Bilkent University

Department of Computer Engineering

Senior Design Project

Paper Atlas - T2334

Project Specifications Report

Ahmet Hakan Yılmaz - 21803399

Akın Kutlu - 21803504

Aybala Karakaya - 21801630

Selbi Ereshova - 21901326

Zehra Erdem - 21801977

Supervisor: Uğur Doğrusöz

Jury Members: Erhan Dolak and Tağmaç Topal

Contents

1. Introduction

Research papers can be reached online with the help of internet and online platforms such as Google Scholar. However, because of the increasing number of research papers, it becomes hard to keep track of information [1]. Reaching newest findings related to a topic or old papers that explains some fundamentals becomes hard and time consuming. Though lots of search algorithms are used to find papers related to a topic, finding relations between these papers is still tedious [2]. It needs to be improved. Reference relations between papers are important because if the papers that cited a given paper are known, the newest finding related to that paper can be found easily. Moreover, to understand a topic better, some old papers need to be looked at. To find the ones that have fundamental information about the topic, commonly cited papers by already known papers should be looked at. However, finding the common references can be hard when papers are looked at separately. This application, Paper Atlas, makes it easy for researchers to find correct papers by visualizing papers and their reference connections as a graph.

The innovation we are planning to implement is process innovation. Process innovation aims to implement a new or significantly improved product or service. Paper Atlas will be an application that provides services for searching papers and visualization of these papers as a graph. After deciding on the project topic, similar projects were looked at, and it was found that there is a website that provides a service for graph visualization of papers. We want to create a new application by improving this old application by adding new features. As we have an example website we want to improve, and we will add new features different from the previous websites. The right innovation type for Paper Atlas is process innovation. When we also look at other types of innovations one by one, we observe that process innovation is the innovation that resembles most of the innovations that we are going to implement in Paper Atlas. This will be a transformative innovation because we are creating an application that provides significant operational enhancement.

The innovation of this project will be a sustained innovation. The new improvement that uses graph visualization satisfies the needs of the users that have not been satisfied. The sites have been providing services for reaching other research papers using lists to show cited papers or the papers that cite a paper. This becomes problematic when they need to compare lists of references to find the correct paper. Paper Atlas provides a visual version of these references and papers. With this innovation, users do not need to look at the references as lists, but they can easily detect the related or most used papers by looking at all reference relations of the papers within a graph visual.

The purpose of this report is to inform about the project description, discuss the ethical and professional sides of the project and give constraints, functional and non-functional requirements of the project.

1.1 Description

Paper Atlas will be a web application available on all modern web browsers. With Paper Atlas, users can visualize the hierarchy of research papers as a graph-based and interactive structure in order to find the most relevant paper to their topic. The goal of this project is to enhance the way academicians and students conduct their research by providing a better way to find papers that best match their research topic by showing the hierarchy of papers related to the input paper of the user. This application will show the papers as nodes with edges specifying which one references which one and which one is being referenced by which. The users will be able to read the details such as name, keywords, and abstract by clicking on the node of a paper. A link to the original paper will be available as well. The graph will be interactive; in other words, the users can move the nodes around, zoom in and out, apply layouts, and highlight nodes. Users can filter their results by time interval, keywords, author, and publication journal. Users can also search authors or keywords instead of paper titles. In this case, relative nodes will be shown for authors or keywords. The application will also provide extended features such as grouping of the nodes and ranking papers by their importance.

1.2 Constraints

1.2.1 Implementation Constraints

Node.js framework will be used for the implementation of backend with Javascript. Cytoscape.js library will be used for visualization of the graphs. Neo4j which is a graph database will be used to store information. Cypher will be used for graph query language. Response time will be less than two minutes.

1.2.2 Data Constraints

Publicly published papers will be used as data. Publicly accessible information of papers such as author, title and abstract will be stored and used.

1.2.3 Cost Constraints

Open API's will be used to gather papers. Google Scholar and other free websites will be used to gather data.

1.2.4 License Constraints

For all the used libraries, APIs and frameworks, if their licenses are appropriate for the project, they will be used.

1.2.5 Schedule Constraints

Before the first semester ends, a prototype demo will be done. Before the CS Fair 2023 the project will be done.

1.3 Professional and Ethical Issues

1.3.1 User Data

We will not store any data about our users.

1.3.2 Accuracy

The graphs we create must display the data correctly. Additionally, all the information we give to the users must be accurate.

1.3.3 Right to be Forgotten

If an author gets into contact with us stating that they want to be removed from the graphs created on our website, we must remove them from our graphs in the future.

2 Requirements

2.1 Functional Requirements

2.1.1 Viewing Papers as an Interactive Graph

The user can view research papers and their reference relations within a graph. Papers will be shown as circles which are called nodes. İf a paper has reference to another paper, this reference relation will be indicated with an arrow between nodes of the papers. These arrows are called directed edges between nodes. It will be an interactive graph. It means the user can move the nodes of the paper and can change the appearance of the graph or apply a different layout to get a different appearance.

2.1.2 Viewing the Details of a Paper

The user can view details of a paper by clicking the node of the paper. The details are the title, authors, keywords, journal it was published and abstract of the paper. A link to the original paper will be given.

2.1.3 Viewing Papers of an Author

The user can choose an author and view the papers of the author and reference relations of the papers as a graph.

2.1.4 Viewing Papers and Related Keywords

Users can select some keywords and can see papers that are related with these as a graph visualization. Papers and keywords will be shown as nodes while the reference relation of the papers and which keywords belong to which paper will be shown as edges.

2.1.4 Grouping Papers Based on Journals

The papers can be grouped while preserving structure of the graph. They will be in the group of the journal they have been published in.

2.1.5 Searching by Different Parameters

The user can search authors, titles, keywords, journals or universities while specifying what kind of input they are providing.

2.1.6 Filtering Based On Different Values

The user can filter search results with different criteria. The user also can use these filters at the same time as filtering based on multiple values.

2.1.7 Filtering Output Based on Time Interval

The user can select a start time and an end time to obtain papers which are published between the chosen time interval. If the end date is not specified then the end date will be taken as today.

2.1.8 Ranking papers based on ‘Cited by’

The nodes that represent papers can be ranked by how many other papers have referenced this paper. The users have an option to enable or disable this feature. The ranking of nodes can be shown by the size of the nodes, by the color of the nodes or by a badge on the nodes.

2.1.9 Ranking papers based on ‘Cited’

The nodes that represent papers can be ranked by how many references it has or how strong are these references. The users have an option to enable or disable this feature. The ranking of nodes can be shown by the size of the nodes, by the color of the nodes or by a badge on the nodes.

2.1.10 Find Common References of given papers

The users can search more than one research paper and find common reference papers of these papers. The user can also specify to look down to more than one level of references. For example, if a user chooses max level as 2 then they will see a common reference of the references of the papers they have input if exists; common reference of references.

2.1.11 Displaying a Graph of Authors

The users can view a graph with nodes representing authors. This will be a directed graph. An edge from Author A to Author B will show how many references are made between authors. The users can view the list of references from Author A to Author B by clicking on the related edge.

2.2 Non-Functional Requirements

2.2.1 Performance

The response time for the application besides running layout and retrieving a graph from the database should not be more than 30 milliseconds. But since a graph can be very large, running layouts and retrieving such graphs from the database can take longer but should not exceed 1 minute.

2.2.2 Usability

The user interface of the application should be user friendly. Everyone using the application should be able to understand how to handle functionalities of the application. As the application is based on graph visualization, graphs should have interactive design for everyone to use it easily.

2.2.3 Maintainability

As the initial goal of the application is to provide services for a specific topic, the application should be able to continue to function if it is decided to provide services for other topics. In addition, information that the application provides should be up to date.

2.2.4 Scalability

The application must be able to handle 10000 users at the same time.

3. Risks

3.1 Fetching Data

The application needs to fetch data in order to construct the graphs. We will use some APIs for that. One risk is that we might not be able to find a suitable API, or the APIs might have some limitations, such as only returning the name of the paper instead of its authors or having a cap on how many queries we send in a specific time frame.

3.2 Storing Data

The application will store paper information in a database. Another risk occurs when there is too much data to be stored since it would be difficult to maintain the database.

3.3 Complex Graphs

The application will perform graph visualization. However, when the graph is complex, for example, because there are too many edges since some papers have many common references, the graph might not seem very neat.

4. Expected Goals

The expected goals of this projects are:

- To make researchers' work easier when they need to look at so many papers.
- To visualize and connect the papers according to their references in an interactive graph.
- To search and filter papers with different parameters.

According to our plans in the first stage our database will not be large scale, it will be related to some specific topics' papers but it will be small scale. After our application is developed and tested we can enhance the dataset of our application. Paper Atlas will have different functionalities related to graph visualization of papers, authors, journals, keywords or citations. When developing the application priority functionality will be graph visualization of papers with reference relations to each other. After completing one functionality, other functionalities will be implemented easier as they include similar parts.

5. References

[1] M. Gusenbauer and N. R. Haddaway, “Which academic search systems are suitable for systematic reviews or Meta‐analyses? evaluating retrieval qualities of google scholar, pubmed, and 26 other resources,” *Research Synthesis Methods*, vol. 11, no. 2, pp. 181–217, 2020. [Online]. Available: https://onlinelibrary.wiley.com/doi/full/10.1002/jrsm.1378. [Accessed: 10-Oct-2022].

[2] A. Martín-Martín, E. Orduna-Malea, M. Thelwall, and E. D. López-Cózar, “Google scholar, web of science, and Scopus: A systematic comparison of citations in 252 subject categories,” *Journal of Informetrics*, 05-Oct-2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1751157718303249. [Accessed: 10-Oct-2022].