Git repo
- Run $cd .\server\Show-Me-App\
- Run $npm install
- Update the URL and credentials for MongoDB and Neo4J in the following file:
  - /server/Show-Me-App/config.json
- Run $npm start. (Your server should be starting now)

1c.5 Crawler Infrastructure and Setup:

There are two major steps to setting up the crawler the infrastructure.

1c.5.1 Set up the environment

Crawler environment needs Python 2.7 installed in the OS.

To install Python:

- Download Python from here: https://www.python.org/download/releases/2.7/
- Run the executable and follow the on screen instructions.

1c.5.2 Execute the Crawler

- Go to the directory - Show-Me-Related-Work/Crawling/**NodeQueryRunner**/
- Here execute the command - npm install
- Go to the directory - Show-Me-Related-Work/Crawling
- Start running the crawler by executing the command - python crawler.py

The crawler will continue running until it’s stopped manually. If and when the crawler is restarted, it will resume crawling from the point it was stopped.

1c.6 Frontend Setup:

1c.6.1 Prerequisites

Install Node using the following link: https://nodejs.org/en/download/
We need Node package manager(NPM). Latest node version comes with NPM.

Run the executable and follow the on-screen instructions.

1c.6.2 Installation and configuration

- Follow the given instructions below (Presuming that you have cloned the ShowMe git repo):
- Open terminal and go to the path of Git repo - /FrontEndDevelopment/vue/**show-me**/
- Run $npm install
- Change the 'IP_Config' value as per requirement (IP address and port) in the following config file:
    - /FrontEndDevelopment/vue/show-me/src/**store.js**
- For Auth0 configuration, after configuring an account, setup the configuration file, to the corresponding feeds:
    - /FrontEndDevelopment/vue/show-me/src/Auth/**auth0-variables.js**
    - Update the values for the following tags - clientId, domain, callbackUrl, apiUrl
    - Auth0 webpage - https://auth0.com/
- Run $npm run dev

1d Links

Video Link: https://youtu.be/e7mq6n3wNc8

Github repo: https://github.com/shinigami1392/Show-Me-Related-Work