CREATING AN ACADEMIC DATABASE

Harshda Ghai

Introduction

Due to the rise of recent technological developments, the volume and availability of data has increased substantially. This sudden rise in the availability of information is likely to continue to expand, resulting in new opportunities and challenges in different fields. This phenomenon of increasing data has had a significant impact on the generation of research papers.

While this increase in research papers and other forms of research data is a positive outcome, locating relevant research papers has become a challenging task.

A few common problems faced when trying to access research resources include -

- Subscriptions - Several research resources are published in journals that require subscriptions, making it hard to access for people people who the lack the financial resources
- Restrictive Copyrights - Other papers may have limiting copyright agreements, which restricts the distribution and thereby access to resources.
- Complicated Search Inputs - Finding revenant research papers on the internet is also not an easy task. Several research papers require extremely complex search queries. Being able to input efficient queries to find required resources is a skill that many people are not familiar with.
- Changing Website Locations - Even when these papers are located, users face the problem of constantly changing URLs that makes it hard to trace papers.

However, the biggest challenge to the widespread availability of research papers is the lack of centralisation. Without the presence of a central medium of distribution (such as a database), research papers are dispersed over multiple platforms and websites. The need to search multiple websites and sources makes finding papers extremely inefficient.

To address these challenges, this paper talks about leveraging the advantages of a relational database to introduce a level of centralisation amongst the vast number of research papers available. This will allow users to navigate the various resources more efficiently, without having to scrap information from different parts of the web.

In this paper, I use the OpenAlex catalog to construct an academic database to provide efficient and easy access to several research papers and journals published by individuals, research organisations, and institutions across different parts of the world.

Importance of an Academic DataBase

The construction of an academic database for locating research papers offers numerous benefits, including -

- Centralisation - This academic database solves the problem of centralisation. The database is a central location for a wide range of research papers, thus eliminating the need to search multiple websites and platforms.
- Efficient Search Options - This also provides users with an efficient search option. Instead of having to use complex input search queries, users can simply search the paper or organisation name and obtain results from the database. Moreover, the presence of several filter and keyword options allow users to narrow down their search results and obtain relevant papers.
- Authenticity and Credibility - The papers, institutions, and entities in the database are reviewed, therefore ensuring quality and credibility.

In short, a database provides a centralised, organised, and credible way of locating research papers. This helps users navigate the enormous amount of data available in the research community.

Challenges in Constructing an Academic DataBase

Constructing an academic database can involve several challenges, such as -

- Large Scale data collection and cleaning - Gathering sufficient research papers, journals, and other forms of content. Further, cleaning and ensuring that the entire dataset follows the same format can be a tiring process and requires a huge time investment.
- Ensuring data credibility - Ensuring that the data collected is accurate and comes from reliable sources is extremely important. This may involve going through all the papers and its citations to ensure quality control.

However, OpenAlex solves these challenges for us. OpenAlex is a fully open and accessible catalog/repository of research papers from all over the world. OpenAlex also enforces quality control by reviewing all entries in the database

Related Works

Over the past years, a significant amount of work has been done in creating and maintaining academic databases. These databases have improved the process of locating research sources and have provided users with a one stop option to browse research papers from different fields. In this section, I have reviewed relevant real-world examples of academic search engines and databases and how their presence have transformed the process of locating relevant research sources.

- Google Scholar - Google Scholar has reshaped the way researchers are able to discover research material. The user friendly interface and filter options has made it a go to website to locate research papers for several users. Google Scholar, however, ranks the search results based on citation counts, which may not provide users with the most relevant search results. Due to this ranking algorithm, users may have to go through several pages of results just to try and locate a research resources that may be more revenant to their input.

- IEEE Xplore - IEEE Xplore is maintained by the Institute of Electrical and Electronics Engineers (IEEE). It serves as a database for computer science and other engineering related topics. This platform offers users access to various kinds of sources, including - articles, journals, books, conferences proceedings and many more.
- PubMed Central (PMC) - PubMed Central (PMC) is developed by the National Library of Medicine. It is a great example of an academic database specialised in biomedical research. PubMed Central provides free access to a vast collection of peer-reviewed research articles.

These examples showcase the amount of work being done to create and maintain academic databases. Several organisations are still working on improving on the current state of these databases to improve the accuracy of results provided to users and improve the usefulness of these academic databases in locating research papers revenant to the users search.

Designing an Academic DataBase (Experiment)

In order to centralise the enormous amount of data available in the research community, I have described a database web application approach.

- The first step in this process is constructing a database. It is essential to ensure that the large amounts of data in the database is obtained from credible sources. While constructing the database, it is also necessary to make sure that all the information follows the same format.
- The OpenAlex catalog was utilised to construct the database. OpenAlex is a fully open index of hundreds of millions of interconnected entities across the global research system. Therefore, steps involving data gathering and quality checks can be skipped as OpenAlex already implements those steps.

- I set up the academic database on a MySQL server on a remote server. The first step in constructing the database is to build a schema and write SQL queries to construct the tables. OpenAlex provides a sample schema (written in PostgreSQL) that can be used for the same. However, I decided to build the schema from scratch. Using the schema, I ensured that the SQL commands for the tables have the appropriate indices and keys (primary and foreign keys) to improve query performance. (The schema and table commands are provided in a file in the GitHub repository)
- The next step to setting up the database is to import CSV files of the data available on OpenAlex. The data on the Amazon Cloud storage is in the JSON Lines format. Using a python script provided on the OpenAlex website, I was able to convert the JSON LINES file into CSV files. The last step is to import these CSV files to complete constructing the database. Due to the large size of the CSV files, the process of importing the files takes a few days.

This completes the set up for the OpenAlex academic database in SQL. After this, I worked on building the website.

- Using this database, I built a web based database application. The website used primarily HTML, CSS, NodeJS, SQL, and Python. The website returns a list of relevant research papers upon a user search query. The website uses SQL queries and OpenAlex API calls to return results to the user.
- For each result, the following information is displayed to the user - the title of the research paper, a URL that re-directs the user to the paper, a list of authors for the paper, the publication type, and the year of publication.
- The results are ordered in the form of relevancy to the user input search query. This relevancy is determined using the citation count of each papers, so papers with a higher citation count are displayed on the top.

The website also allows users to filter search results based on the following parameters - number of citations, publication year, or publication type.

This website is a basic example of using database to centralise and organise a large amount of data to provide relevant research papers upon a user search query. There can be some changes that can be made to improve the quality of user search results.

Experiment

I tested the website by asking a small group of 10 people (friends and family) to play around with the interface and asked for their feedback. The changes suggested are mentioned in the future work section of the paper.

Learnings

- User application design workflow **-** I understood how the entire process of designing a user application is streamlined. I also gained a better understanding of working with the frontend and backend and how both components interact with each other.
- Expanded my skillset by gaining better knowledge of NodeJS, Python, and SQL. I also learnt how to work with remote servers and the competitive hardware and software advantage they provide.
- Solving problems by efficient search inputs - I understood how to find solutions to a majority of my problems by efficiently crafting search inputs on Google and what kind of resources to go through to find useful solutions and new ideas to tackle problems.

Challenges faced

- Importing all the CSV files - Due to the large size of a few CSV files (80GB+), I faced several issues importing the CSV files into the SQL tables. Due to this, I had to use API calls, in the backend, to return some of the results.
- Designing a user interface - I faced a few difficulties trying to figure out the most efficient user interface that would compel people to use the website. I have taken the feedback received into account and will continue working on making the interface more attractive and user friendly.

Future Work

- Implementing a search rank algorithm - Currently, I rank the search results in decreasing order of the citation count. This may not be an accurate representation of how relevant of paper may be to the search result. Therefore, by implementing/designing an algorithm, I might be able to return more accurate results to the user.
- Adding additional search filters - By adding more search filters, users can alter results to find the most revenant research paper.
- Displaying additional information - Apart from displaying research papers, other information such as related works and referenced works would help users navigate and come across a large variety of resources that they might have not found by simply searching.
- Improving the user interface - Changing the colour scheme and font type would make the website more appealing to a larger group of people.
- Deploying the application on Docker

Conclusion

In conclusion, through this paper, I constructed a web based database application to organise a large amount of research based data to help users access revenant research papers efficiently. In this paper, I describe the process required to construct a database and the steps taken to build the web application.