Otto-von-Guericke University

Faculty of Computer Science

Master's Thesis

Concept and Execution of an Analysis of Bibliographic Data in the OpenAlex Dataset using Apache Spark

Author:
Satish Khadka

February 27,2024

Supervisors:

Prof. Dr. rer. nat. habil.
Gunter Saake
Department of Computer Science
Otto-von-Guericke University
39106 Magdeburg, Germany

Dr.
Eike Schallehn
Department of Computer Science
Otto-von-Guericke University
39106 Magdeburg, Germany

Satish Khadka:
Concept and Execution of an Analysis of Bibliographic Data in the OpenAlex Dataset using Apache Spark
Master's Thesis, Otto-von-Guericke University, 2024.

Acknowledgement

I want to express my deepest gratitude to my supervisors, Prof. Dr. Gunter Saake and Dr. Eike Schallehn, for providing me with the opportunity to work on this insightful thesis project. I am immensely grateful for their invaluable guidance, encouragement, and support throughout the entire process of completing this thesis. Their expertise and insightful feedback have been instrumental in shaping this work.

Special thanks to my family for their love, admiration, and unwavering belief in my abilities. Their support and encouragement have been my source of strength and motivation.

I would also like to thank my friends and colleagues for their encouragement, advice, and meaningful discussions, which have provided me with inspiration and motivation throughout this journey.

Abstract

In an era dominated by the rapid growth of digital information, analyzing bibliographic data has become crucial to understanding the patterns, impact, and evolution within the scholarly realm. This thesis delves into the conceptualization and execution of an analysis of bibliographic data, utilizing the OpenAlex dataset and the computing capabilities of Apache Spark. The rich and comprehensive collection of scholarly publications in the OpenAlex repository makes it a valuable resource for exploring patterns, trends, and insights within the scholarly domain.

The study begins by establishing the foundations, beginning with a description of publications and the publication process. The context then transitions into the overview of Bibliometric Analysis(a quantitative method used to analyze various aspects of bibliographic data), exploring various bibliographic metrics and challenges of Bibliometric Analysis. The concept of Big Data and its characteristics have been presented to show its resemblance with bibliographic data. Finally, the groundwork concludes with a brief description of Apache Spark, the key components and the benefits of Apache Spark, and its capabilities that make it a robust tool for Bibliometric Analysis.

A dedicated chapter reveals the comprehensive details of the OpenAlex dataset, providing an in-depth exploration of its entities, storage mechanism, and the procedures for acquiring the dataset. The chapter further details the data integrity issues faced during the analysis phase of this thesis work.

The implementation phase elaborates on detailed insight into data preprocessing, optimization, and the application of bibliographic metrics that uncover trends within publications; investigate citation and self-citation patterns; reveal authorship collaborations; explore the interconnection between citations, publications, and authorship; identify dominating concepts; assess the contribution of sources, publishers, and institutions; and evaluate the h-index. Each section in the implementation phase offers a discussion that interprets the findings and provides a possible explanation behind the nature of the result.

Finally, the thesis concludes by reflecting on the contributions made, acknowledging the limitations, and suggesting areas for further research work.

In conclusion, this thesis contributes to the area of bibliometric research by harnessing the capabilities of Apache Spark for conducting in-depth analysis of the bibliographic data within OpenAlex. The findings of this work not only enrich the understanding of the publication realm but also highlight the significance of robust and efficient tools in extracting valuable insight from bibliographic data. The methodology and insight presented in this thesis serve as a valuable resource for future Bibliometric Analysis.

Contents

List of Tables

List of Figures

List of Code Listings

List of Abbreviations

Acronym	Meaning
API	Application Programming Interface
SQL	Structured Query Language
MLib	Machine Learning Library
ETL	Extract, Transform, Load
RDD	Resilient Distributed Dataset
TM	Text Mining
WoS	Web of Science
MAG	Microsoft Academic Graph
FoS	Field of Study
REST	Representational State Transfer
JSON	JavaScript Object Notation
CSV	Comma-Separated Values
WORM	Write Once Read Many
JVM	Java Virtual Machine
APC	Article Processing Charge

1 Introduction

The volume of scientific publications is constantly expanding. It is crucial to extract valuable insights from the vast amount of scholarly literature available. With the increasing volume of publications, it is necessary to comprehend how they are produced, shared, and linked to each other. Bibliometric Analysis is a research methodology that guides researchers to explore the world of scholarly publications.

The span of bibliographic data today is massive. The publication volume has risen steeply from less than 1 million papers published in 1980 to more than 7 million in 2014 (Fire and Guestrin [5]). This increasing number of publications also necessitates the importance of maintaining quality, reliability, and impartiality in published articles. *Peer review* is a process that ensures scholarly works meet the quality standards before they are published by subjecting them to the scrutiny of experts in the same field (Kelly et al. [6]). Despite the absence of any viable alternatives to peer review in the contemporary digital era (Kelly et al. [6]), the digital revolution that prompted the mass publication of scholarly articles has led the process into facing criticism with its slowness, biases, and lack of transparency a few of many).

In order to accelerate the publication process and cope with the mass production of digital articles, the concept of *open-access* was introduced. Since journals today are no longer restricted by page limits, the need to restrict publication based on their mainstream ideas and novelty is no longer required. In other words, valuable works of authors are given the opportunity while providing the guarantee of scientific quality (Walker and Rocha da Silva [7]). On the one hand, it offers authors greater flexibility for publishing their works. On the other hand, readers are provided with free unrestricted access to scholarly

articles, promoting rapid dissemination of knowledge and facilitating wider research opportunities.

While the publication patterns continue to evolve and expand, the metrics used to gauge academic success have primarily remained unchanged (Fire and Guestrin [5]). The shifts towards open-access journals and online repositories have revolutionized the dissemination of publications. Nevertheless, the fundamental criteria for assessing scholarly impact continue to revolve around traditional metrics like citation counts, h-index, and g-index. These metrics are major in evaluating and ranking authors and their publications, allocating resources, and issuing funds. However, The competitive nature of academia has led researchers to strategically target and manipulate citation-based metrics through tons of research outputs and self-citation (Fire and Guestrin [5]). Therefore, shifting the research prioritization more towards quantity than quality.

Analyzing the bibliographic data constitutes the foundational aspect of this thesis. Through rigorous analysis, this thesis aims to discover insights about research productivity, collaboration trends, and the impact of scholarly work.

1.1 Aim of the Thesis

This thesis analyzes the bibliographic data using the openly accessible data provided in the OpenAlex dataset. A detailed exposition regarding the dataset's intricate structure and the comprehensive procedures for locally acquiring it has been provided in Chapter 4 of the thesis. The key objective of the thesis can be described as follows:

- **Understanding publication trends:**

  Monitoring the frequency of publication over time enables researchers to follow up on the advancement of the particular subject of interest or to identify transformation in the publication pattern as a whole. The idea here is to track the progression of different types of publications over time and record any following changes.

- **Citation impact analysis:**

  The process of distributing knowledge shared within scholarly works is through citation. When a work is cited within another literature, it signifies a connection between them. This work unveils the impact of

citation and how the citation is influenced by factors such as field of study, number of authors, and accessibility of publications.

- **Academic collaboration:**

  Collaboration refers to the practice of two or more like-minded researchers cooperating to create scholarly works. In such collaborative works, each author presents their ideas and contributes to the design, analysis, and execution of the research study. Analyzing the collaborative trend and studying how the publication pattern has shifted over time through such an alliance is paramount.

- **Assessing the relationship between publications, authorship, and citations:**

  Analyzing the relationship between publications, authorship, and citations offers multi-dimensional benefits in understanding scholarly communication and academic impact. By examining citation patterns associated with authors and their publications, researchers can assess the impact of their work, and identify influential authors. Similarly, assessing publications associated with authors reveals the productivity of authors.

- **Identifying dominating research fields:**

  Identifying dominating subjects within a research domain holds significant importance as it provides insights into the current trends, influential areas of study, and emerging fields of interest. This thesis undertakes a detailed assessment of the dominating research fields, leveraging both publication count and citation count as key metrics. The goal is to uncover subjects with the most significant influence and impact.

- **Unraveling the Influence: Institutions, Sources, and Publishers in Scholarly Publishing:**

  Tracking the publication of various institutions, sources, and publishers provides an understanding of their contributions to disseminating knowledge worldwide. These entities shape the landscape of scholarly publishing, influencing the impact, and accessibility of research outputs.

- **Assessing the h-index:**

  The h-index is a quantitative metric that reflects both the productivity and citation impact of an individual. It is a fundamental metric for assessing the scholarly impact of authors. This thesis aims to explore and analyze the h-index of authors involved in the field of Computer Science.

1.2 Research Questions

This thesis primarily centers on the investigation of answers to the subsequent research questions.

Research Question 1: What preprocessing strategies and optimizations can be employed to enhance the efficiency of bibliographic data analysis? How can the bibliographic metrics be implemented using Apache Spark?

To answer this query, various data preprocessing steps, such as the removal of duplicate data, flattening of the data, and changing the data formats have been explained in detail. Different file formats have been studied and compared to determine the most optimal one for enhancing the processing efficiency of Apache Spark. Additionally, comprehensive detail has been provided regarding the Spark Application Programming Interface(API) utilized for the context of this thesis work, along with the procedure optimization, resource allocation, and code execution undertaken for improving efficiency. Furthermore, the implementation phase has been described in detail including discussions related to the nature of the results. Snippets of code have been provided in every section that is involved in the calculation of bibliographic metrics to demonstrate how the capabilities of Apache Spark can be leveraged for the implementation of bibliographic metrics.

Research Question 2: How is the publication pattern affected by the accessibility of publications, authorship, and what is the impact on citation?

To address this question, the initial step involves providing an overview of the total number of publications per year and the total number of authors per year. Subsequently, an analysis is conducted to ascertain the influence of citations, specifically, the total count of citations each year.

A more in-depth investigation is carried out that aims to establish the relationships among publication frequency, authorship, and citation impact and how they interplay. For this purpose, initially, the interaction between the publications, authorship, and citations is observed. Following that, the total number of publications and the change in citation patterns between accessible publications and other types of publications are analyzed. Similarly, another comparative analysis is conducted to delineate the disparity between works published by individual authors and those produced through collaboration, alongside the shifts in citation trends associated with these two different forms of authorship.

Research Question 3: How does self-citation vary over the years?

In response to this inquiry, self-citation trends are examined across the years. For every year, the total number of citations received during that specific year is computed, followed by determining the proportion of these total citations attributed to self-citations by authors. Additionally, within the dataset encompassing all self-citations, a comparative study is performed to establish the proportions of self-citations originating from the primary authors versus those stemming from coauthors.

Research Question 4: What are the dominant research fields, and how do they differ concerning the publication output and citation impact?

The question is initially addressed by identifying the top research fields based on the total number of published works within each. Subsequently, the same identification process is conducted based on the total number of citations received by each research field.

Following that, the research fields ranked as top according to their publication count undergo an analysis of the total number of publications each year within these fields. Similarly, the research fields ranked as top according to their citation count undergo a determination of the total number of citations earned each year within these fields.

Research Question 5: How has the publication trend evolved per year in terms of contribution by different sources, publishers, and institutions?

To address this query, the overall publication and yearly publication hosted are assessed across various sources. Subsequently, the total publication output distributed is evaluated concerning the publisher. Furthermore, a thorough analysis is conducted concerning the contribution to publications by various types of institutions and the yearly total publications by different institutions.

Research Question 6: How has the metric h-index changed over time among authors in Computer Science?

To provide a comprehensive response to this question, a thorough analysis is conducted involving the calculation of the h-index. This analysis focuses explicitly on authors operating within the domain of Computer Science. By examining the evolution of this metric over a span of time, valuable insights are gleaned into how scholarly impact and influence have transformed authors in the field of Computer Science.

1.3 Structure of the Thesis

The following list describes the structure of this thesis:

- **Chapter 2** explains the fundamentals for this work, such as the publication process, bibliographic metrics, introduction and characteristics of big data, and description of Apache Spark.
- **Chapter 3** describes relevant literature on the analysis of bibliographic data and related concepts.
- **Chapter 4** provides a detailed introduction to OpenAlex, the entities of OpenAlex, procedures for data acquisition, and the data integrity challenges encountered while analyzing the dataset.
- **Chapter 5** details the procedures for data preprocessing and optimization. All the implementations undertaken and the discussion of generated results have been elucidated in this chapter. This chapter is dedicated to answering all the research questions previously mentioned in Section 1.2
- **Chapters 6** presents an overview of the study's findings and brings the thesis to a close, offering essential observations and potential avenues for future research endeavors.

2
Background

This Chapter explains the concept of publication and the process for publishing, Bibliometric Analysis, Big Data, and Apache Spark. Section 2.1 explains publications and the step-by-step process for publishing a publication. Section 2.2 describes Bibliometric Analysis; metrics for performance analysis, such as publication analysis, citation analysis, and h-index; and the limitations of Bibliometric Analysis. Similarly, Section 2.3 elaborates on Big Data and the characteristics of Big Data. Section 2.4 details the similarities between bibliographic data and Big Data. Section 2.5 describes Apache Spark, the components of Apache Spark, the benefits of Apache Spark, and Apache Spark as a tool for Bibliometric Analysis.

2.1 Publication Process

2.1.1 Publications:

Publication encompasses a broad range of scholarly works that contribute to the distribution of knowledge. They play a significant role in offering researchers a means through which they can share their findings. There are various types of publications.

1. **Journal:**

   A journal constitutes a repository for scholarly research works. These periodical publications come in various forms, including those that are dedicated to a specific topic or field. The peer-review process is a standard practice followed for publishing in journals to ensure the quality

and validity of the contents; while there are journals that follow different procedures for maintaining quality.

2. **Conference papers:**

   Conference papers are articles that are written by authors in order to present their ideas and discuss the results to the academic community. These papers outline the scope, methodology, and results of the research. Presenting a conference paper allows researchers to engage in live discussions with the audience and gain feedback. Conference papers generally tend to be short and are published in collections known as proceedings.

3. **Books:**

   Books are comprehensive written works that allow authors to share their ideas and delve into diverse perspectives. They are generally long, ranging from hundreds or even thousands of pages. Books cover a wide range of content but their main aim is to provide in-depth coverage of a topic. Given their comprehensive nature, books take longer to produce, unlike other articles.

4. **Thesis:**

   A thesis is a research paper that presents the findings of an individual and is commonly associated with the completion of a degree program. The primary purpose of the thesis is to demonstrate a candidate's ability to perform original research and communicate the findings. They are usually written under the guidance of a supervisor.

These represent only a subset of publications, additional publications such as patents, proceedings, and monographs, collectively contribute to the growth of knowledge across various disciplines.

2.1.2 Process for publishing papers:

According to the articles published by Pubrica Academy [1] and AIJR publisher [8], the publishing process is a lengthy and laborious process to transform a manuscript into a polished publication. It typically begins with the authors finding a journal with a similar goal and scope of the research and formatting the manuscript according to the guidelines of the journal. The next step involves submitting the manuscript to the journal. Upon submission, the

manuscript undergoes a rigorous peer-review process where the experts review the quality, originality, and validity of the work. Based on the review, the authors receive one of the three decisions.

- **Revision required:** The manuscript needs to be revised based on the provided feedback.
- **Decline:** The manuscript has been declined and thus needs to be improved and resubmitted.
- **Accepted:** The manuscript will be edited for publication.

After the acceptance, the author receives a galley proof version for minor proofreading modifications, and the paper will be published. The pictorial representation of the steps involved during the publication process in a journal is displayed in Figure 2.1.

The publication process for other types of publications, such as journals, books, or conference papers, shares similarities with publication in a journal. However, the process might vary depending upon the specific requirement of the publishers or sources; but the goal is to ensure the dissemination of accurate, high-quality information.

2.2 Introduction to Bibliometric Analysis

The landscape of academic research has been evolving so rapidly that the need to extract insight from the massive volume and diversity of scholarly publications has become crucial. Bibliometric Analysis is an approach that helps uncover the complexities of scholarly communication, collaboration, and knowledge distribution. It involves the examination of data associated with academic publications that allow researchers to uncover the relationships, patterns, and trends in academic publications. This approach not only addresses the challenges posed by the abundance of scholarly publications but also provides an opportunity to reshape the academic landscape by fostering a deeper understanding of the dynamics within academic fields and facilitating informed decision-making.

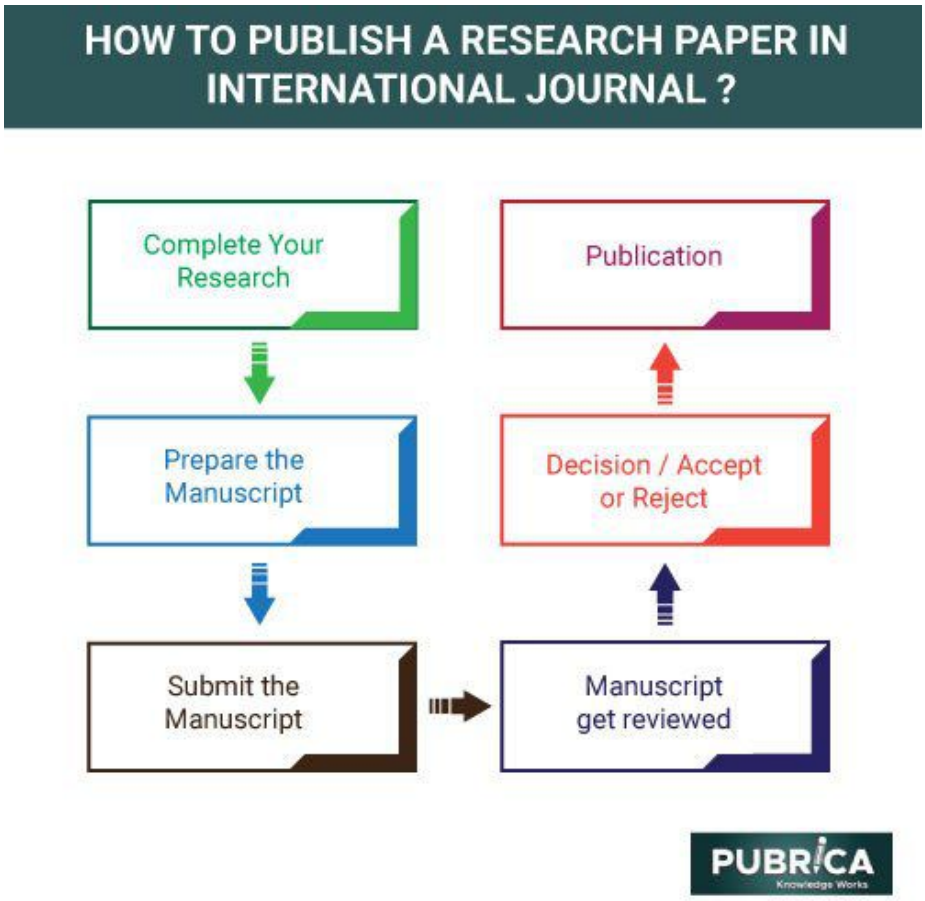


Figure 2.1: Pictorial representation of steps involved in publishing a journal (Pubrica Academy [1])

2.2.1 Bibliographic metrics

Metrics play a crucial role in bibliometric analysis. They provide quantitative measures that help the researchers assess the performance and the impact of scholarly publications. Some of the key performance analysis metrics are:

- **Publication analysis:**

  Publication analysis is not just a single metric but a wide range of metrics to evaluate scholarly publications. It involves the examination of several key metrics to assess the impact of publication:

1. **Total publication:**

   This metric depicts the overall proportion of publications associated with an author, journal, or institution. It is a general method of measurement of productivity and research output.

2. **Publication from authors:**

   Analyzing the publication from authors provides insight into the research productivity and collaborative pattern within the realm of publication. This metric can be further divided to assess the impact of publications by single authors and those produced through collaboration.

3. **Publication from institution:**

   Distribution of publications associated with various institutions helps assess the scientific contribution of research institutions. This metric involves examining the quantity and quality of scholarly works attributed to various institutions.

4. **Publication from sources:**

   It involves categorizing the scholarly publication based on different sources like Journals, Conferences, or Repositories and calculating the overall productivity. This metric helps the researchers identify the most influential and impactful source for publication.

In summary, these metrics offer a collective understanding of the scholarly activity. Analysis of the publications provides quantitative insight into the contributor and disseminator of the scholarly outputs.

- **Citation analysis:**

  Citation analysis is a method that is used to evaluate the impact and influence of publications. They are a very general measure of the level of contribution an individual makes to the practice of science (Garfield [9]). It is used to gauge the importance and visibility of publications, authors, and journals. The key metric involves counting the total number of citations that the publications receive. Although considered a valuable metric, citation analysis is not exempt from limitations. Some of the limitations of this metric include the following:

  – A high citation count could be produced as a result of criticism about a publication with low-quality work (Garfield [9], Purdue University [10]).

- Self-citation, a practice of an author citing their own previous publication. While the method is generally considered to be practiced to increase the citation on one's own publication, many scientists, however, tend to build on their own work (Garfield [9]).
- Citation count cannot be used to compare scientists in different fields (Garfield [9]).
- Some research fields tend to attract higher citations than others. For instance, most of the papers oriented to methodology in the chemical literature do not tend to be highly cited (Garfield [9], Purdue University [10]).
- Some scientific works receive rapid citations soon after getting published, whereas for others, it takes years to gain recognition (Purdue University [10]).
- Citation count is a measure of scientific activity and it does not say anything about the importance of scientific work to the advancement of science or society (Garfield [9]).
- Citation count provides an objective measure of the impact of scientific work. It does not explain the reason for the impact (Garfield [9]).

Despite the limitations, citation analysis remains a valuable tool for evaluating the research works and identifying influential works and authors.

- **H-index:**

  The h-index is a quantitative metric used to assess an individual researcher's scholarly impact and productivity. The metric was introduced in 2005 by Jorge Hirsh (Hirsch [11]). The main aim behind devising this metric was to solve the problem of overreliance on a single indicator such as the total citation count that might not fully represent the influence of an author. The h-index is determined by counting the total number of author's publications(h) that have been cited at least h times. "A scientist has index h if h of his or her N_p papers have at least h citations each and the other (N_p - h) papers have ≤ h citations each" (Hirsch [11]). For instance, an h-index of 10 means that the author has 10 publications with at least 10 citations. This means that the h-index is insensitive to papers with very low citations, as well as the ones that are severely cited (Bornmann and Daniel [12]). Figure 2.2 provides a visual representation for the calculation of the h-index.

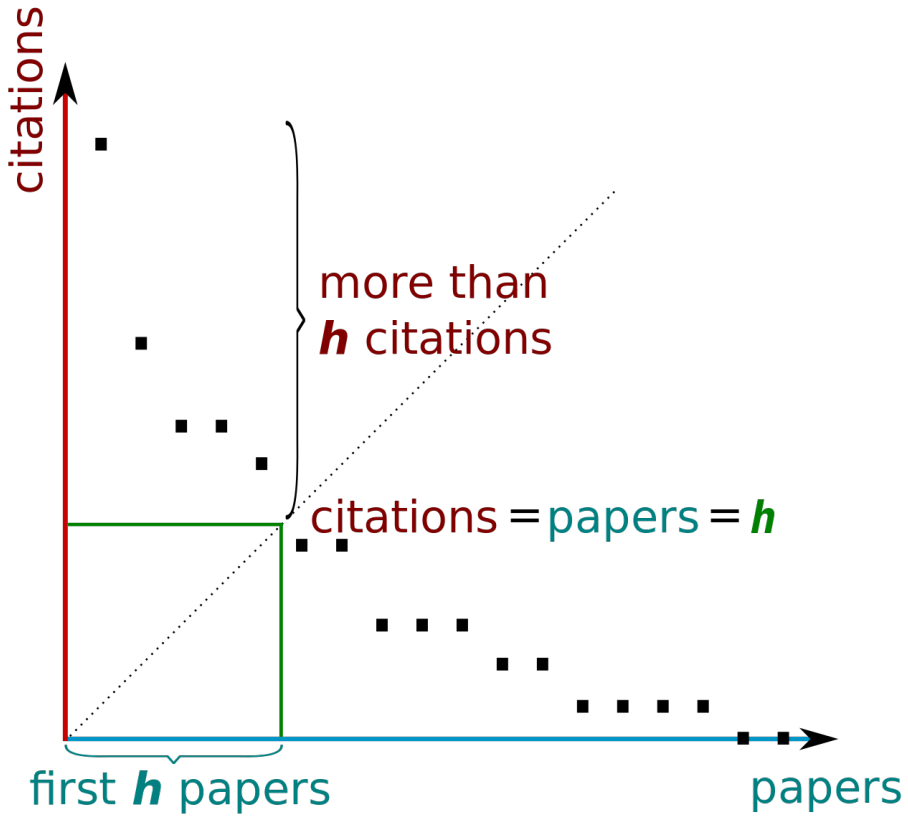


Figure 2.2: H-index determined from a graphical representation of an author's papers sorted in descending order (Wikipedia [2])

The applications of the h-index however, reach beyond the assessment of individual researchers. It can be applied for instance to calculate the h-index for a research group in a university or even for evaluating the scientific impacts of journals (Bornmann and Daniel [12]). Despite its advantages, the h-index does have various shortcomings.

– The researchers cannot be assessed based on a single measure. Several indicators are necessary to shed light on different aspects of performance (Bornmann and Daniel [12]).

- The h-index does not differentiate between active and inactive scientists. It fails to discern between the works from the past and trendy works which puts the newcomers at a disadvantage (Bornmann and Daniel [12]).
- Self-citation can have an impact during the calculation of the h-index as researchers tend to cite their own previous publications.
- The h-index can only be used to compare the performance of the scientists within the same discipline. The number of citations received varies based on discipline, therefore it would be only meaningful to assess the performance of scientists if they contribute to the same discipline.

In spite of its drawbacks to certain factors, the h-index remains widely used and accepted in the scientific community. The popularity of the h-index can be attributed to its simplicity of calculation and advantage over other bibliometric measures; due to which it offers as an evaluative measure for assessing the research output of scientists (Bornmann and Daniel [12]).

2.2.2 Limitations of Bibliometric Analysis

Bibliometric Analysis plays a crucial role in academic research. It allows the tracing of new ideas, evaluation of the impact of research productivity, and assessing the impact of authors or publications. However, it is essential to recognize the limitations of Bibliometric Analysis.

- **Selecting relevant repository:**

  There are many bibliographic repositories to choose from for conducting a Bibliometric Analysis. Each of the repositories offers distinct insight into the scholarly publication and research impact. Nonetheless, the challenge lies in selecting the appropriate repository because different repositories have different levels of quality, detail, and coverage regarding the total publications or citation counts, and not all repositories are completely accessible to all users. Researchers must carefully understand the strengths and limitations of repositories to ensure accurate analysis (Ferrara and Salini [13], Romanelli et al. [14]).

- **Selecting search term:**

  The process of conducting Bibliometric Analysis begins with identifying the context of interest such as "sustainable energy". This context is then broken down into "building blocks" or concepts that are transformed into search terms. Constructing an effective search string involves combining these search terms to maximize the retrieval of relevant results while minimizing irrelevant ones. For instance, the search string for "sustainable energy" can be "renewable energy" or "energy efficiency" (Romanelli et al. [14]).

- **Defining the timestamp:**

  Initiating a bibliometric study involves defining the timestamp. Based on the objective of the study, boundaries can be set to include only the years of publication for which the analysis is to be carried out. However, it is also crucial to note that the research trends often shift substantially depending on the timestamp (Romanelli et al. [14]).

- **Multidimensional analysis:**

  "In order to achieve meaningful results from Bibliometric Analysis, bibliographic data features cannot be taken into account separately. Bibliometric analysis must be supported by a multidimensional data model" (Ferrara and Salini [13]).

- **Duplicate detection and data normalization:**

  "Bibliographic data often contain multiple references to the same objects. Data must be cleaned, normalized, and disambiguated to the end of Bibliometric Analysis" (Ferrara and Salini [13]).

2.3 Big Data

Big Data refers to the vast volume of structured and unstructured information that is generated at an exceptional speed (Google Cloud [15]). It is a term for massive data sets characterized by their size, diverse and complex structures with difficulties in terms of storage, analysis, and visualization for further processes or results (Sagiroglu and Sinanc [16]). The volume of datasets encompassed by Big Data is huge; ranging from terabytes to petabytes which the traditional data processing methods struggle to handle. Big Data originates

from various sources; some of the common ones include social media, transaction records, and customer databases. Big Data is characterized by three main components:

- **Volume:**

  Volume describes a huge amount of data that is constantly being collected from different sources and devices (Google Cloud [15]).

- **Variety:**

  Big Data originates from a variety of sources and is typically categorized into three types: structured, semi-structured, and unstructured. Structured data is organized and can be easily sorted in a data warehouse, while unstructured data is random and challenging to analyze. Semi-structured data does not adhere to fixed fields but includes tags to distinguish data elements (Sagiroglu and Sinanc [16]).

- **Velocity:**

  Velocity refers to the rate at which the data is generated. In the present day, data is frequently produced in real-time or near real-time. Consequently, it needs to be processed, accessed and analyzed at a matching speed to make any significant impact (Google Cloud [15]).

Big Data is being used more often across various industries, transforming the way organizations operate and make decisions. Utilizing the power of Big Data helps organizations extract valuable insights and make data-driven decisions.

2.4 Bibliographic Data as Big Data

Bibliographic data encompasses extensive datasets including details like the title of publication, author's name, date of publication, citation counts, and subject classification. In this context, bibliographic data can be considered as Big Data. Moreover, bibliographic data aligns with the three key characteristics of Big Data:

- **Volume:**

  Just a few decades ago, the volume of bibliographic data was very modest. However, due to the rise of digital publishing and internet accessibility,

the growth of bibliographic data has become exponential. The repositories now host an extensive number of publications, articles, books, and related scholarly works that are generated worldwide.

- **Variety:**

  Bibliographic data includes diverse formats, languages, and schemas. While bibliographic records often appear structured with standardized metadata, the metadata standards vary across various sources. Moreover, the inclusion of various types of publications like books, conferences, and journals and diverse subject classifications from humanities to sciences and beyond further contribute to the diversity of bibliographic data.

- **Velocity:**

  Bibliographic data are generated at a rapid speed as the publications are produced continuously. New publications or articles are added to the database on a regular basis to keep up with the flow of information.

In summary, bibliographic data is consistent with the dimensions of Big Data, since it comprises a massive volume of information, being generated at high velocity, and including diverse types of data, thus making it a valuable resource to derive meaningful insight.

2.5 Introduction to Apache Spark

The rise of Big Data posed significant challenges in terms of processing and analyzing these massive datasets efficiently. Traditional data processing frameworks struggled to keep up with the volume and complexity of Big Data, thus motivating the need for powerful tools that could address the requirement of Big Data analysis. Apache Spark emerged as one such solution, offering a distributed computing framework designed specifically for Big Data processing tasks.

Apache Spark is a powerful tool that helps analyze large-scale data. It comes with built-in modules for SQL, streaming, machine learning, and graph processing ([Google Cloud] [17]). It is an open-source, distributed computing system that provides a fast and general-purpose cluster-computing framework for Big Data processing. Since its development in 2009, Apache Spark has been adopted by enterprises across a wide range of industries; such as Netflix, Yahoo, and eBay ([Databricks] [18]).

The Spark ecosystem includes five key components ([Google Cloud] [17]):

1. **Spark Core:**

   It is a general-purpose, distributed engine for processing data. It plays a crucial role in memory management, recovering from faults, scheduling, distributing, and monitoring jobs, as well as interacting with storage systems.

2. **Spark Structured Query Language(SQL):**

   It is a Spark module for handling structured data. This module enables querying of structured data within Spark programs, using either SQL or a familiar DataFrame API.

3. **Spark Streaming:**

   Scalable and fault-tolerant streaming solutions are made easy with Spark Streaming. The Spark language-integrated API is brought to stream processing, allowing streaming jobs to be written in the same manner as batch jobs.

4. **Machine Learning Library (MLib):**

   MLib is Spark's scalable machine learning library that is equipped with tools that facilitate the scalability and simplicity of machine learning. It contains, numerous common learning algorithms, including classification, regression, recommendation, and clustering.

5. **GraphX:**

   GraphX is the Spark API for graphs and graph-parallel computation. It is characterized by its flexibility and seamless compatibility with both graphs and collections, integrating extract, transform, load (ETL), exploratory analysis, and iterative graph computation within a single system. In addition to a highly flexible API, GraphX includes a variety of graph algorithms.

2.5.1 Benefits of Apache Spark

Apache Spark offers several benefits that make it a popular choice for large-scale data processing.

- **Speed:**

  Spark's in-memory processing capability makes it "100 times" more efficient than Hadoop MapReduce for large-scale data processing ([Databricks] [18]).

- **Ease of use:**

  Spark has easy-to-use APIs including a collection of over 100 operators for transforming data and familiar data frame APIs for manipulating semi-structured data ([Databricks] [18]).

- **Generality:**

  Spark comes packaged with higher-level libraries, including SQL and DataFrames, MLlib for machine learning, GraphX, and Spark Streaming. These libraries can be combined seamlessly in the same application. ([Google Cloud] [17], [Databricks] [18])

- **Fault Tolerance:**

  The Resilient Distributed Datasets (RDDs) in Spark automatically recover lost data partitions in case of node failures, ensuring fault tolerance.

These mentioned benefits are only a few advantages that Spark offers. There are numerous other reasons that contribute to the widespread use and effectiveness of Spark for handling big data.

2.5.2 Apache Spark for Bibliometric Analysis

Apache Spark is a robust tool for analyzing Big Data, and as explained in section 2.4, bibliographic data falls under this category. Given the massive volume of bibliographic data, Spark's speed and scalability are well-suited for managing and analyzing such data. Its in-memory processing capability and built-in module for SQL, machine learning, and graph processing facilitate diverse analysis, including citation analysis, authorship patterns, or trend analysis. Apache Spark provides the necessary tools to analyze and extract meaningful insight from the massive volume of bibliographic data.

3
Related Works

In this chapter, the focus is on providing an exploration of works related to Bibliometric Analysis. The chapter is organized as follows: Section 3.1 encompasses reviews of relevant literature that delve into the methodologies and usage of Bibliometric Analysis and Section 3.2 explains literature about performing Bibliometric Analysis exploiting the OpenAlex database and comparing the OpenAlex database to other similar databases.

3.1 Research Studies Exploring Bibliometric Analysis

3.1.1 Methodologies and usage of Bibliometric Analysis

A study was conducted by Nagarkar and Kumbhar [19] to analyze the Text Mining(TM) literature within the "Information Science Library Science" subcategory, utilizing the Web of Science(WoS) database. The primary focus was to understand the chronological evolution of TM literature. However, moving beyond the temporal consideration, the study investigated the contributions of countries, institutions, and departments and analyzed collaboration patterns. Additionally, the investigation evaluated the ranking of journals, institutions, and departments based on TM research productivity, with a specific emphasis on identifying highly cited journals and authors. The analysis was performed using Microsoft Excel and HistCite software for data analysis; Pajek and VoSviewer were used for data visualization.

An article by Stefaniak [20] depicted the usage of bibliographic databases for scientometric studies by reviewing the existing literature about the diverse applications of bibliographic databases in scientometric research. It explored the identification of leading journals in specific fields; analysis of particular

disciplines, including trend analysis and forecasting; and the contribution of countries to scientific literature. Furthermore, the paper also acknowledged the potential limitations of bibliographic databases such as coverage issues, time delays, and inconsistencies in data presentation, thus, providing an overview of both the capabilities and limitations of using bibliographic databases.

A more detailed study was conducted by Nguyen et al. [21] to examine the bibliographic features and content of articles on education published in Scopus-indexed journals by authors with Vietnamese affiliations between the years 2009 and 2018. The search was carried out using the Scopus database. The search option "affiliation country" was used with "Vietnam OR Viet Nam" as the country name, and the "subject area" was specified as "social sciences" and "education". Lastly, the search was narrowed to the period from 2009 to 2018, with the document type limited to articles. The analysis involved calculating the annual publication count of education articles for the last 10 years. Additionally, the study investigated the countries where coauthors of Vietnamese researchers held affiliations, identified the higher education institutions and research institutes in Vietnam that had the most article publications, and identified the Scopus-indexed journals that published papers on education by Vietnamese researchers. Finally, the frequently used keywords in the articles were highlighted and quantity of articles across various subfields of education were explored.

The aforementioned studies provide granular insight into conducting Bibliometric Analysis; emphasizing a specific field of study or a country. In contrast, this thesis provides a comprehensive overview, offering both a specific and more generalized understanding of the Bibliometric Analysis as a whole. The aim is to provide a more holistic perspective and deeper comprehension of the concepts and methodologies used in Bibliometric Analysis.

3.2 Studies on OpenAlex Database

3.2.1 Bibliometric Analysis using OpenAlex database

A research work by Krause and Mongeon [22] aimed to investigate the relationship between dataset creators and authors referencing the dataset at individual, institutional, and national levels utilizing the data set citation in the OpenAlex database. The research was conducted by accessing the data through a snapshot of OpenAlex that contained 211 million works of which only 531299 were

identified as datasets. The objective of the work was to examine the distribution of dataset citations and determine the proportion of self-citations. Further, the frequency with which the datasets were cited by authors affiliated with the same institution and country was explored and the patterns in the citation of datasets across various institutions and countries were identified.

In a project utilizing the OpenAlex API, Schares and Mierz [23] gathered, refined, and analyzed a year's worth of open cited reference data from publications authored by Iowa State University. The process was automated using Python and Jupyter Notebook. The objective was to investigate the total and per-publication count of references, the journals and publishers that were cited, the year of publication of the cited articles, and the average age of references per publication.

Similar to the papers reviewed in this section, this thesis work also adapts the approach of utilizing OpenAlex for Bibliometric Analysis. Based on the successful utilization of the database in Bibliometric Analysis as conducted in the studies above, the decision to incorporate OpenAlex aligns with the purpose of executing a broader analysis in this thesis.

3.2.2 Comparison with other similar databases

After the discontinuation of the Microsoft Academic Graph(MAG), OpenAlex incorporated all the data from MAG, except for the patent data. A comparative analysis of OpenAlex with MAG was performed by Scheidsteger and Haunschild [24] restricting the analyses to publication years before 2021. The results revealed that more than 90% of MAG documents align with equivalent document types in OpenAlex. Additionally, OpenAlex hosts a higher number of documents classified into 26 document types unlike 7 in MAG as OpenAlex inherits its data from another major data source, Crossref. Furthermore, a proportion of documents in MAG and the Field of Study(FoS) they belong to have been reclassified in OpenAlex. The research concluded that OpenAlex is better suited for Bibliometric Analysis than MAG due to the broader coverage of document type assignments.

In another comparative study, Jiao et al. [25] investigated the indexing of 18 exclusive data journals (journals primarily dedicated to data papers) in four popularly scholarly databases: Web of Science, Scopus, Dimensions, and OpenAlex. The primary goal of the research was to assess the coverage and accuracy of indexing the journals and their papers within the database. The procedures

involved finding the data journals, collecting the metadata information of publications within each journal from the databases, and comparing how the papers are classified into different document types by different databases. The findings reveal that Dimensions and OpenAlex stand out in terms of comprehensive coverage of publications. Only WoS and Scopus define a distinct document type for data papers, however, the assignment of document type is very inconsistent between both databases.

The comparative analysis of the OpenAlex database with other similar databases studied in the papers above emphasizes that the coverage provided by OpenAlex surpassed that of other databases. This notable advantage in comprehensiveness becomes a rationale for using OpenAlex in the framework of this thesis work. Given its extensive coverage, OpenAlex is positioned as a valuable resource for thorough analysis of scholarly articles, which aligns with the goal of this thesis.

4

Overview of the OpenAlex

This chapter serves as an introductory overview of OpenAlex. In Section 4.1, a detailed description of OpenAlex, including the motivations behind its establishment as a freely accessible repository, along with the diverse entities within OpenAlex are presented. Section 4.2 explains the data storage mechanism in the OpenAlex. Section 4.3 details the steps for acquiring the OpenAlex data. Finally, Section 4.4 delves into the complexities of data integrity challenges encountered during dataset analysis.

4.1 Introduction to OpenAlex

Since its introduction in 2015, Microsoft Academic Graph (MAG) had emerged as a powerful and comprehensive resource for Bibliometric Analysis (Scheidsteger and Haunschild [24]). Due to its vast coverage of scholarly literature, many researchers exploited it to conduct their investigation, comparison to other databases being one of many (Scheidsteger and Haunschild [24]). However, a notable turning point occurred in May 2021, when Microsoft announced discontinuing support for MAG (Priem et al. [3]), (Scheidsteger and Haunschild [24])). To address the growing concern about finding an adequate replacement for such a rich and diverse data source, a nonprofit organization, OurResearch[1], introduced the OpenAlex project (Priem et al. [3], Scheidsteger and Haunschild [24]).

Launched in January 2022, OpenAlex emerged as a free and open repository encompassing the breadth of scholarly literature, researchers, journals, and institutions, Along with the mapping of their interconnections (Piwowar et al. [26]). This repository facilitates the construction of scholarly search engines

[1] OurResearch: `https://ourresearch.org/`

and recommender services and empowers research management through citation tracking and identification of emerging topics (Piwowar et al. [26]). Although considered a free alternative to other large-scale, subscription-based repositories like Scopus and Dimensions (Piwowar et al. [26]), the data coverage of OpenAlex is second to none (Akbaritabar et al. [27], Ortega and Quirós [28])

4.1.1 OpenAlex schema and entities

It is crucial to highlight that the details concerning the counts of entities within OpenAlex and the corresponding indices assigned to each entity are derived from the most recent OpenAlex dataset obtained for this thesis. The dataset was procured on March 16th, 2023, ensuring that all subsequent insight conforms to the data accessible then.

The OpenAlex dataset constitutes a diverse directed graph, combining six distinct categories of scholarly entities interconnected through their associations (Priem et al. [3]). The outline of the graph is shown in Figure 4.1.

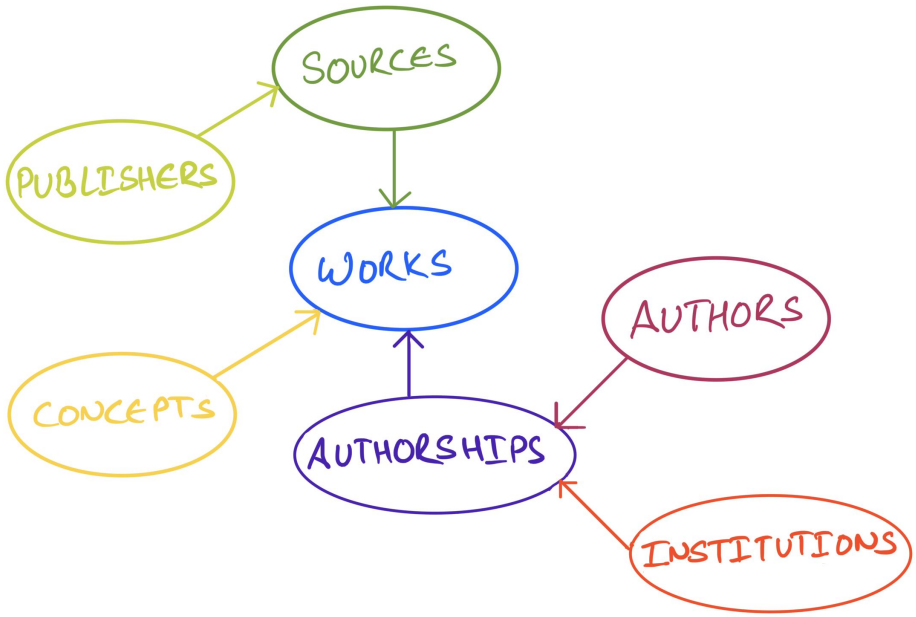


Figure 4.1: Outline of OpenAlex graph data model (Priem et al. [3])

The foundation of OpenAlex comprises the following entities:

- **Works:**

  Work refers to a discrete piece of intellectual creation contributing to a specific field of study. Within the OpenAlex graph, the significance of works stands out, as it binds authors, institutions, publishers, sources, and concepts together (Priem et al. [3]). The OpenALex database contains approximately 248 Million works.

- **Authors:**

  Authors are the individuals who contribute significantly to the creation of work. They are responsible for designing the research, performing necessary experiments, interpreting the results, and writing the manuscript. There are about 102 Million authors listed within the OpenAlex repository.

- **Concepts:**

  Concepts refer to abstract or general ideas, theories, or frameworks, within which the research is based. They are the main foundation for work in OpenAlex. The concepts are organized hierarchically, with 19 primary concepts forming the root level, with 5 layers of dependents (Priem et al. [3]). There are about 65 Thousand concepts indexed in open Alex.

- **Institutions:**

  Institutions are established organizations that play a significant role in supporting and facilitating academic or research-related activities. There are 8 different categories of institutions listed by OpenAlex: education, health care, facility, government, company, nonprofit, archive, and others. The total number of institutions indexed by OpenAlex is about 108 Thousand.

- **Sources:**

  Sources are the place where works are hosted. They are the platform where authors can showcase, present, and share their works with the scholarly community. OpenAlex lists 4 types of sources: journals, ebook platforms, conferences, and repositories. The total number of different sources indexed by OpenAlex is about 226 Thousand.

- **Publishers:**

  Publishers are the intermediaries that act as a bridge between authors and scholarly communities. They are responsible for facilitating the dissemination of work and ensuring quality control. There are about 3 Thousand publishers indexed in the OpenAlex dataset.

4.2 Data Storage Mechanism in the OpenAlex

Openalex is a dynamic repository that serves as a valuable resource for researchers in the field of academic research. Understanding how the data are stored within OpenAlex is crucial for exploiting its functionality.

1. **Data collection and sources:**

   Academic research data are first collected from a variety of sources and then stored in OpenAlex. After the discontinuation of the Microsoft Academic Graph, all of its corpora were preserved and incorporated by OpenAlex (Scheidsteger and Haunschild [24]). Crossref constitute another primary source, and the other sources include ORCID, ROR, DOAJ, and Pubmed, to name a few (Piwowar et al. [26]).

2. **Data storage and structuring:**

   All data is securely housed within the Amazon S3 platform, residing in the designated "OpenAlex" bucket. These data files adopt a format of gzip-compressed JSON Lines, with each entity represented as a distinct row. Notably, the gzip-compressed snapshot consumes approximately 330 GB of storage, expanding to approximately 1.6 TB when decompressed. Within the bucket, there exists a specific prefix (or folder) corresponding to each entity type, including "Work," "Author," "Source," "Institution," "Concept," and "Publisher" (OpenAlex Documentation [4]).

3. **Data query and retrieval:**

   OpenALex offers an accessible and versatile interface for querying and retrieving data. The API serves as the predominant means of accessing OpenAlex data, and it comes at no cost, requiring no authentication. Users are allocated a daily limit of 100,000 requests per day for API calls. Additionally, the files are stored on S3, making it convenient to download the snapshot on the local machine using Amazon tools (OpenAlex Documentation [4]).

4. **Maintainance and updates:**

   To sustain the integrity and relevance of its data, OpenAlex undergoes consistent maintenance and quality assurance. The entities within OpenAlex undergo daily updates, incorporating approximately 50,000 new works and thousands of authors each day(Priem et al. [3]). Meanwhile, the snapshot undergoes a monthly update to ensure its accuracy (OpenAlex Documentation [4]).

4.3 Acquiring the OpenAlex Data

In this section, the procedures for obtaining the OpenAlex data are explored. There are multiple ways to acquire and exploit the OpenAlex data. The first is by using the Representational State Transfer Application Programming Interface (REST API) that OpenAlex provides and the second is by downloading the snapshot of the database on the local machine.

1. **Using the API:**

   The API serves as the primary way for accessing OpenAlex data. It offers free access without the need for authentication. Every user is allotted a limit of 100,000 requests per day. Using the API, each entity of the OpenAlex dataset can be accessed individually. The user can either get a single entity, lists of entities, or groups of entities as per the requirement. The data however are in the JavaScript Object Notation (JSON) Lines format (OpenAlex Documentation [4]).

2. **Downloading the snapshot:**

   The snapshot comprises six files, divided into smaller units for ease of use, each corresponding to one of the six entity types. The files are also in the JSON Lines format.

   The entire dataset is stored within the Amazon S3 storage, specifically in the "openalex" bucket. The most convenient method for obtaining the files is by installing the Amazon Web Services Command Line Interface (AWS CLI). The compressed file takes more than 300 gigabytes and will most likely change over time, therefore, the users must make sure that they have enough storage on their local machine to accommodate the file.

   After the download, the file structure in the local machine corresponds to the structure displayed in Figure 4.2. Furthermore, in order to keep

the data up to date, the users can exclusively download the updated data instead of downloading the whole snapshot again (OpenAlex Documentation [4]).

```
openalex-snapshot/
├── LICENSE.txt
├── RELEASE_NOTES.txt
└── data
    ├── authors
    │   ├── manifest
    │   └── updated_date=2021-12-28
    │       ├── 0000_part_00.gz
    │       └── 0001_part_00.gz
    ├── concepts
    │   ├── manifest
    │   └── updated_date=2021-12-28
    │       ├── 0000_part_00.gz
    │       └── 0001_part_00.gz
    ├── institutions
    │   ├── manifest
    │   └── updated_date=2021-12-28
    │       ├── 0000_part_00.gz
    │       └── 0001_part_00.gz
    ├── sources
    │   ├── manifest
    │   └── updated_date=2021-12-28
    │       ├── 0000_part_00.gz
    │       └── 0001_part_00.gz
    └── works
        ├── manifest
        └── updated_date=2021-12-28
            ├── 0000_part_00.gz
            └── 0001_part_00.gz
```

Figure 4.2: OpenAlex file structure (OpenAlex Documentation [4])

4.4 Data Integrity Issues

Open Alex is not exempt from common data quality challenges in bibliographic datasets. Some issues related to data reality faced during the thesis are:

1. **Inaccurate data**

   The data set in OpenAlex exhibits noticeable issues related to data accuracy, primarily marked by some portion of publications that have been erroneously assigned to false publication years, as shown in Figure 4.3.

This inaccurate assignment for the year of publication poses challenges to the data set's overall quality and its potential to offer comprehensive Insights into scholarly works.

id	display_name	publication_year
https://openalex.org/W4200400827	Conclusion	2101.0
https://openalex.org/W4200263325	Business Marketing	2050.0
https://openalex.org/W2979197763	Eightieth Anniversary Message	2081.0

Figure 4.3: Inaccurate data

2. **Duplicate data**

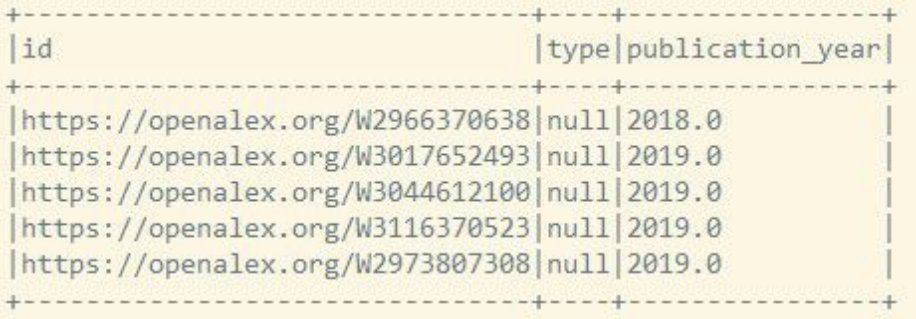
   Another challenge OpenAlex encounters is related to duplicate data, where specific publications and their corresponding authors are replicated within the dataset. Figure 4.4 depicts one such example. These occurrences of duplicate entries pose potential complications for accurate analysis.

work_id	author_id
https://openalex.org/W2995930366	https://openalex.org/A3214014075
https://openalex.org/W2995930366	https://openalex.org/A3214014075
https://openalex.org/W2995930366	https://openalex.org/A3214014075

Figure 4.4: Duplicate data

3. **Incomplete data**

   Openalex contends with the issue of incomplete data. The presence of Null values in the data set can hinder the overall comprehensiveness and undermine the reliability of the data set. An instance of this is illustrated in Figure 4.5.



id	type	publication_year
https://openalex.org/W2966370638	null	2018.0
https://openalex.org/W3017652493	null	2019.0
https://openalex.org/W3044612100	null	2019.0
https://openalex.org/W3116370523	null	2019.0
https://openalex.org/W2973807308	null	2019.0

Figure 4.5: Incomplete data

5

Implementations and Results

This chapter serves as the core of this thesis work, encapsulating all the details and methodologies involved during the analysis. Firstly, Section 5.1 provides an overview of methods and approaches adopted for the analysis of the OpenAlex dataset. Section 5.2 details the data preprocessing steps undertaken. Section 5.3 interprets the steps taken for optimization, resource allocation, and code execution for implementing the bibliographic metrics using Apache Spark. Finally, Section 5.4 provides a comprehensive representation of the implementation and discussion of bibliographic metrics implemented for this thesis work.

5.1 Conceptual Overview:

This section provides a conceptual overview of the methodologies and approaches employed in the analysis workflow for bibliographic data.

5.1.1 Data preprocessing steps:

The data preprocessing phase involved various steps aimed at preparing the bibliographic data for analysis. Firstly, the challenge of nested data was dealt with by flattening the dataset into a more structured format, Parquet format, to be precise. Importantly, the flattening process was carried out to the requirement of the analysis, ensuring optimized data for efficient processing.

Secondly, The file format was changed to enhance compatibility and efficiency. This step involved converting the file into a format that is well-suited for processing within Apache Spark.

Lastly, duplicate deletion was performed to remove redundant data and ensure data integrity.

The data preprocessing steps have been explained thoroughly in Section 5.2

5.1.2 Optimization and execution steps:

The optimization and execution steps highlight the procedure for optimization, resource allocation, and code execution undertaken. While the optimization and resource tuning phase describes the Spark API utilized for implementing bibliographic metrics and allocation of resources for boosting efficiency, the execution phase involves interpreting the step-by-step execution of code for conducting analysis. The in-depth procedure for optimization, resource tuning, and code execution has been explained in Section 5.3.

5.1.3 Implementation steps:

The implementation phase emphasizes analyzing bibliographic data to derive meaningful insight. This includes the analysis of publications; authorship trends; citation analysis; interconnections between publications, authors, and citations; self-citation trends; and the computation of the h-index.

For publication analysis, the trend in publications was analyzed for the period 1992 to 2022. A comparative analysis was performed to compare publications during the period 1981 to 2001 and 2002 to 2022. Similarly, the distribution of retracted publications and distribution of open-access publications was analyzed to identify the shift in publication trends.

Similarly, the analysis of authors involved investigating trends in authorship for the period 1992 to 2022. A comparative evaluation was carried out to determine the popularity of single-authored and multiauthored publications.

Citation analysis focused on examining the citation patterns for the years 1992 to 2022. The impact of accessibility and modes of authorship on citations were thoroughly analyzed. Additionally, the investigation of self-citation highlighted the patterns of self-referencing in the scholarly realm.

Likewise, the relationship between publications, authors, and citations was determined through four metrics: "average publications per authors", "average authors per publications", "average citations per publications", and "average citations per authors".

The analysis of research fields revealed the most dominating research field in terms of citations and publication counts. It underscored the importance of analyzing both metrics to get a clearer insight.

Moreover, the investigation of the contribution of institutions, sources, and publishers revealed the proportion of publications associated with each type of institution, source, and publisher, highlighting the most dominating type within each entity. Further, the number of publications per year associated with each type of institution and source was determined. However, due to the ambiguous results generated during the analysis of publishers, further investigations were avoided.

Finally, the h-index was calculated to assess the productivity and impact of authors within the field of Computer Science.

Section 5.4 provides more comprehensive coverage of the implementation phase explained above, including discussions regarding the behavior exhibited by the results.

5.2 Dataset Preprocessing

In this subsection, the procedures carried out for preprocessing the OpenAlex dataset have been explained. Each procedure has been described in distinct subsections below.

5.2.1 Flattening and splitting the nested data

The downloaded files are in the JSON Lines format and the data have been nested in order to represent a hierarchical relationship. Such nested format can become challenging to read and may cause problems while querying and filtering the data. For this reason, the nested data structure has been flattened to simplify the analysis and ease the querying. For instance, the nested structure of the work entity as displayed in Listing 5.1(edited the length for readability), has been flattened and split into two different files, however, retaining the ID of the entity in both files. The resulting files correspond to the structure as depicted by Listing 5.2.

One important thing to note here is that for this thesis work, the files were flattened as per the requirement of the analysis. For instance, to analyze the citations, the "work" file, one of the major files in the OpenAlex snapshot as

displayed in Figure 4.2, was split to contain only the metadata like publication ID, year of publication, referenced publications, that were relevant for the analysis. The reason for splitting in such a manner is because loading the whole file would take up more storage and processing time and since the analysis was carried out on a local machine, the processing and storage capabilities were limited, thus necessitating a selective approach to optimize storage and processing efficiency.


```
{
  "works": [
    {
      "id": "https://openalex.org/W2741809807",
      "doi": "https://doi.org/10.7717/peerj.4375",
      "title": "The state of OA: a large-scale
         analysis of the prevalence and impact of
         Open Access articles",
      "publication_year": 2018,
      "open_access": {
        "is_oa": true,
        "oa_status": "gold",
        "oa_url": "https://peerj.com/articles
           /4375.pdf"
      }
    }
  ]
}
```


Listing 5.1: JSON data format of the OpenAlex data


```
{
  "works": [
    {
      "id": "https://openalex.org/W2741809807",
      "doi": "https://doi.org/10.7717/peerj.4375",
      "title": "The state of OA: a large-scale
         analysis of the prevalence and impact of
         Open Access articles",
      "publication_year": 2018
```

```
    }
  ]
}

{
  "works_open_access": [
    {
      "work_id": "https://openalex.org/W2741809807
         ",
      "is_oa": true,
      "oa_status": "gold",
      "oa_url": "https://peerj.com/articles/4375.
         pdf"
    }
  ]
}
```

Listing 5.2: JSON data format of the OpenAlex data after flattening

5.2.2 Converting file format

The file format was transformed from JSON to CSV (Comma Separated Values), resulting in a shift from a hierarchical and nested data structure to a flat and tabular format. In the process of conversion, the JSON files, with their nested data, were flattened to represent information in rows and columns. The original field names in the JSON data were used as headers in the CSV file. This conversion led to a reduction in file size and improved readability

Upon further investigation, it was found that the Parquet file format is better suited for handling big data than the CSV format (MultiTech [29], Chopra [30]). Thus, the files were transformed from CSV to Parquet making a transition from a flat and tabular format to a highly optimized, columnar storage structure. Parquet's columnar storage improved compression and facilitated reduced and efficient storage. It is designed following the Write Once Read Many (WORM) model. While it may exhibit slower write speeds, its strength lies in exceptionally fast read performance (MultiTech [29]), which is perfect for the analysis tasks that are to be carried out in this thesis work.

A comparative analysis was conducted to analyze the processing times between CSV and Parquet format in the context of Apache Spark data processing. The

experiment involved loading the dataset about the work entity in both formats into the Spark DataFrames and using the *count* function to calculate the total number of publications listed in the OpenAlex repository. Figures 5.1 and 5.2 display the processing time taken by the CSV format and Parquet format respectively. While the CSV file took more than 25 minutes to process, the processing time for the Parquet file was about 24 seconds. The results indicate that Parquet outperformed CSV in terms of processing efficiency.

```
Total publications : 248431999

Process time for Parquet : 0:25:54.733220
```

Figure 5.1: Total processing time for CSV format

```
Total publications : 248431999

Process time for Parquet : 0:00:23.212857
```

Figure 5.2: Total processing time for Parquet format

5.2.3 Removing duplicates

Duplicate datasets refer to records that are identical to one another. It can occur when there is redundant information within the dataset. Handling the duplicates is an important step in data preprocessing. For this thesis work, the duplicate data are removed at the time of performing the analysis as can be seen later in code snippets provided in Section 5.4. The reasons for this are twofold. First, removing the duplicate entries and creating a new dataset would require additional storage space. Second, performing most of the analysis would require Join operations among the datasets which could further result

in the creation of duplicate values. Therefore, the duplicate data were handled whenever and wherever required.

It is essential to highlight that the elimination of duplicates in this context was achieved by comparing the IDs of the entities. For instance, if the analysis involved the examination of authorship, the data sharing both the same publication ID and the same author ID would be removed.

5.3 Optimization, Resource Tuning and Execution

Before delving into the actual implementation of the bibliographic metrics using Apache Spark, it is important to understand the API of Spark that has been utilized for carrying out the implementation. This section provides a detailed understanding of the API of Spark that has been leveraged, alongside resource tuning for effective analysis. Furthermore, the steps involved in the execution of the analysis have been explained.

5.3.1 Spark API and resource allocation

The implementation was carried out utilizing PySpark, a Python API for Apache Spark, highlighting that Python has been used as the primary programming language. The code leverages PySpark's DataFrame API, which structures the data into a tabular format as rows and columns. The DataFrame API allows users to work with data similar to working with tables in a relational database. It provides a rich set of functions like "filter", "join", "groupBy", "orderBy", and many more that are crucial for data transformation and analysis.

Further, optimizations were incorporated through tuning of resources within Apache Spark, such as setting driver memory and specifying execution mode and executor cores. For instance, to carry out the analysis of authors, the "local" mode of execution along with the allowance of utilizing all the available cores and a total of 14 gigabytes of driver memory has been provided. The snippet of the code is displayed in Listing. 5.3. The reason for providing only the driver memory is that in the local mode of execution, the executor is within the same Java Virtual Machine(JVM) as the driver. The resources have been allocated based on the specifications of the local machine. These optimizations aim to enhance the performance and efficiency of the Spark job execution.

```
# Importing the "SparkSession" class from the "
    pyspark.sql" module
from pyspark.sql import SparkSession
# Building a SparkSession and setting up resources
    for optimization
spark = SparkSession.builder.appName("Authors")/
          .master("local[*]")/
          .config("spark.driver.memory", "14g")/
          .getOrCreate()
```

Listing 5.3: Source code for distribution of publications and open accessibility

5.3.2 Execution steps

Several key steps were undertaken for executing the code to process and analyze the bibliographic data during the implementation phase. The implementation strategy along with the code snippet is explained thoroughly in Section 5.4. This section provides a step-by-step breakdown of the entire procedure for code execution.

Initially, A SparkSession was established, allocating resources as per the requirement of analysis. Following this, the relevant Parquet files were loaded into the Spark environment. The data integrity was assured by removing the duplicates from the dataset. Subsequently, to improve the performance, only the columns necessary for the analysis were extracted from the dataset. Throughout the execution, the DataFrame functions for filtering, joining, grouping, aggregating, ordering, and other similar functions were employed as needed. Finally, the results of the analysis were displayed. The majority of the execution in this thesis work followed these procedures, although certain tasks, like the analysis of the h-index, required a few additional steps. The pictorial representation of the steps is depicted in Figure 5.3. Furthermore, due to the processing constraints, all the graphs and charts displayed in Section 5.4, were constructed using Microsoft Excel.

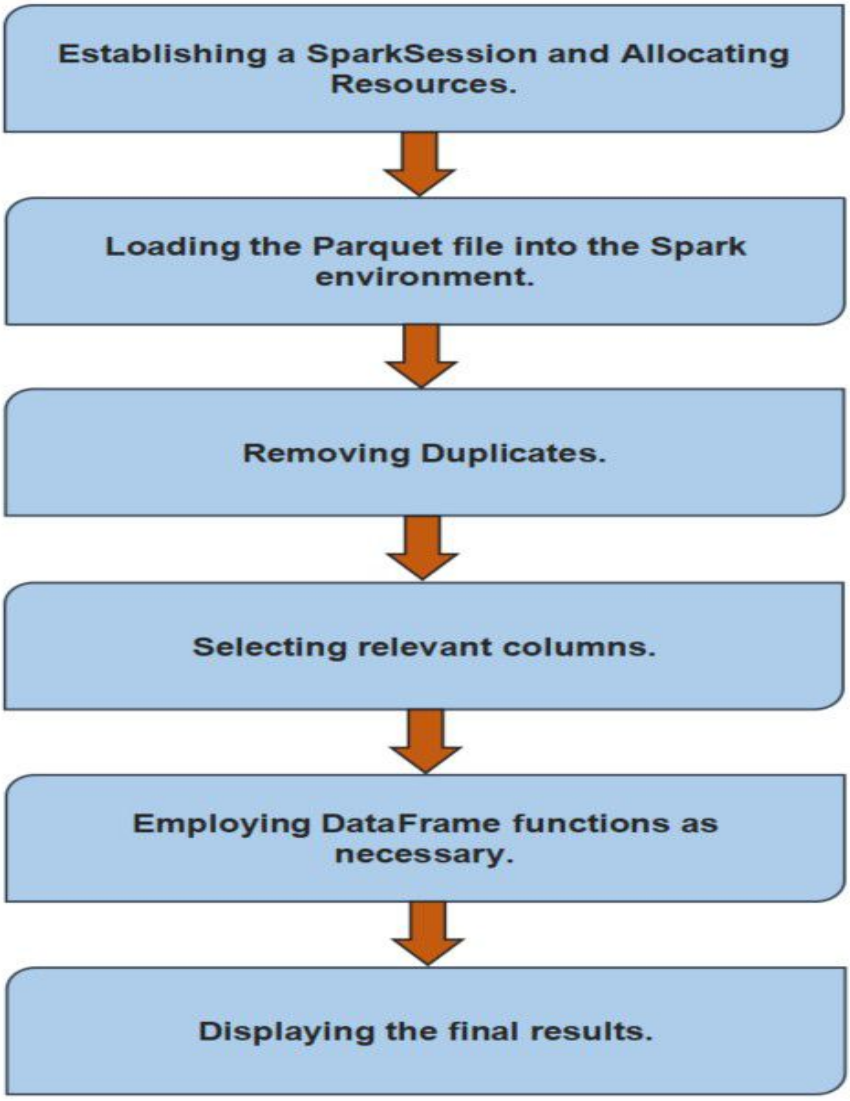


Figure 5.3: Pictorial representation of steps involved in the execution of codes during implementation.

5.4 Implementation and Discussion

In this section, a comprehensive explanation of all the implementations undertaken has been explained. Additionally, the sections contain snippets of codes to provide a clear understanding of how the implementation has been carried out. A dedicated section has been included after each execution of the bibliographic metrics discussing the potential cause for the characteristics of the results.

5.4.1 Distribution of publications:

A large number of scholarly papers are published each year. While all of the papers share a common ground that they are the outcome of intensive research, they belong to different categories of publication, such as Journals, Books, or proceedings. The accessibility of these published papers is dependent on factors like publication venue and choice of authors. Some of the papers are published such that they are openly accessible to all their readers, while others are found behind paywalls that require subscription payment.

Similarly, not every paper that has been published retains the same status; some of the papers eventually get retracted due to errors based on their methodology, reliability, and validity.

For this thesis work, the distribution of publication was analyzed based on two factors: distribution of publication among various types of publication along with the proportion of their open accessibility within each type and distribution of types of publication that have been retracted or are no longer available. Both of these topics are explained in the following subsections.

5.4.1.1 Distribution of publications and open accessibility:

OpenAlex lists about 26 different types of publications. The distribution of publications within these 26 publication types and the proportion of openly accessible publications ordered according to the publication count is represented in Table 5.3.

Implementation:

A data frame was generated, encompassing all the works and their respective types. The schema for this data frame is depicted in Table 5.1.

In the first phase, the datasets were grouped by the type of publications they represented, and the overall distribution of publications for each publication type was calculated using the count method. By dividing the distribution of each publication type by the total number of works, the proportion of each of these Publication types was computed.

Additionally, a separate data frame was employed for calculating the proportion of publications that are openly accessible. This data frame consisted of all the publications along with their open accessibility status, as illustrated in Table 5.2. The two data frames were joined together on columns *id* from the first data frame and *work_id* from the second data frame as key columns. Subsequently, the *is_oa* column was filtered to determine the open accessibility status of the works. Specifically, *is_oa* = *true* indicated that the publication was openly accessible, while *is_oa* = *false* indicated that the publication was not openly accessible. Finally, the dataset was grouped based on publication types, and the total number of openly accessible publications for each publication type was calculated. The proportion of openly accessible publications was determined by dividing the distribution of openly accessible publications for each publication type by the total number of publications. The snippet of the code is displayed in Listing 5.4.

Column Name	**Data Type**	**Description**
id	String	The OpenAlex ID for publication
type	String	Types of publication

Table 5.1: Data frame with publication ID and publication type.

Column Name	Data Type	Description
work_id	String	The OpenAlex ID for publication
is_oa	Boolean	Boolean value for open accessibility

Table 5.2: Data frame with publication ID and status of open access.

```
#reading first parquet file
    read_works = spark.read.parquet("D:\
        open_alex_parquet\works.parquet")
    #dropping duplicates if any
    filter_works = read_works.dropDuplicates(["id"
        ])
    #selecting only the required columns
    works = filter_works.select("id","type")
    #displaying the distribution of publications
        within publication types
    works.select("type").groupBy("type").count().
        orderBy(col("count").desc()).show(truncate=
        False)
    #reading second parquet file
    read_works_open_access = spark.read.parquet("D
        :\open_alex_parquet\works_open_access.
        parquet")
    #dropping duplicates if any
    filter_works_open_access = read_works.
        dropDuplicates(["work_id"])
    #selecting only the required columns
    works_open_access = filter_works_open_access.
        select("work_id","is_oa")
    #joining the dataframes
```

```
16    joined_works = works.join(works_open_access,
          works.id ==  works_open_access.work_id,"
          inner")
17    #displaying the distribution of openly
          accessible publications within publication
          types
18    joined_works.select("type").filter(lower(col("
          is_oa"))=="true").groupBy("type").count().
          orderBy(col("count").desc()).show(truncate=
          False)
```

Listing 5.4: Source code for distribution of publications and open accessibility

Document Type	Total Publication	Percentage of Total	Oplenly Accessible Publications	Percentage Openly Accessible
Journal-article	125354155	50.458	36262249	28,928
Null	69299776	27.895	31328	0,045
Book-chapter	19996152	8.049	1420249	7,103
Proceedings-article	9131851	3.676	1565147	17,139
Dissertation	6473616	2.606	363198	5,610

Continued on next page

Document Type	Total Publication	Percentage of Total	Oplenly Accessible Publications	Percentage Openly Accessible
Posted-content	4784211	1.926	1667595	34,856
Book	4720118	1.899	281905	5,972
Dataset	2881947	1.160	314971	10,929
Reference-entry	1084756	0.437	169559	15,631
Journal-issue	1011348	0.407	273998	27,092
Other	871640	0.351	108462	12,443
Report	789516	0.318	531069	67,265
Monograph	583477	0.235	219476	37,615
Standard	363972	0.147	302	0,083
Peer-review	331312	0.133	142318	42,956

Continued on next page

Document Type	Total Publication	Percentage of Total	Oplenly Accessible Publications	Percentage Openly Accessible
Reference-book	222764	0.089	2208	0,991
Component	201963	0.081	53497	26,489
proceedings	159664	0.064	97284	60,930
Grant	65217	0.026	4272	6,550
Journal	55964	0.023	16694	29,830
Report-series	18492	0.007	13559	73,324
Book-part	16500	0.006	521	3,158
Journal-volume	8251	0.003	2290	27,754
Book-series	3158	0.001	258	8,170
Proceeedings-series	1728	6.96E-4	453	26,215

Continued on next page

Document Type	Total Publication	Percentage of Total	Oplenly Accessible Publications	Percentage Openly Accessible
Book-set	451	1.82E-4	59	13,082

Table 5.3: Distribution of types of publications and distribution of open accessibility.

Discussion:

From the result set in Table 5.3, it can be clearly argued that Journal-article is the most dominant publication type, constituting more than 50% of the total publication, whereas Book-set is the category with the lowest number of publications. It is also interesting to note that almost 28% of the publications are assigned to the Null type. Upon closer examination, it was found that certain publications that are categorized as Null belong to other categories. This observation indicates that OpenAlex does suffer from incomplete information.

Although Journal-article stands out as the most prolific publication type in terms of the total number of publications, the total proportion of publications in Journal-article that are openly accessible accounts for only about 28%.In contrast, the publication type with the highest number of openly accessible publications is Report-series, with about 74% open accessibility among its publications. Another intriguing observation is that, of almost 28% of the publications that are assigned to Null type, less than one percent, 0.045% to be precise, are openly accessible. The overall data suggests that while open accessibility provides easy access to publications, the dominance of subscription-based access remains prevalent in the publication realm.

5.4.1.2 Distribution of retracted publications:

OpenAlex indexes about 11631 publications that had been retracted. The proportion of these retracted works that belonged to different types of publications is displayed in Table 5.5.

Implementation:

The total number of publications that were retracted was calculated utilizing the data frame that includes publications, their respective type, and their retracted status. The schema of the data frame is shown in Table 5.4.

Initially, the publications were grouped according to their type, and the boolean column *is_retracted* was exploited to ascertain whether the publications within each individual publication type had been retracted. In particular, *is_retracted = True* signified that the publication has been retracted, while *is_retracted =False* signified otherwise. The snippet of code is shown in Listing 5.5.

Column Name	Data Type	Description
id	String	The OpenAlex ID for publication
type	String	types of publication
is_retracted	Booelan	Boolean value for retraction

Table 5.4: Data frame with publication ID, publication type, and retracted status.

```
#reading parquet file
read_works = spark.read.parquet("D:\
   open_alex_parquet\works.parquet")
#dropping duplicates if any
filter_works = read_works.dropDuplicates(["id"])
#selecting only the required columns
works = filter_works.select("id","type","
   is_retracted")
#displaying the distribution of retracted
   publications within publication types
```

```
8 works.select("type").filter(lower(col("
    is_retracted"))=="true").groupBy("type").count
    ().orderBy(col("count").desc()).show(truncate=
    False)
```

Listing 5.5: Source code for distribution of retracted publications

Document Type	Total Publication	Total Retracted	Percentage of Retracted
Journal-article	125354155	11441	9.13E-03
Book-Chapter	69299776	146	2.10E-04
Other	19996152	27	1.35E-04
Proceedings-article	9131851	6	6.57E-05
Posted-content	6473616	4	6.17E-05
Null	4784211	3	6.27E-05
Component	4720118	3	6.35E-05
book	2881947	1	3.46E-05

Table 5.5: Distribution of retracted publications.

Discussion:

Judging by the results presented in Table 5.5, it is apparent that out of 26 different publication types, the publications had only been retracted from 8 publication types. Notably, Journal-article, which boasts the highest number of publications, also records the highest number of retractions, i.e., 11441. Considering the vast majority of total publications among different publication types, it is evident that only a small fraction has been retracted so far.

5.4.1.3 Comparative analysis of the distribution of publications:

A comparative analysis of the distribution of publications was conducted, spanning two distinct periods, from 1981 to 2001 and from 2002 to 2022. These time frames would cover both the beginning of digital publishing, I.e., around the 1980s and the evolution through advancements, I.e., early 2000s and beyond. The goal here was to examine the patterns and trends in publications among the various types of publications. By carefully exploring the data, valuable insight was gained into how publishing has evolved over time.

Implementation:

A data frame was used that contained all the publications, their respective type, and their year of publication. The schema of the data frame is shown in Table 5.6. Initially, the publications were filtered according to their year of publication, first within the years ranging from 1981 to 2001 and next by applying the years between 2002 to 2022. The resulting publications were then grouped together according to their type using the "groupBY" function, and then the "count" function was used to count the total number of publications. This resulted in the total number of publications for different publication types for the mentioned years. Additional calculation was conducted to determine the extent to which the publication count had increased within each of the publication types. The result of the analysis is shown in Table 5.7. The snippet of the source code is shown in Listing 5.6.

Column Name	Data Type	Description
id	String	The OpenAlex ID for publication
type	String	Types of publication
publication_year	Integer	The year of publication

Table 5.6: Data frame with publication ID, publication type, and year of publication.

```
#reading parquet file
read_works = spark.read.parquet("D:\
    open_alex_parquet\works.parquet")
#dropping duplicates if any
filter_works = read_works.dropDuplicates(["id"])
#selecting only the required columns
works = filter_works.select("id","type","
    publication_year")
#Filtering first by publication years between 1980
     to 2001
works_40_yrs = works.select("type").filter(col("
    publication_year").between(1981,2001)).groupBy(
    "type").count().withColumnRenamed("count","
    Total_count(1981-2001)" ).orderBy(col("
    Total_count(1981-2001)").asc())
works_40_yrs.show(truncate=False)
#filtering by publication years between 2002 to
    2022
works_20_yrs = works.select("type").filter(col("
    publication_year").between(2002,2022)).groupBy(
    "type").count().withColumnRenamed("count","
```

```
    Total_count(2002-2022)" ).orderBy(col("
    Total_count(2002-2022)").asc())
works_20_yrs.show(truncate=False)
```

Listing 5.6: Source code for comparative analysis of publications for the years (1981 to 2001) and (2002 to 2022)

Document Type	Years(1981 - 2001)	Years(2002 - 2022	Increment by factor
Journal-article	26526703	78056646	2.943
Null	13068130	50691724	3.879
Book-chapter	2952543	15265340	5.170
Proceedings-article	1278468	7699773	6.023
Dissertation	717703	5554663	7.740
Posted-content	466952	4192377	8.978
Book	1288758	2288857	1.776
Dataset	201153	2626792	13.059

Continued on next page

Document Type	Years(1981 - 2001)	Years(2002 - 2022	Increment by factor
Reference-entry	22695	1060372	46.723
Journal-issue	138553	735865	5.311
Other	105540	721616	6.837
Report	245301	349188	1.424
Monograph	68190	369918	5.425
Standard	330	361450	1095.303
Peer-review	0	325244	-
Reference-book	40704	181655	4.463
Component	11000	190844	17.349
proceedings	32763	118092	3.604

Continued on next page

Document Type	Years(1981 - 2001)	Years(2002 - 2022	Increment by factor
Grant	3555	60634	17.056
Journal	552	54810	99.293
Report-series	2481	15493	6.245
Book-part	144	16156	112.194
Journal-volume	1	8014	8014
Book-series	8	3141	392.625
Proceeedings-series	125	1600	12.800
Book-set	0	441	-

Table 5.7: Comparative analysis of publications for the years (1981 to 2001) and (2002 to 2022).

Discussion:

From the results displayed in Table 5.7, it can be clearly stated that the publication was on increasing trend for all the publication types. Two new publication types, Peer-review and Book-set were introduced between the years 2002 and 2022, which were non-existent during the years 1981 to 2001. Journal-articles

held the record for the highest number of publications for both time spans, however, with an increment factor of only about 2.943. The publication type Standard gained more popularity over the years, from just 330 publications between the years 1981 and 2001 to 361450 publications between 2002 and 2022.

Several key factors could elucidate the disparities observed between these two time frames. First, the digital revolution in the early 2000s has significantly enhanced the accessibility of publications. The ease of online publishing and open-access initiatives has facilitated the rapid dissemination of publications. Secondly, the shift in authorship dynamics with an increase in international collaborations has fostered the growth of publication(Kwiek [31]). Furthermore, there is rapid growth in the use of new channels for publications, such as open archives, and personal websites(Larsen and von Ins [32]). Lastly, the decrease in gender disparities with the increase in women's participation has largely changed the publication trend (West et al. [33]). Together, these factors represent the evolving landscape of scholarly publication.

5.4.2 Impact of accessibility and authorship on citation and publication patterns:

To gain a comprehensive understanding of the publication pattern, it is beneficial to analyze and understand the key metrics such as publication over time, authorship over time, and citation over time. These metrics provide valuable insight into the dynamics of publication. The following subsections describe each of the metrics in detail, thereafter delving deeper into the relationship that exists between them.

An important point to emphasize is that the majority of the forthcoming time analysis spans the time between 1992 and 2022. The 30-year window reveals meaningful patterns and changes. Another main reason for choosing the time frame is for readability and ease of presentation.

5.4.2.1 Publication per year:

Publication per year is a metric used to determine the volume and pace of scholarly publication. Tracking publications per year is crucial for understanding the growth, evolution, and distribution of publications over time.

Implementation:

In order to calculate the total works published per year, a data frame was created that contained all the publications and the year they were published. The schema of the data frame is shown in Table 5.8.

The *published_year* column was filtered to include only the years between 1992 and 2022. The total publications for each individual year were then calculated by grouping the publications together according to their respective published year. The visual presentation of the findings is shown in Figure 5.4. The snippet of the source code is shown in Listing 5.7.

Column Name	Data Type	Description
id	String	The OpenAlex ID for publication
publication_year	Integer	The year of publication

Table 5.8: Data frame with publication ID and year of publication.

```
#reading parquet file
read_works = spark.read.parquet("D:\
    open_alex_parquet\works.parquet")
#dropping duplicates if any
filter_works = read_works.dropDuplicates(["id"])
#selecting only the required columns
works = filter_works.select("id","publication_year
    ")
#filtering by publication years between 1992 to
    2022 and displaying the results
yearly_publication = works.select("
    publication_year").filter(col("publication_year
    ").between(1992,2022)).groupBy("
```

```
    publication_year").count().orderBy(col("
    publication_year").asc())
yearly_publication.show(truncate=False)
```

Listing 5.7: Source code for publications per year

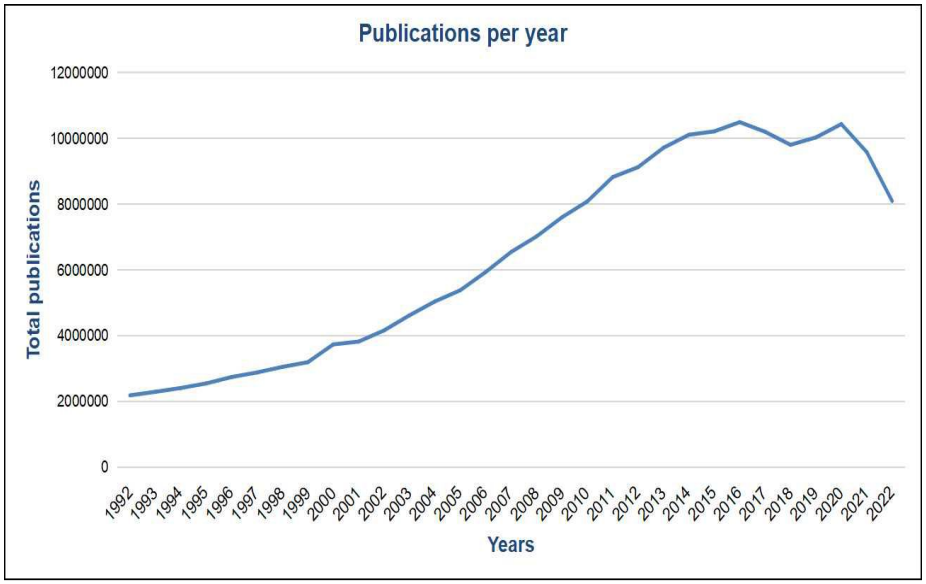


Figure 5.4: Publications per year

Discussion:

Figure 5.4 indicates a consistent upward trend in the annual total number of publications. Many factors can contribute to the increment of publication, such as the growth of research topics, advancement of technology, international collaboration among authors, and increase in research funds. While these factors generally support the increment of publications, several other reasons, including the article's quality, the publisher's publication procedures, and the editors' workload, can add to the decrease in the rate of publication(Taşkın et al. [34]). One such gradual decrement can be witnessed in the figure during the period 2016 to 2018, with total publications declining from more than 10 million in 2016 to a little less than 10 million in 2018. Right after that, a progressive inclination in publications can be seen during the period 2019 to 2020, with the total publications crossing 10 million again in 2020, which,

according to (Rosenfeld and Aviv-Reuven [35]), might be due to the COVID-19 pandemic. Further, a steep decline can be seen in the publication in most recent years, from 2021 to 2022. A reason for this could be that not all the papers published during this time frame have been completely updated in the OpenAlex repository.

5.4.2.2 Authors per year:

Analyzing authorship per year provides knowledge about the evolving landscape of authorships. Following the authorship trend is important as it helps institutions and organizations make decisions regarding resource allocation. Over the course of time, such analysis could unveil changes in author demographics, gender, for instance, and overall literacy rates as the increasing number of authors indicate greater access to publications and larger contributions to scholarly outputs.

Implementation:

In order to determine the total number of authors over the years, two data frames were made. The first data frame consisted of publications and their authors, which is identical to the data frame shown in Table 5.8. The second data frame consisted of publications and the year in which they were published. The schema of the second data frame is shown in Table 5.9.

Initially, the second data frame was filtered to contain only the years from 1992 to 2022. Subsequently, the data frames were joined together on columns *id* from the first data frame and *work_id* from the second data frame. The total number of authors was then calculated by counting the number of *unique* authors that published the papers during each particular year. The resulting graph is represented by Figure 5.5. The snippet of the source code is shown in Listing 5.8.

Column Name	Data Type	Description
work_id	String	The OpenAlex ID for publication
author_id	String	The OpenAlex ID for author

Table 5.9: Data frame with publication ID and author ID.

```
#reading first parquet file
read_authorship = spark.read.parquet("D:\
    open_alex_parquet\works_authorships.parquet")
#reading second parquet file
read_works = spark.read.parquet("D:\
    open_alex_parquet\works.parquet")
#dropping duplicates if any
filter_works = read_works.dropDuplicates(["id"])
filter_authorship = read_authorship.dropDuplicates
    (["work_id","author_id"])
#selecting only the required columns
works = filter_works.select("id","publication_year
    ")
authors = filter_authorship.select("work_id","
    author_id")
#filtering by publication years between 1992 to
    2022 and displaying the results
select_years = works.select("id","publication_year
    ").filter(col("publication_year").between
    (1992,2022))
#joining the dataframes
combined = authors.join(select_years,authors.
    work_id == select_years.id ,"inner")
#displaying the total number of unique authors per
     year
```

```
16 combined.select("*").groupBy("publication_year").
     agg(countDistinct("author_id")).orderBy(col("
     publication_year").asc()).show(truncate=False)
```

Listing 5.8: Source code for unique authors per year

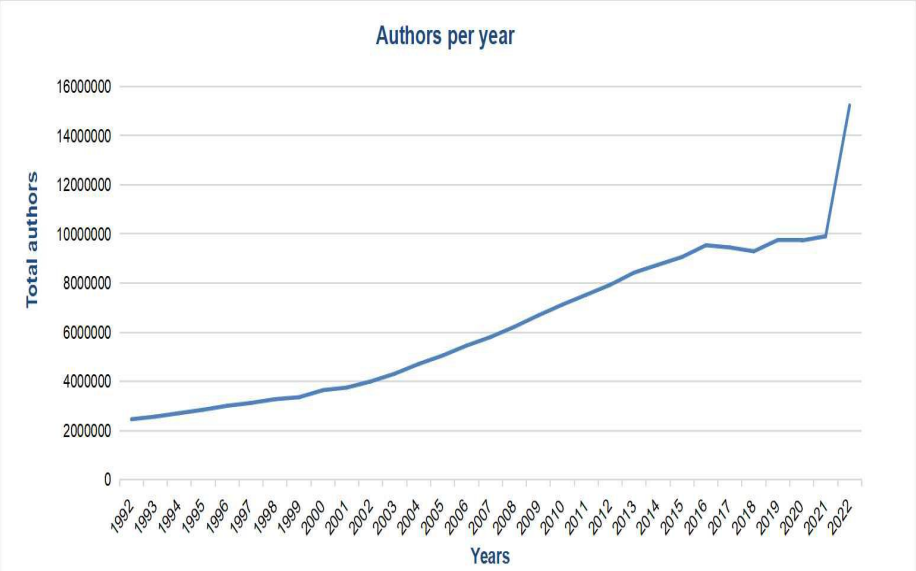


Figure 5.5: Authors per year

Discussion:

A steady upward trajectory can be noticed in the number of authors each year, meaning that every year, more authors were contributing to research publications. This increasing trend in authorship can be attributed to several factors, such as advancement in technologies, increasing national and international collaboration, and diversified areas of research.

In a more detailed comparative analysis between the total number of authors per year and the total number of publications per year, as displayed in Figure 5.6, It was clearly visible that until the year 1999, the total number of authors exceeded the total number of publications. This indicates that the publications until then were a result of the collaborative effort of multiple authors. On the other hand, after the year 2000, the total number of publications surpassed

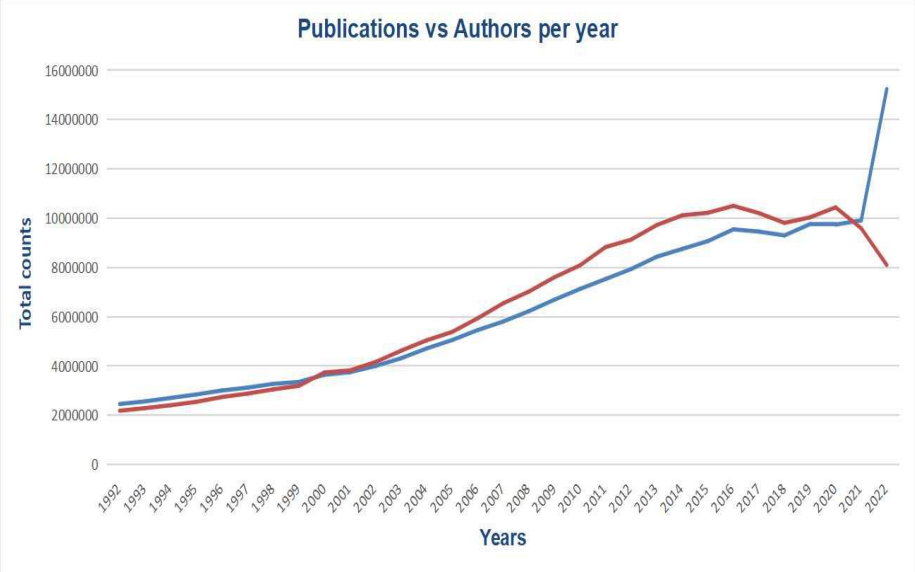


Figure 5.6: Authors vs Publications per year

the total number of authors. The change can be ascribed to influences such as digital publications and open-access journals. However, the graph shows a drastic increase in the number of authors in recent years, i.e., 2021 to 2022, while the total number of publications has significantly declined. This could be as a result of an inadequate update of publications in the OpenAlex.

5.4.2.3 Citations per year:

Citations per year is a metric used to measure the impact of scholarly articles. This metric is very crucial as it helps to gauge the influence and significance of publications. A higher number of citations indicates a significant contribution by the publication. Governments and institutions often rely on this metric to make decisions regarding research funding.

Implementation:

A data frame was created that contained all the publications and the publications which were cited by them. Table 5.10 displays the schema of this data frame.

The data frame was first filtered only to include the years from 1992 to 2022. The publications were categorized based on these years. All the cited/referenced publications were counted for each particular year to determine the total citations that were made during that year. The final result is displayed in Figure 5.7. The snippet of the source code is shown in Listing 5.9.

Column Name	Data Type	Description
work_id	String	The OpenAlex ID for publication
publication_year	Integer	The year of publication
referenced_work_id	String	The OpenAlex ID for referenced publication

Table 5.10: Data frame with publication ID, year of publication, and referenced publication ID.

```
#reading parquet file
read_citation = spark.read.parquet("D:\
    open_alex_parquet\citation.parquet")
#dropping duplicates if any
filter_citations = read_citation.dropDuplicates(["
    work_id","referenced_work_id"])
#selecting only the required columns
citation = filter_citations.select("work_id","
    publication_year","referenced_work_id")
#filtering by publication years between 1992 to
    2022 and displaying the results
citation.filter((col("publication_year").between
    (1992,2022))).groupby("publication_year").count
```

```
    ().orderBy(col("publication_year").asc()).show(
    truncate=False)
```

Listing 5.9: Source code for citations per year

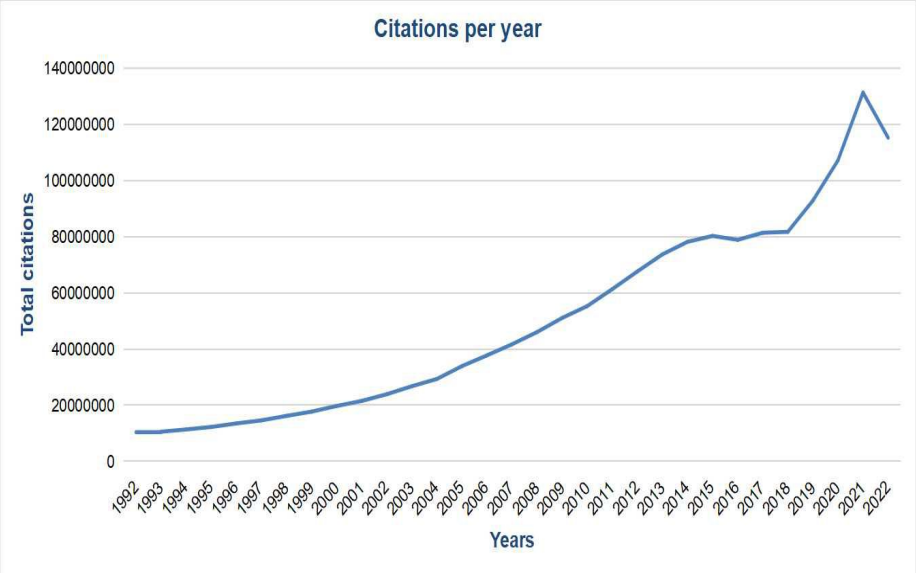


Figure 5.7: Citations per year

Discussion:

A progressively increasing trend in total citation count can be observed in Figure 5.7. A meaning that can be extracted from this is that every year, publications had a substantial impact. A slight decline in citations can be seen between the years 2016 to 2018. A reason for this is due to the lower number of publications during these years, as seen in Figure 5.4. A substantial rise in the total number of citations can be seen in the subsequent years up until the recent years (2021-2022). It could be inferred that the lack of updates in OpenAlex could be the reason behind this downfall of citations in those years.

5.4.2.4 Average publications per authors per year:

The average publications per authors per year is used to gauge the authors' productivity. It expresses the frequency of publications by the authors. A higher average value indicates that the authors publish works consistently. However, the higher value does not always indicate high-quality output. Mathematically, it is calculated as:

$$\text{Average publications per authors per year} = \frac{\text{Total publications in a year}}{\text{Total unique authors in a year}} \tag{5.1}$$

Implementation:

In order to calculate the average publications per authors per year, the result of publications per year and authors per year from the sections 5.4.2.1 and 5.4.2.2, respectively, were used. For each year, the total number of publications was divided by the total number of authors to get the average publications per author per year. The result is displayed in Figure 5.8. The snippet of the source code is displayed in Listings 5.8 for the total number of authors and 5.7 for the total number of publications.

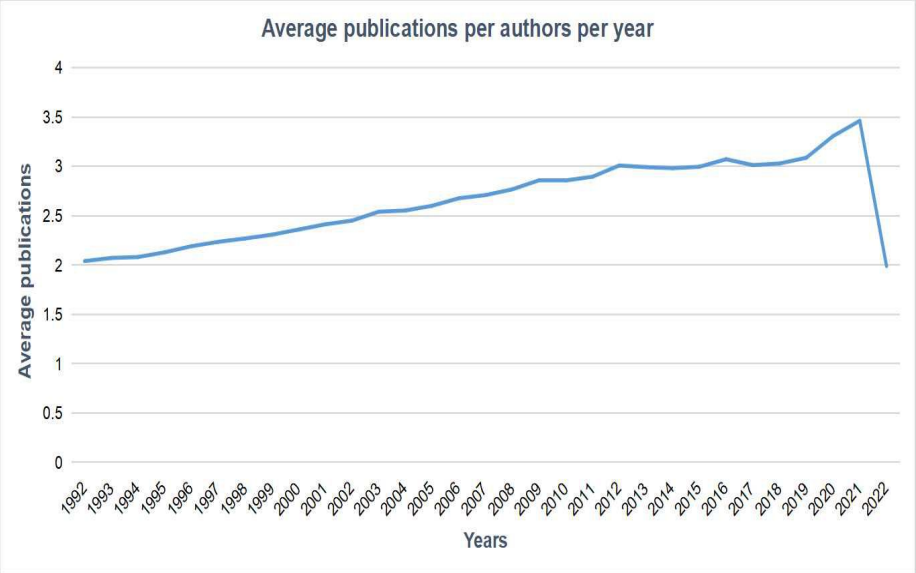


Figure 5.8: Average publications per authors per year

Discussion:

Looking at the graph in Figure 5.8, it can be clearly stated that the average number of publications per authors has been steadily rising up until the year 2019, after which there was a rapid surge until the year 2021. The reason for this growth can be attributed to the emergence of digital publication, the globalization of research, and the increase in the field of study. The downfall during recent years could have resulted from a lack of up-to-date data in the OpenAlex. Other reasons for the downfall could be lack of funding, delay in publications, and prioritization of quality works over quantity. Further, another possible explanation for the decline is that, in certain disciplines, an increasing amount of work is demanded by publications. (Larsen and von Ins[32]).

5.4.2.5 Average authors per publications per year:

The average authors per publications per year is a metric that represents the collaborative nature of research work. It varies based on disciplines and the nature of research. A higher number of average values signifies greater collaboration among researchers, while a lower value signals that the research works have fewer authors. Mathematically, it is calculated as,

$$\text{Average authors per publications per year} = \frac{\text{Total authors in a year}}{\text{Total publications in a year}} \tag{5.2}$$

Implementation:

To derive the average authors per publication per year, the result of publications per year and authors per year from the sections 5.4.2.1 was used. However, for this calculation, instead of the unique number of authors, as derived in 5.4.2.2, the total number of authors responsible for each publication was counted. For each year, the total number of authors was divided by the total number of publications to get the average authors per publications per year. The result is displayed in Figure 5.9. The snippet of the source code is displayed in Listings 5.10, for the total number of authors and 5.7, for the total number of publications.

```
 #reading first parquet file
read_authorship = spark.read.parquet("D:\
   open_alex_parquet\works_authorships.parquet")
#reading second parquet file
read_works = spark.read.parquet("D:\
   open_alex_parquet\works.parquet")
#dropping duplicates if any
filter_works = read_works.dropDuplicates(["id"])
filter_authorship = read_authorship.dropDuplicates
   (["work_id","author_id"])
#selecting only the required columns
authors = filter_authorship.select("work_id","
   author_id")
works = filter_works.select("id","publication_year
   ")
#filtering by publication years between 1992 to
   2022 and displaying the results
select_years = works.select("id","publication_year
   ").filter(col("publication_year").between
   (1992,2022))
#joining the dataframes
combined = authors.join(select_years,authors.
   work_id == select_years.id ,"inner")
#displaying the total number of authors per year
combined.select("*").groupBy("publication_year").
   count("author_id").orderBy(col("
   publication_year").asc())
```

Listing 5.10: Source code for authors per year

Discussion:

The number of authors per publication is on the rise. It is primarily influenced by collaborative research efforts and serves as an effective means for each author to enhance their productivity (Plume and van Weijen [36]). The results displayed in Figure 5.9 also seem to be in support of this claim. The average authors per publications has risen from less than 2.5 to more than 3.5 in the span of 30 years. While technological advancement and ease of global commu-

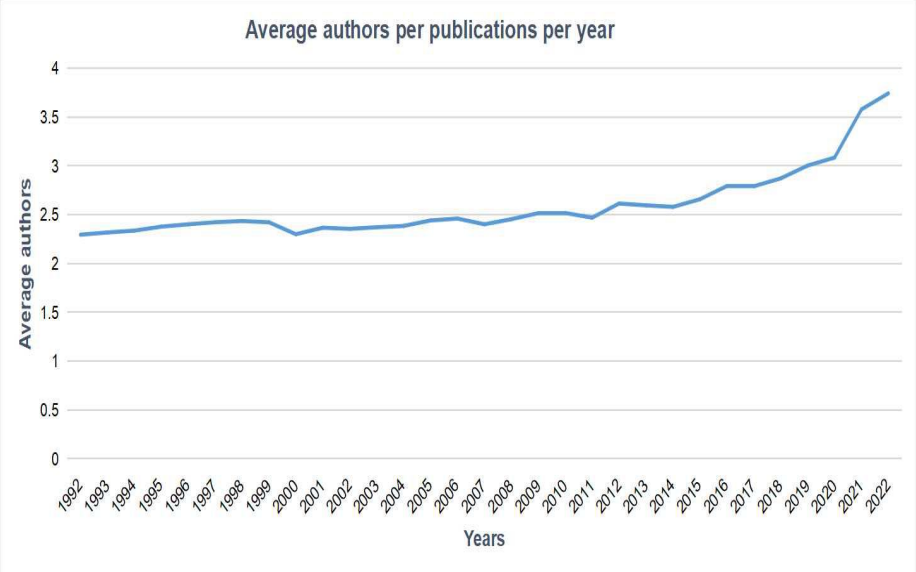


Figure 5.9: Average authors per publications per year

nication are some of the positive factors contributing to the growth of authors every year, the fear of perishing has also led authors to co-author more papers. An individual author can either publish a single-authored article once every two years or co-author an article with one other collaborator annually (Plume and van Weijen [36]).

5.4.2.6 Average citations per publications per year:

The average citations per publications per year is a metric used in assessing the impact of the publications. The publications that are highly influential tend to have higher citation rates. This metric reflects the quality of the paper and its contributions towards its respective discipline. Mathematically, it is calculated as:

$$\text{Average citations per publications per year} = \frac{\text{Total citations in a year}}{\text{Total publications in a year}} \tag{5.3}$$

Implementation:

To determine the average number of citations per publications per year, the data obtained from sections 5.4.2.1 and 5.4.2.3 was employed. For each year, the total number of citations was divided by the total number of publications to compute the average number of citations per publications. The result of the calculation is displayed in Figure 5.10. The snippets of the source code are displayed in Listings 5.9, for the total number of citations and 5.7, for the total number of publications.

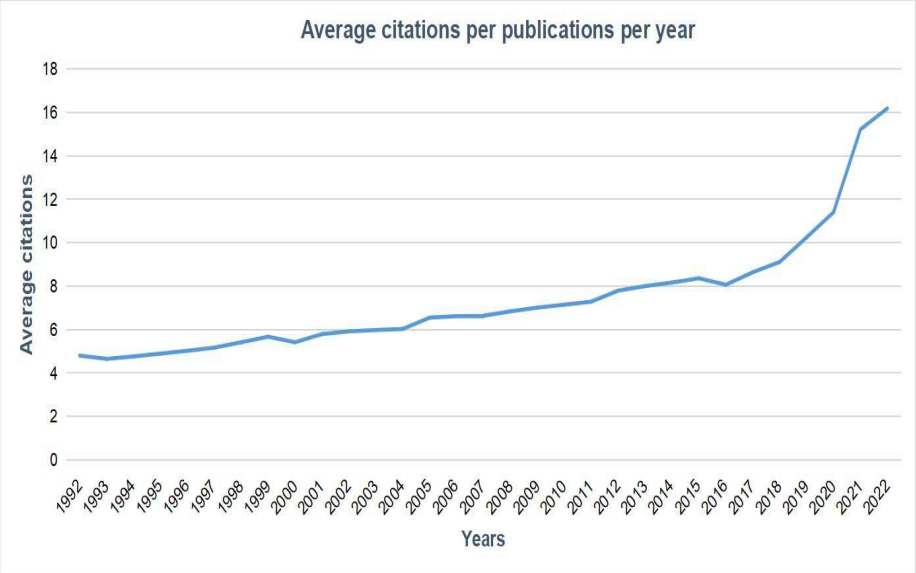


Figure 5.10: Average citations per publications per year

Discussion:

Figure 5.10 illustrates that the average citations per publication was consistently soaring every year. Many aspects could influence this upward trend, including high-quality research works, Open accessibility of publications, or increasing collaboration among researchers. The average number of citations had increased by almost 4 times within the last 30 years, from an average of 4.80 in 1992 to 16.18 in 2022. The high surge in citations during recent years could, however, potentially be caused by insufficient updates in the OpenAlex.

5.4.2.7 Average citations per authors per year:

In this section, the research impact of the authors was measured using the metric average citations per authors per year. It calculates the average number of times the publications of authors were cited by other authors annually. Mathematically, it is calculated as:

$$\text{Average citations per authors per year} = \frac{\text{Total citations in a year}}{\text{Total unique authors in a year}} \tag{5.4}$$

Implementation:

The calculation of average citations per authors per year required two things: the total number of citations for the year and the total number of authors who were responsible for publishing at least one publication during that particular year. For calculating the total number of citations, the results obtained in Section 5.4.2.3 were utilized. Similarly, for calculating the total number of authors, the results obtained in Section 5.4.2.2 were applied. Finally, for each year, the total no. of citations was divided by the total number of authors in order to get the average number of citations per author per year. The result of the calculation is shown in Figure 5.11. The snippets of the source code are displayed in Listings 5.9, for the total number of citations and 5.8, for the total number of authors.

Discussion:

As displayed in Figure 5.11, the average citations per authors per year had an upward momentum. This can infer that every year, more and more quality works were being published by authors that their peers appreciated. On the other hand, increasing the number of citations could also result from self-citation to manipulate the citation-based metrics. A slight decline in average citations can be seen in the year 2016; the reason for this is a small decrease in the total number of citations during the year, while the total number of authors was still on the rise. A steep upsurge can be seen in the average citation between the years 2018 and 2021. This could result from increasing publication during this period as described in Section 5.4.2.1. The downfall right after can be due to the lack of updates in the OpenAlex.

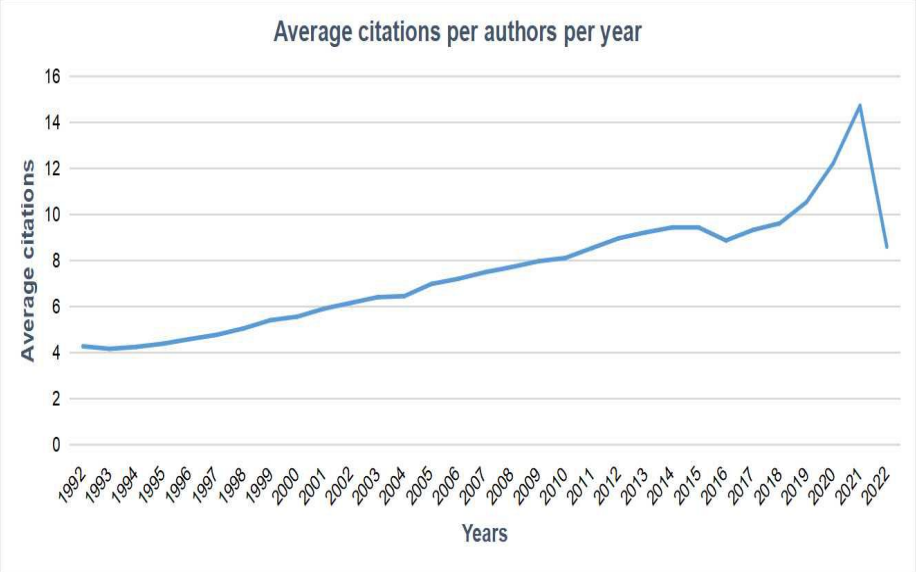


Figure 5.11: Average citations per authors per year

5.4.2.8 Analysis of openly accessible publications vs other publications:

The analysis of openly accessible publications versus other publications sheds light on the evolving dynamics of publication. Openly accessible publications, which are also referred to as open-access publications, are readily accessible to readers without any subscription fees. The term "other publications" in this context defines the publications that are subscription-based, have limited access, or have any other forms of restrictions. OpenAlex has divided the openly accessible publications into four categories: Gold, Green, Hybrid, and Bronze. For all the publications belonging to these four categories, their open access status is set to true, while for all the other publications, the status is set to false.

A comparative evaluation was conducted to determine the proportion of publications every year that are openly accessible to that of other publications. The main aim of this analysis was to evaluate the effectiveness of open-access publications in distributing research works.

Implementation:

For performing this analysis, two data frames were constructed. The first data frame contained all the publications and their publication year, and the second data frame consisted of the publications and their accessibility status. The schemas of the data frames are identical to the data frames depicted in Tables 5.8 and 5.2, respectively.

Utilizing the first data frame, the *publication_year* column was first filtered to include only the years between 1992 and 2022. The data frames were joined together on columns *id* from the first data frame and *work_id* from the second data frame. Thereafter, operating the combined data frame, the *is_oa* column was filtered to identify all the publications marked as "true," denoting that the publications are openly accessible. The resulting output was grouped according to their publication years to calculate the total number of openly accessible publications for each year.

A similar calculation was performed to calculate the total works for works that are not openly accessible by filtering the *is_oa* column with the condition *is_oa = false*. The outcome of the calculations is represented in Figure 5.12. The snippet of the source code is displayed in Listing 5.11.

```
#reading first parquet file
read_works = spark.read.parquet("D:\
    open_alex_parquet\works.parquet")
#reading second parquet file
read_works_oa = spark.read.parquet("D:\
    open_alex_parquet\works_open_access.parquet")
#dropping duplicates if any
filter_works = read_works.dropDuplicates(["id"])
filter_works_oa = read_works_oa.dropDuplicates(["
    work_id"])
#selecting only the required columns
works_oa = filter_works_oa.select("work_id","is_oa
    ")
works = filter_works.select("id","publication_year
    ")
#filtering by publication years between 1992 to
    2022
```

```
select_years = works.select("id","publication_year
    ").filter(col("publication_year").between
    (1992,2022))
#joining the data frames
combined = select_years.join(works_oa,select_years
    .id ==  works_oa.work_id,"inner")
#for openly accessible works
total_works_oa = combined.select("publication_year
    ").filter(lower(col("is_oa"))=="true").groupBy(
    "publication_year").count().orderBy(col("
    publication_year").asc())
total_works_oa.show(truncate=False)
#for works not openly accessible
total_works_non_oa = combined.select("
    publication_year").filter(lower(col("is_oa"))==
    "false").groupBy("publication_year").count().
    orderBy(col("publication_year").asc())
total_works_non_oa.show(truncate=False)
```

Listing 5.11: Source code for openly accessible publications vs other publications

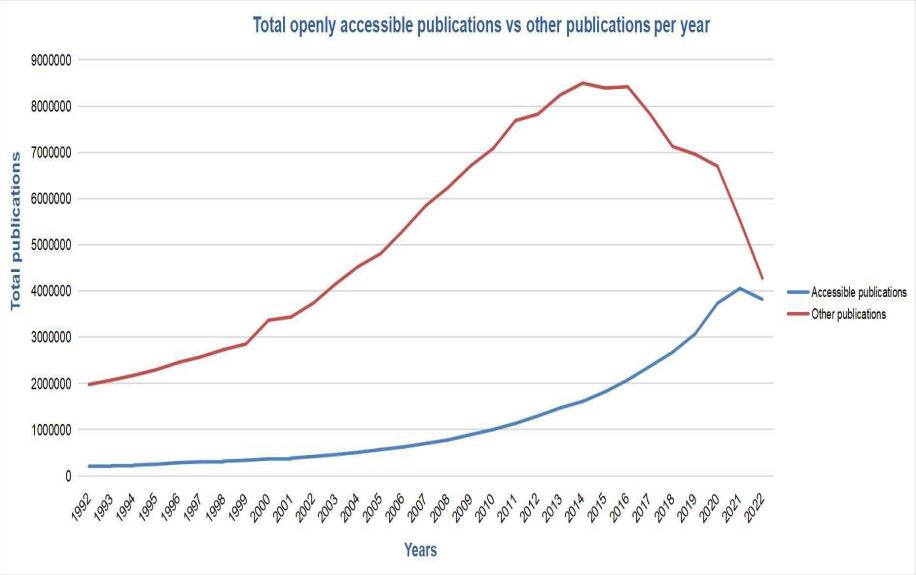


Figure 5.12: Openly accessible publications vs other publications per year

Discussion:

Judging by the result as displayed in Figure 5.12, it can be clearly stated that even though the publication dynamics have changed due to the introduction of the open-access publication model, the total number of publications in other models of publications still remain dominant. One major reason for this dominance could be because of Article Processing Charges(APC) that the authors have to pay in order to publish their articles in open-access journals. Not every author can afford to pay the charge, which leads to a smaller number of publications in such journals (Momeni et al.[37]).

A rise in the total number of publications every year for both of the publication models can be seen until the year 2014. Thereafter, the total publications in other publication models show a steep decline, while in the open-access publication model, the publications are still on the rise. The reason for this phenomenon could be that many closed-access journals are flipping to open-access journals, and reportedly, the total number of publications after the flipping has increased substantially (Momeni et al.[37]).

5.4.2.9 Proportion of citations on openly accessible publications vs other publications:

Calculating the proportion of citations between openly accessible publications and other publications provides insight into the impact and influence of openly accessible publications as compared to the impact of other publications. It reflects the extent to which fellow authors and audiences prefer these publication models. A higher proportion of citations for openly accessible publications would indicate their reach and impact on reshaping the traditional publication model.

Implementation:

To accomplish this objective, two data frames were created. The first data frame consisted of information, including the publications, the year of their publication, and references they made to other publications. The second data frame comprised publications along with their accessibility status. The schemas of the data frames are corresponding to data frames shown in Tables 5.10 and 5.2.

Using the first data frame, the publications were filtered to exclusively include the publications published between the years 1992 and 2022. Subsequently, two data frames were merged together on columns *referenced_work_id* from the first data frame and *work_id* from the second data frame in order to determine the accessibility status of all the cited publications. The resulting dataset was further refined to include only the publications that were openly accessible, followed by grouping the publications according to the year of publication. The total count of cited publications for each year was computed.

The total number of citations, as calculated in Section 5.9, was utilized to calculate the proportion of citations received by openly accessible publications. The computation involved dividing the total number of citations received by openly accessible publications by the total number of citations for each year.

A similar calculation was performed to derive the proportion of citations received by other publications that were not openly accessible. The findings obtained through these calculations are displayed in Figure 5.13. The snippet of the source code is displayed in Listing 5.12

```
#reading first parquet file
read_citation = spark.read.parquet("D:\
    open_alex_parquet\citation.parquet")
#reading second parquet file
read_works_oa = spark.read.parquet("D:\
    open_alex_parquet\works_open_access.parquet")
#dropping duplicates if any
filter_citations = read_citation.dropDuplicates(["
    work_id","referenced_work_id"])
filter_works_oa = read_works_oa.dropDuplicates(["
    work_id"])
#selecting only the required columns
citation = filter_citations.select("work_id","
    publication_year","referenced_work_id")
works_oa = filter_works_oa.select("work_id","is_oa
    ")
#filtering by publication years between 1992 to
    2022
citation_years = citation.filter((col("
    publication_year").between(1992,2022)))
```

```
#joining the data frames
combined = citation_years.join(works_oa,
    citation_years.referenced_work_id ==  works_oa.
    work_id,"inner")
#citations for openly accessible works
citations_works_oa=combined.filter((lower(col("
    is_oa"))=="true")).groupby("publication_year").
    count().orderBy(col("publication_year").asc())
citations_works_oa.show(truncate=False)
#citations for works not openly accessible
citations_works_non_oa=combined.filter((lower(col(
    "is_oa"))=="false")).groupby("publication_year"
    ).count().orderBy(col("publication_year").asc()
    )
citations_works_non_oa.show(truncate=False)
```

Listing 5.12: Source code for citations on openly accessible publications vs other publications

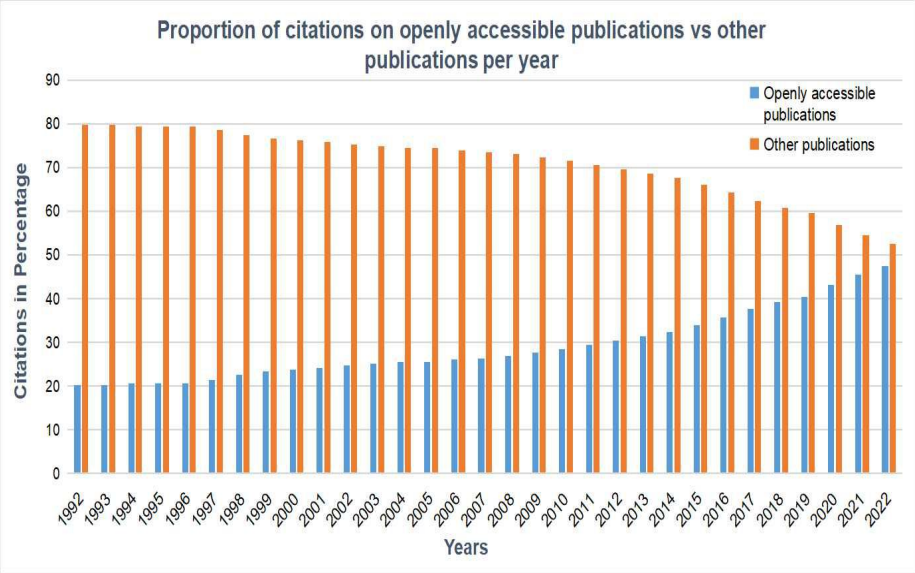


Figure 5.13: Proportion of citations on openly accessible publications vs other publications per year

Discussion:

Figure 5.13 presents a strongly contrasting trend regarding the proportion of citations between openly accessible publications and other publications. While the citations on openly accessible publications are gradually inclining, the exact opposite can be observed for the other publications. The higher citation rates in open-access publications can be credited to the greater accessibility and wider readership of research works. A conclusion that can be derived from this result is that, over the years, openly accessible publications have had a larger impact and are evolving as the preferred model for publication. Several studies support this trend, indicating that open-access publications have received substantially higher citations than closed-access publications (Momeni et al.[37]).

5.4.2.10 Total publications single-authored vs coauthored:

Analysis of total publications, comparing single-authored and coauthored contributions, emphasizes the diversity in authorship for creating and disseminating scholarly works. While single-authored works provide full control and direction over the works to their author, coauthored works are more about collaboration among diverse talents to produce the end result. The decision for single-authored or coauthored publications relies mostly upon the available resources and the nature of the subject. The idea here was to analyze the trend in single-authored versus coauthored publications and how they have evolved over time.

Implementation:

For the analysis, two data frames were formed. The first data frame included the publications and their publication year, and the second data frame consisted of publications and their authors. The schemas of the data frames are the same as the data frames depicted in Tables 5.8 and 5.9.

The first data frame was initially filtered based on the publication years to include only the required years, i.e., 1992 - 2022. Subsequently, using the second data frame, the total number of authors for each publication was counted. The data frames were joined together on columns *id* from the first data frame and *work_id* from the second data frame. Successively, limiting the total author count to 1 on the combined data frame provided only the publications that

were single-authored, while all the other publications with more than one author were categorized as coauthored. The dataset was grouped together on a yearly basis, and the total number of publications was counted for each mode of authorship. The end result of the calculation is displayed in Figure 5.14. The snippet of the source code is displayed in Listing 5.13.

```
#reading first parquet file
read_authorship = spark.read.parquet("D:\
    open_alex_parquet\works_authorships.parquet")
#reading second parquet file
read_works = spark.read.parquet("D:\
    open_alex_parquet\works.parquet")
#dropping duplicates if any
filter_works = read_works.dropDuplicates(["id"])
filter_authorship = read_authorship.dropDuplicates
    (["work_id","author_id"])
#selecting only the required columns
authors = filter_authorship.select("work_id","
    author_id")
works = filter_works.select("id","publication_year
    ")
#filtering by publication years between 1992 to
    2022 and displaying the results
select_years = works.select("id","publication_year
    ").filter(col("publication_year").between
    (1992,2022))
#counting the number of authors for each
    publication
count_authors = authors.groupBy("work_id").count()
#Single-authored publications
#limiting author's count to 1 for single authored
    publications
limited_authors = count_authors.select("*").filter
    (col("count")==1).withColumnRenamed('count',"
    total_authors" )
#joining the dataframes
```

```
19 combined = limited_authors.join(select_years,
       limited_authors.work_id == select_years.id ,"
       inner")
20 #Total single-authored publications
21 works_single_authored=combined.groupby("
       publication_year").count().orderBy(col("
       publication_year").asc())
22 works_single_authored.show(truncate=False)
23 #Coauthored publications
24 #limiting author's count to more than 1 for
       coauthored publications
25 limited_authors = count_authors.select("*").filter
       (col("count")>1).withColumnRenamed("count","
       total␣authors" )
26 #joining the dataframes
27 combined = limited_authors.join(select_years,
       limited_authors.work_id == select_years.id ,"
       inner")
28 #Total coauthored publications
29 works_couthored =combined.groupby("
       publication_year").count().orderBy(col("
       publication_year").asc())
30 works_couthored.show(truncate=False)
```

Listing 5.13: Source code for total publications single-authored vs coauthored

Discussion:

The graph depicted in Figure 5.14 clearly illustrates that the total number of publications for both single and coauthored mode of authorship have been escalating gradually. However, the publications by single authors downsized after the year 2016 while the coauthored publications continued to rise. The complexity of research works, the increasing need for diverse expertise, and the pressure of publishing could be the reason why authors opt to publish collaboratively. Over the years, there has been a notable surge in co-authored works across various fields and disciplines, primarily in subject areas that require quantitative research techniques, conducting experiments, and emphasizing the division of labor (Henriksen [38]). Notably, the upward trajectory in collab-

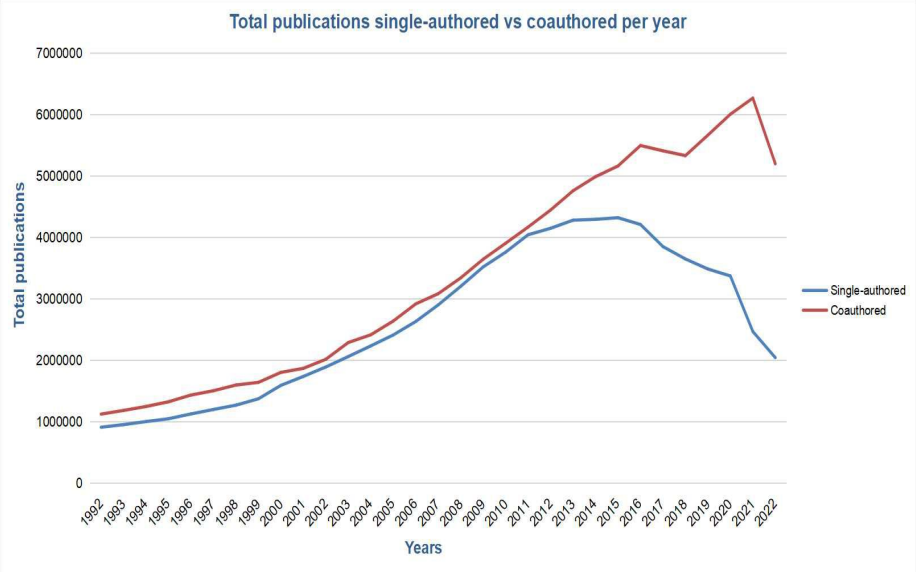


Figure 5.14: Total publications single-authored vs coauthored per year

orative publications has remained consistent in recent years, as indicated by the figure.

5.4.2.11 Proportion of citation on single-authored works vs coauthored publications:

Analysis of citations in single-authored versus coauthored publications offers valuable insight into the contrasting influence and extent of these distinct authorship modes. Single-authored publications constitute the contribution of individual intellect. Coauthored publications, on the other hand, offer collective expertise through collaboration among authors. The goal of this analysis is to reveal the extent of readership and impact that these two modes of authorship generated over time.

Implementation:

To conduct this analysis, two data frames were created. The first data frame consisted of all the publications, including their publication year and all the publications that they cite. The second data frame consisted of all the publications and their respective authors. The schemas of the data frames are identical to the ones displayed in Tables 5.10 and 5.9.

Using the second data frame, the total number of authors for each publication was computed by grouping the individual publications and counting the total number of authors involved. Subsequently, two data frames were merged together on columns *referenced_work_id* from the first data frame and *work_id* from the second data frame in order to determine the total number of authors responsible for the cited publications. The resulting dataset was further refined to include only the publications published by single authors by limiting the total authors count to 1. The publications were then grouped according to the year of publication. The total count of citations for single-authored publications for each year was computed.

The total number of citations, as calculated in Section 5.9, was utilized to calculate the proportion of citations received by single-authored publications. The computation involved dividing the total number of citations received by single-authored publications by the total number of citations for each year.

Similar operations were carried out to calculate the proportion of citations received by coauthored publications by limiting the number of total authors to more than 1. The findings obtained through these calculations are displayed in Figure 5.15. The snippet of the source code is displayed in Listing 5.14.

```
1  #reading first parquet file
2 read_authorship = spark.read.parquet("D:\
     open_alex_parquet\works_authorships.parquet")
3 #reading second parquet file
4 read_citation = spark.read.parquet("D:\
     open_alex_parquet\citation.parquet")
5 #dropping duplicates if any
6 filter_citations = read_citation.dropDuplicates(["
     work_id","referenced_work_id"])
7 filter_authorship = read_authorship.dropDuplicates
     (["work_id","author_id"])
```

```
#selecting only the required columns
authors = filter_authorship.select("work_id","author_id")
citation = filter_citations.select("work_id","publication_year","referenced_work_id")
#filtering by publication years between 1992 to 2022 and displaying the results
citation_years = citation.filter((col("publication_year").between(1992,2022)))
#counting the number of authors for each publication
count_authors = authors.groupBy("work_id").count()
#Citations on single-authored publications
#limiting author's count to 1 for single authored publications
limited_authors = count_authors.select("*").filter(col("count")==1).withColumnRenamed("count","total_authors" )
#joining the dataframes
combined = limited_authors.join(citation_years, limited_authors.work_id == citation_years.referenced_work_id ,"inner")
#Total citations on single-authored publications
citations_single_authored=combined.groupby("publication_year").count().orderBy(col("publication_year").asc())
citations_single_authored.show(truncate=False)
#Citations on coauthored publications
#limiting author's count to more than 1 for coauthored publications
limited_authors = count_authors.select("*").filter(col("count")>1).withColumnRenamed('count','total_authors" )
#joining the dataframes
combined = limited_authors.join(citation_years, limited_authors.work_id == citation_years.referenced_work_id ,"inner")
#Total citations on coauthored publications
```

```
29 citations_couthored =combined.groupby("
       publication_year").count().orderBy(col("
       publication_year").asc())
30 citations_couthored.show(truncate=False)
```

Listing 5.14: Source code for citation on single-authored works vs coauthored publications

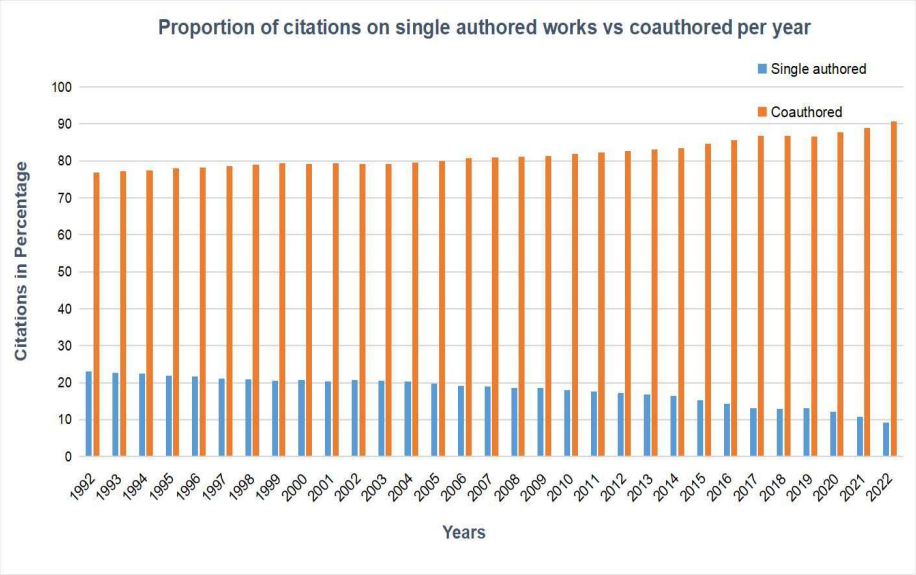


Figure 5.15: Citations single-authored vs coauthored publications per year

Discussion:

According to a paper published in 1984, papers published by multiple authors receive more citations than papers published by single authors (Abt [39]). Remarkably, this trend persists even after nearly four decades, as indicated by the graph displayed in Figure 5.15, which showcases the proportion of the citations received by single-authored and coauthored publications. Notably, the proportion of citations was continuously receding for single-authored publications, while for coauthored publications, the proportion of citations was incrementally improving. The plausible cause for this difference in citations in these different modes of authorships could be because the coauthored publications benefit from the contribution of multiple expertise. Furthermore,

the involvement of multiple authors makes the publication more reliable and trustworthy.

5.4.3 Analysis of self-citations:

Self-citation is a practice wherein an author references their previously published works in their new publications. There are various reasons why an author would self-cite their publications; on the positive side, the self-citation could be carried out to establish a foundation of concepts about research that have been covered in previous publications and is necessary to understand the new research work. Conversely, authors might resort to self-citation to manipulate the citation-based metrics for self-promotion. Self-citations can take many forms, including direct self-citations, where the author cites their own paper; co-author self-citations, in which co-authors of the publication cite it in their subsequent publication; or coercive self-citations, where authors are pressurized to cite a publication (Ioannidis [40]).

Within this thesis work, the proportion of self-citation is analyzed from 1992 to 2022. Furthermore, the focus here is solely on self-citations by the main author and co-author. Consequently, a comparative study is conducted to determine the frequency of self-citations by the main author and the co-authors. Both of the analyses are described in the following subsections.

5.4.3.1 Proportion of self-citations per year:

Analyzing self-citations per year is crucial because it offers perspective on the development of the researcher's work over time. It helps understand how the prior publications of researchers build the foundation for new publications, while, on the other hand, it also provides awareness about the potential self-promotion that the authors might be trying to achieve by manipulating the citation-based metrics. Utilizing this metric, the shifts in self-citation are detected; for instance, the sudden rise in self-citations could indicate the possibility of self-promotion, while the gradual increase could indicate the development of a research program.

Implementation:

The evaluation was performed using two data frames. The first data frame consisted of columns with all the publications, their authors, and the other publications that these publications cited. The second data frame consisted of all publications and their authors. The schemas of these data frames are corresponding to the data frames shown in Tables 5.10 and 5.9.

To begin the analysis, the first data frame was filtered on the *publication_year* column to include only the publications published in the years between 1992 and 2022. Following that, a join was established between these two data frames using the columns *referenced_work_id* from the first data frame and *work_id* from the second data frame as the key columns. For each individual publication in this combined data frame, the author of this publication was compared to the author of the publications that it cites. When a match was identified, it was counted as a self-citation. The total number of self-citations was grouped according to the publication year and then summed in order to find the total self-citations per year.

Finally, to calculate the proportion of self-citations per year, the self-citation for each year was divided by the total citation for the corresponding year, as calculated in Section 5.9. The final result is presented in the Figure 5.16. The snippet of the source code is displayed in Listing 5.15.

```
#reading first parquet file
read_authorship = spark.read.parquet("D:\
    open_alex_parquet\works_authorships.parquet")
#reading second parquet file
read_citation = spark.read.parquet("D:\
    open_alex_parquet\citation.parquet")
#dropping duplicates if any
filter_citations = read_citation.dropDuplicates(["
    work_id","referenced_work_id"])
filter_authorship = read_authorship.dropDuplicates
    (["work_id","author_id"])
#selecting only the required columns
authors = filter_authorship.select("work_id","
    author_id")
```

```
citation = filter_citations.select("work_id","
    publication_year","referenced_work_id")
#filtering by publication years between 1992 to
    2022
citation_years = citation.filter((col("
    publication_year").between(1992,2022)))
#joining the dataframes
combined = citation_years.join(authors,
    citation_years.referenced_works == authors.
    work_id ,"inner")
#Counting self-citations
self_citation_counts=combined.withColumn("
    self_citation_count", when((combined.author_id
    == combined.referenced_author_id), lit(1)).
    otherwise(lit(0)))
#Displaying the results
self_citation_counts.groupBy("publication_year").
    sum("self_citation_count").orderBy(col("
    publication_year").asc()).show()
```

Listing 5.15: Source code for self-citations per year

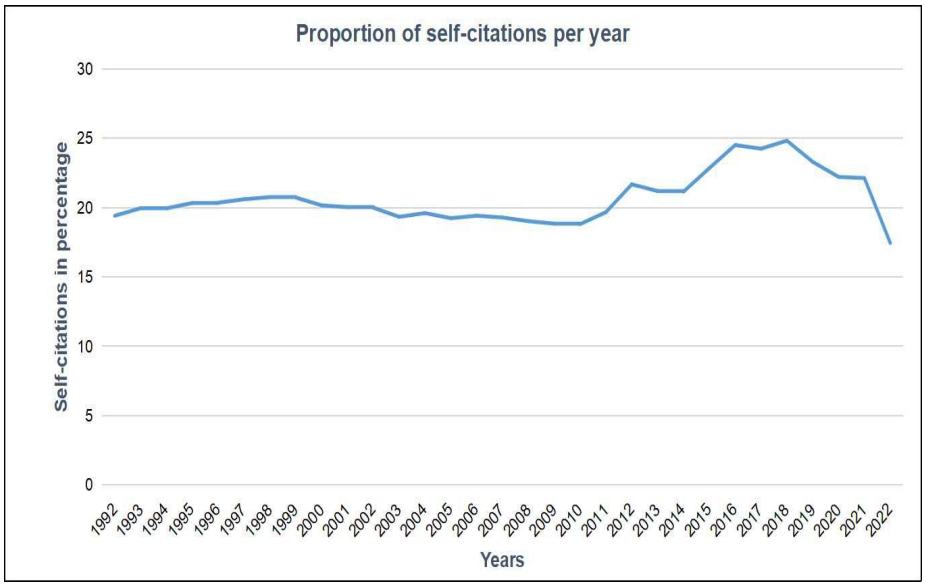


Figure 5.16: Self-citations per year

Discussion:

According to the graph displayed in Figure 5.16, the self-citation remained relatively stable at around 20% per year until 2011, indicating that approximately 20% of total citations are references to the author's own previously published works. After 2011, a sudden increase in self-citations can be seen, reaching up to 25% in the year 2019. The introduction of the h-index in 2005 (Hirsch [11]), which is used as a metric to evaluate the impact and productivity of a researcher's work, could be one of the major reasons for the increase in self-citation as more and more authors would want to self-promote their publication to show the increase in the impact of their publication. The steep decline afterward could indicate the transition of the researcher's focus toward a more diverse field; for instance, during the coronavirus pandemic, numerous papers were published about the virus. However, given that the virus had not been identified before 2019, no previous research works existed that could be self-cited.

5.4.3.2 Proportion of self-citations by main authors vs coauthors:

The proportion of self-citations by main authors compared to coauthors offers a valuable perspective into how the researchers, both the main author and the coauthor, employ self-citation practices. The evaluation reveals the extent to which the main authors credit their previous publications and highlights the degree to which the ideology of the main authors is acknowledged by the coauthors. The objective here is to determine whether the main authors or the coauthors are more likely to reference their previous publications.

Implementation:

For the purpose of this analysis, two data frames were created. The first data frame consisted of publications and their respective authors and was identical to the data frame displayed in Table 5.9. The second data frame consisted of all publications, their authors, the position of the authors, and all the other publications that were referenced by these publications. The schema of the second data frame is shown in Table 5.11.

The *author_position* column featured 3 distinct values: "first," "middle," and "last". For this analysis, an assumption was made that the authors designated as "first" were the main authors of the publication, while the other positions were given to the respective coauthors. Following similar steps as explained in subsection 5.4.3.1, the total self-citations by the main authors and those by coauthors were calculated separately. This involved filtering the *author_position* column to first include only the main authors of the publications and then only the coauthors. The outcome of the analysis is shown in Figure 5.17. The snippet of the source code is displayed in Listing 5.16.

Column Name	Data Type	Description
work_id	String	The OpenAlex ID for publication
author_id	String	The OpenAlex ID for author
author_position	String	The position of the authors
publication_year	Integer	The year of publication
referenced_work_id	String	The OpenAlex ID for referenced publication

Table 5.11: Data frame with publication ID, author ID, position of authors, year of publication, and referenced publication ID.

```
#reading first parquet file
read_authorship = spark.read.parquet("D:\
    open_alex_parquet\works_authorships.parquet")
#reading second parquet file
read_citation = spark.read.parquet("D:\
    open_alex_parquet\self_citation.parquet")
# dropping duplicates if any
filter_citations = read_citation.dropDuplicates(["
    work_id","referenced_work_id"])
filter_authorship = read_authorship.dropDuplicates
    (["work_id","author_id"])
#selecting only the required columns
authors = filter_authorship.select("work_id","
    author_id")
```

```
citation = filter_citations.select("work_id","
    author_id","author_position","
    referenced_work_id")
#filtering only the main authors and the
    publication years between 1992 to 2022
main_authors = citation.filter( (lower(col("
    author_position"))=="first") & ((col("
    publication_year").between(1992,2022))))
#joining the dataframes
combined = main_authors.join(authors,main_authors.
    referenced_works == authors.work_id ,"inner")
#Counting self-citations
self_citation_counts=combined.withColumn("
    self_citation_count", when((combined.author_id
    == combined.referenced_author_id), lit(1)).
    otherwise(lit(0)))
#Displaying the results
self_citation_counts.show()
#filtering only the coauthors and the publication
    years between 1992 to 2022
coauthors = citation.filter((lower(col("
    author_position"))!="first") & ((col("
    publication_year").between(1992,2022))))
#joining the dataframes
combined = coauthors.join(authors,coauthors.
    referenced_works == authors.work_id ,"inner")
#Counting self-citations
self_citation_counts=combined.withColumn("
    self_citation_count", when((combined.author_id
    == combined.referenced_author_id), lit(1)).
    otherwise(lit(0)))
#Displaying the results
self_citation_counts.show()
```

Listing 5.16: Source code for self-citations single-authored vs coauthored per year

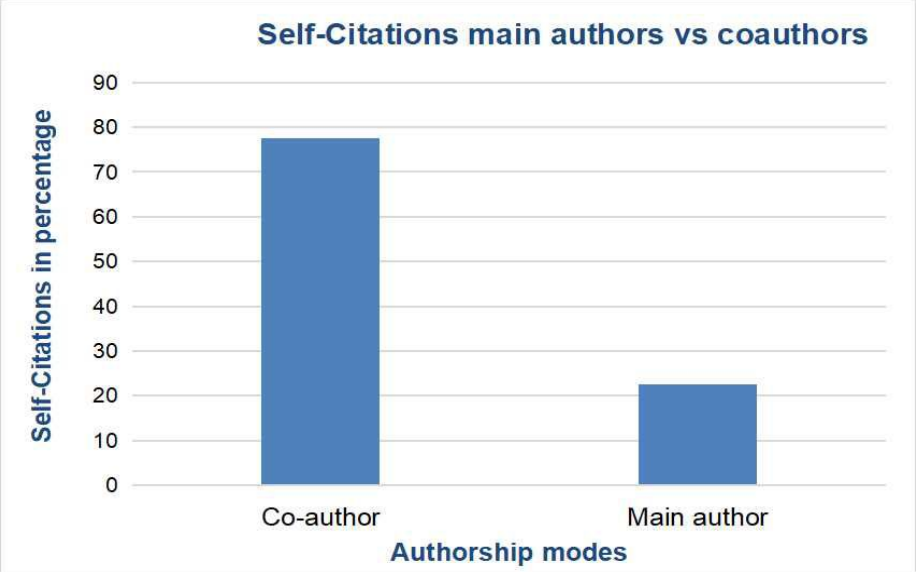


Figure 5.17: Self-citations single-authored vs coauthored per year

Discussion:

From the graph displayed in Figure 5.17, it becomes apparent that the coauthors refer to their publications at a higher rate than the main authors. The coauthors are responsible for almost 80% of self-citations, while the main authors contribute to almost only 20% of self-citations. This observation aligns with the authorship trend discussed in section 5.4.2.10, where collaborative publication with the joint effort of multiple authors is gaining more popularity than single-authored publication. Thus, it follows logically that the publications are being self-cited more by the coauthors because they account for a larger, if not equal, proportion of authorship in any coauthored publications. Furthermore, this pattern of self-citations also means greater recognition of the essential foundations of the main authors, which serves as a basis to carry out further research within these foundations.

5.4.4 Analysis of research fields:

Analyzing the research fields, which in the context of this thesis work refers to disciplines such as Computer Science, Physics, Chemistry, and more, provides a comprehensive understanding of scholarly works within different domains and provides insight into the trajectory of research works. This analysis helps identify the dominant research fields both in terms of the quantity of research work being performed within the research fields and the overall influence of the research fields. It serves as a tool that helps researchers understand emerging concepts and key topics while also facilitating decision-making regarding research priorities and resource allocation.

In this thesis work, the analysis takes two approaches. Initially, the top 5 dominant research fields based on the total publications were identified, and further, the total number of publications in these research fields for each year was evaluated. Subsequently, the top 5 dominant research fields based on the total number of citation counts were revealed, and the total number of citations for each year in these research fields was calculated.

5.4.4.1 Dominant research fields according to total publications:

Investigating the dominant research fields allows for tracking the most active area of study. It facilitates institutions and other funding organizations for efficient resource allocation towards the fields with higher academic publications. Additionally, the knowledge can be used for benchmarking and evaluating the productivity of the researchers and departments.

Implementation:

To determine the top 5 research fields according to publications count, the total number of publications in all the individual research fields was calculated. Following that, the 5 research fields with the highest publication counts were selected.

For this analysis, two data frames were generated. The first data frame contained all the publications, the research fields they belong to, and the title of the research field. The schema of the first data frame is shown in Table 5.12. The second data consisted of the publications and their publication years and is consistent with the data frame displayed in Table 5.8. The second data frame was first filtered to include the years between 1992 and 2022. Thereafter, a

join was established between the two data frames with column *work_id* from the first data frame and column *id* from the second data frame as the key columns. Exploiting the resulting dataset, every individual research field was then grouped according to the ID of the research field. The total number of publications for each research field was calculated by counting the publications in every research field. Finally, the top 5 research fields with the highest number of publications were selected. The result is shown in Figure 5.18. The snippet of the source code is displayed in Listing 5.17.

Column Name	Data Type	Description
work_id	String	The OpenAlex ID for publication
concept_id	String	The OpenAlex ID for research field
display_name	String	The title of the research field

Table 5.12: Data frame with publication ID, research field ID, and title of research field.

```
#reading first parquet file
read_works = spark.read.parquet("D:\
    open_alex_parquet\works.parquet")
#reading second parquet file
read_works_concepts = spark.read.parquet("D:\
    open_alex_parquet\works_concepts.parquet")
# dropping duplicates if any
filter_works = read_works.dropDuplicates(["id"])
filter_works_concepts = read_works_concepts.
    dropDuplicates(["work_id","concept_id"])
#selecting only the required columns
works = filter_works.select("id","publication_year
    ")
```

```
works_concepts = filter_works_concepts.select("work_id","concept_id","display_name")
#filtering the publication years between 1992 to 2022
select_years = works.select("id","publication_year").filter(col("publication_year").between(1992,2022))
#joining the dataframes
combined = works_concepts.join(select_years, works_concepts.work_id == select_years.id ,"inner")
#counting the total publications and displaying the result
combined.groupBy("concept_id","display_name").count().orderBy(col("count").desc()).show()
```

Listing 5.17: Source code for top 5 research field according to publications count

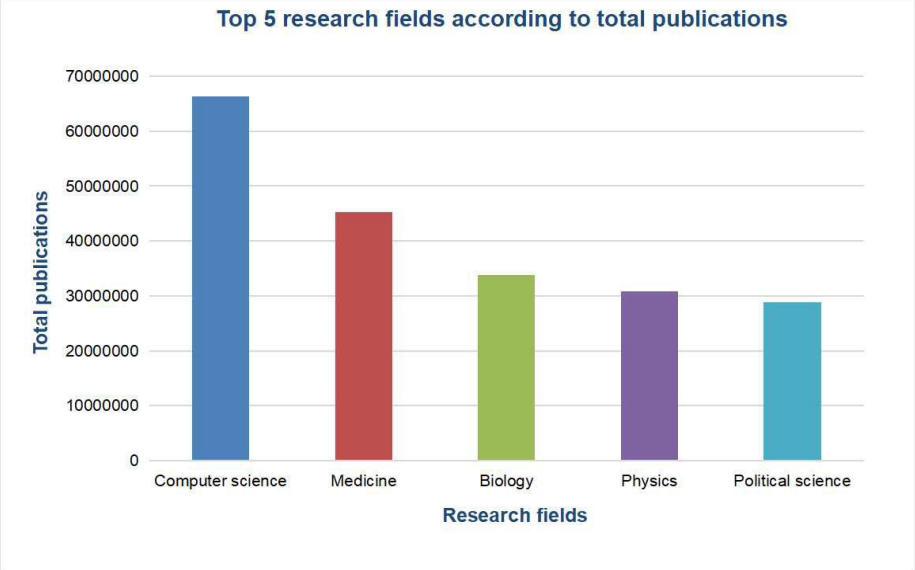


Figure 5.18: Top 5 research fields according to total publications

Discussion:

Figure 5.18 displays the top 5 research fields according to the publication count. It is clear that Computer Science is the most active research field, boasting over 65 million publications, followed by Medicine, with publication counts exceeding 45 million publications, and Biology, with more than 33 million publications. The fourth and fifth place are held by Physics and Political Science, with publication counts surpassing 30 million and 28 million publications, respectively. The reason for the highest number of publications in these research fields could be because of their alignment with events and technological advancements that attract greater attention. Availability of funding and resources could be other main reasons that boost productivity in these research fields.

5.4.4.2 Total publications per year in the dominant research fields:

In this Section, the analysis of publications over the years is carried out for the dominant research fields previously identified in Section 5.4.4.1. This analysis is of paramount importance as it helps identify the shifts in these areas of research. Utilizing this metric, an understanding can be developed about the progression of the publications within these research fields over the years that made them gain more scholarly interest amongst other research fields.

Implementation:

For this analysis, two data frames were created. The first data frame contained all the publications, the research fields they belong to, and the title of the research field. The second data consisted of the publications and their publication years. The schemas of the data frames are identical to the data frames displayed in Tables 5.8 and 5.11. The operations performed on the data frames are similar to the operations performed in Section 5.4.4.1. The difference is that for this task, the dataset was filtered only to include the 5 concepts determined in Section 5.4.4.1, and instead of counting the total number of publications for individual research fields, the publications were counted for each research field for each year in order to track the publication trajectory. The result of the analysis is displayed in Figure 5.19. The snippet of the source code is displayed in Listing 5.18.

```
#reading first parquet file
read_works = spark.read.parquet("D:\
    open_alex_parquet\works.parquet")
#reading second parquet file
read_works_concepts = spark.read.parquet("D:\
    open_alex_parquet\works_concepts.parquet")
# dropping duplicates if any
filter_works = read_works.dropDuplicates(["id"])
filter_works_concepts = read_works_concepts.
    dropDuplicates(["work_id","concept_id"])
#selecting only the required columns
works = filter_works.select("id","publication_year
    ")
works_concepts = filter_works_concepts.select("
    work_id","concept_id","display_name")
#making a list of top 5 dominant research fields
concepts_list=["https://openalex.org/C17744445","
    https://openalex.org/C71924100","https://
    openalex.org/C41008148",
"https://openalex.org/C121332964","https://
    openalex.org/C86803240"]
#filtering the publication years between 1992 to
    2022
select_years = works.select("id","publication_year
    ").filter(col("publication_year").between
    (1992,2022))
#selecting only the top 5 concepts
top_concepts = works_concepts.select("work_id","
    concept_id").filter(col("concept_id").isin(
    concepts_list))
#joining the dataframes
combined = works_concepts.join(select_years,
    works_concepts.work_id == select_years.id ,"
    inner")
#counting the total publications and displaying
    the result
```

```
21 combined.groupBy("concept_id","publication_year").
       count().orderBy(col("concept_id"),col("
       publication_year").asc()).show()
```

Listing 5.18: Source code for publications per year in the top 5 research fields

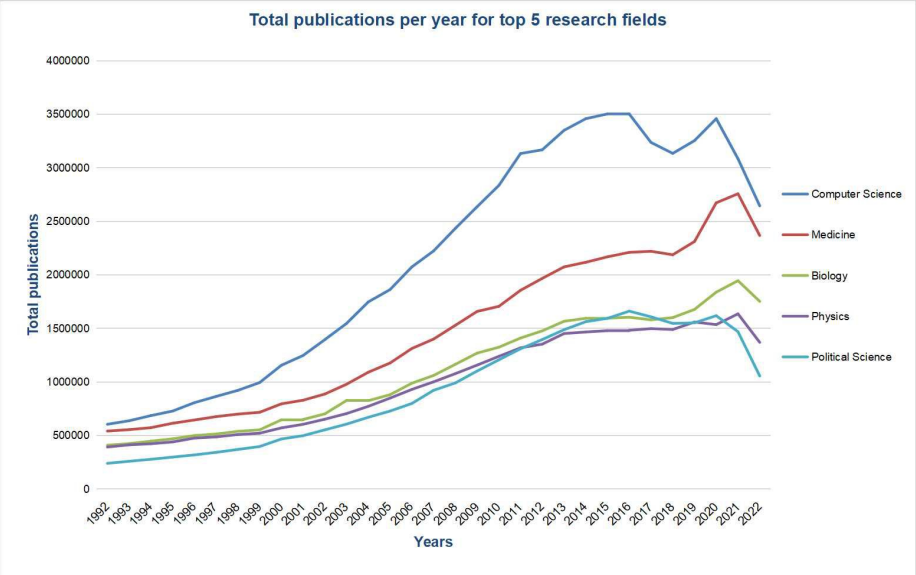


Figure 5.19: Publications per year in the top 5 research fields

Discussion:

Based on the findings presented in Figure 5.19, it can be observed that the top 2 research fields, Computer Science, and Medicine, maintained their position steadily throughout the years. In contrast, fluctuating publications can be seen between the other three research fields. Even though Biology and Physics started off with almost equal publications in 1992 (Physics - 391975 and Biology - 409009), Biology took the lead, gradually surpassing Physics. Another interesting observation is the publications in Political Science. Although it had the least publications compared to the other four research fields, the publications were progressively increasing to the extent that it overtook Physics in the year 2013 and even outpaced Biology for a short while during the year 2016. These shifts in publication trends in these top 5 dominant research fields might be the result of the emerging concepts within the field and the changes in fund

allocation. Further, the growing global collaboration among researchers could have had some impact on increasing the total number of publications.

5.4.4.3 Dominant research fields according to total citations:

Citation count is a metric for measuring the impact of the publications. Focusing on the dominant research fields with the highest citation counts, insight can be gained regarding the most impactful and influential area of study. This type of analysis is beneficial in making decisions about resource allocation and research priorities.

Implementation:

To identify the top 5 research fields based on the total number of citations they received, two data frames were generated. The first data frame contained all the publications and the research fields they belong to. The second data frame consisted of the publications, their publication years, and other publications that these publications cited. Both the data frames are consistent with the data frames previously displayed in Tables 5.8 and 5.11. The second data frame was first filtered to include the years between 1992 and 2022. Following this, a join was established between the two data frames with column *work_id* from the first data frame and column *referenced_work_id* from the second data frame as the key columns. Using the results from the joined data frames, every individual research field was then grouped according to the title of the research field. The total number of citations for each research field was calculated. Finally, the top 5 research fields with the highest number of citations were selected. The result is shown in Figure 5.20. The snippet of the source code is displayed in Listing 5.19.

```
#reading first parquet file
read_works_concepts = spark.read.parquet("D:\
    open_alex_parquet\works_concepts.parquet")
#reading second parquet file
read_citation = spark.read.parquet("D:\
    open_alex_parquet\citation.parquet")
# dropping duplicates if any
filter_works_concepts = read_works_concepts.
    dropDuplicates(["work_id","concept_id"])
filter_citations = read_citation.dropDuplicates(["
    work_id","referenced_work_id"])
#selecting only the required columns
citation = filter_citations.select("work_id","
    publication_year","referenced_work_id")
works_concepts = filter_works_concepts.select("
    work_id","concept_id","display_name")
#filtering the publication years between 1992 to
    2022
citation_years = citation.filter(col("
    publication_year").between(1992,2022))
#joining the dataframes
combined = citation_years.join(works_concepts,
    citation_years.referenced_work_id ==
    works_concepts.work_id ,"inner")
#counting the total publications and displaying
    the result
combined.groupBy("concept_id","display_name").
    count().orderBy(col("count").desc()).show()
```

Listing 5.19: Source code for top 5 research field according to citations count

Discussion:

Figure 5.20 exhibits the top 5 research fields based on total number of citations. The most influential research field was Biology, with citation counts over 580 million, followed by Medicine, with citations surpassing 540 million. The final three places were held by Chemistry, Computer Science, and Physics, with the total number of citations beyond 430 million, 370 million, and 350 million, re-

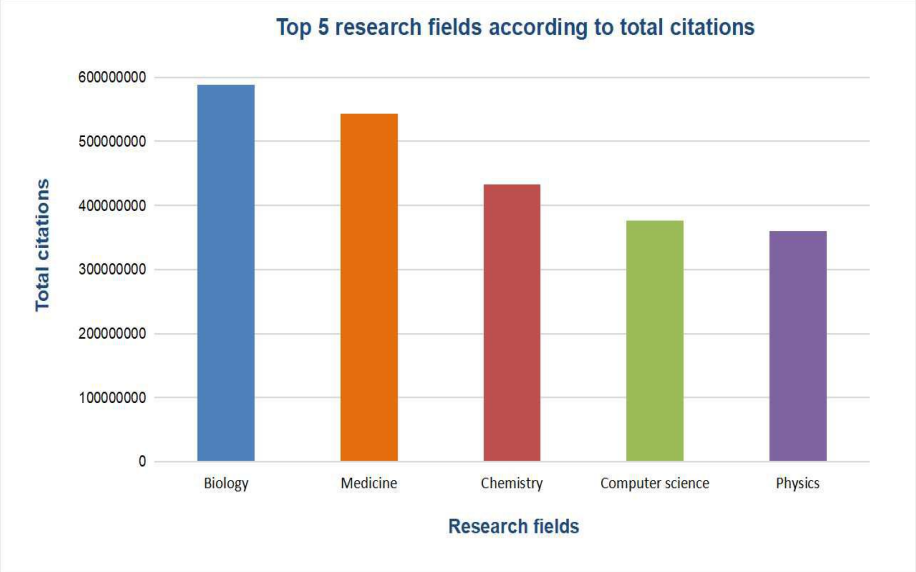


Figure 5.20: Top 5 research fields according to total citations

spectively. It is interesting to note that Computer Science, which was the most dominating research field in terms of total publication according to Figure 5.18, was in fourth place regarding the citations it received. Medicine maintained its second position firmly, both in terms of publications and citations. The research field, Political Science, which was one of the dominating research fields in reference to the total publications, has been replaced by Chemistry. Furthermore, Physics went from fourth position in regards to total publications to last in terms of total citations. Thus, carefully examining both figures underscores that a higher number of publications does not necessarily mean a higher level of influence.

5.4.4.4 Total citations per year in the dominant research fields:

Analyzing the total citations per year in the dominant research fields provides perspective on the impact that these research fields have had over the years. For this analysis, the dominant research fields previously determined in Section 5.4.4.3 have been exploited. This metric helps in evaluating the long-term influence of research within these dominating research fields. Further, it assists in decision-making about resource allocation and prioritization of research.

Implementation:

The data frames and the operations on the data frames in conducting this analysis were almost equivalent to those performed in Section 5.4.4.3. The primary difference lay in the process of filtering the dataset, where, for this analysis, the dataset was filtered only to include the 5 concepts determined in Section 5.4.4.3. Moreover, instead of summing up the citations received by individual research works, the citations were counted for each year for each individual research field. The result of the analysis is displayed in Figure 5.21. The snippet of the source code is displayed in Listing 5.20.

```
#reading first parquet file
read_works_concepts = spark.read.parquet("D:\
    open_alex_parquet\works_concepts.parquet")
#reading second parquet file
read_citation = spark.read.parquet("D:\
    open_alex_parquet\citation.parquet")
# dropping duplicates if any
filter_works_concepts = read_works_concepts.
    dropDuplicates(["work_id","concept_id"])
filter_citations = read_citation.dropDuplicates(["
    work_id","referenced_work_id"])
#selecting only the required columns
citation = filter_citations.select("work_id","
    publication_year","referenced_work_id")
works_concepts = filter_works_concepts.select("
    work_id","concept_id","display_name")
#making a list of top 5 dominant research fields
concepts_list=["https://openalex.org/C185592680","
    https://openalex.org/C71924100","https://
    openalex.org/C41008148","https://openalex.org/
    C121332964","https://openalex.org/C86803240"]
#filtering the publication years between 1992 to
    2022
citation_years = citation.filter(col("
    publication_year").between(1992,2022))
#selecting only the top 5 concepts
```

```
top_concepts = works_concepts.select("work_id","
    concept_id").filter(col("concept_id").isin(
    concepts_list))
#joining the dataframes
combined = citation_years.join(works_concepts,
    citation_years.referenced_work_id ==
    works_concepts.work_id ,"inner")
#counting the total publications and displaying
    the result
combined.groupBy("concept_id","publication_year").
    count().orderBy(col("concept_id"),col("
    publication_year").asc()).show()
```

Listing 5.20: Source code for citations per year in the top 5 research fields.

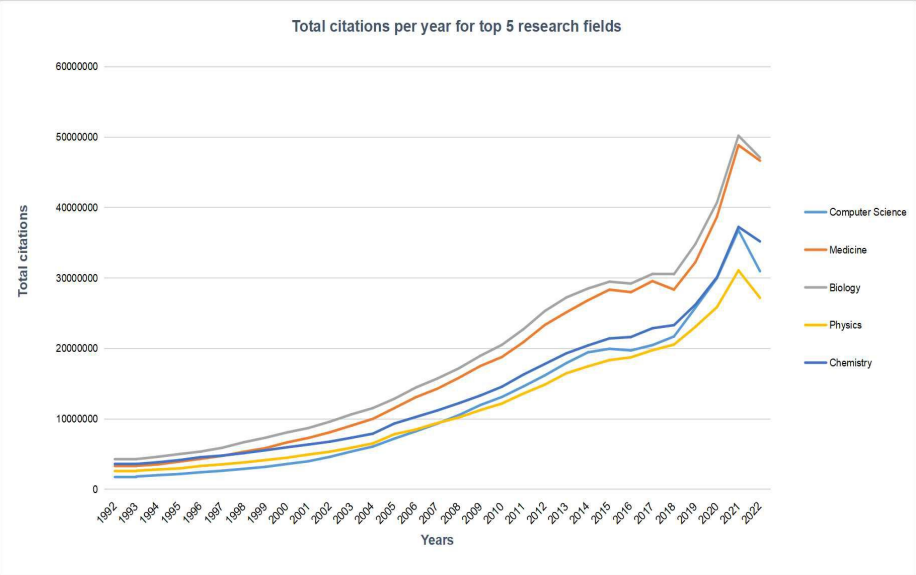


Figure 5.21: Citations per year in the top 5 research fields

Discussion:

According to the graph presented in Figure 5.21, there was a closely contested competition among all 5 research fields up until 2001, after which the gap between the first two research fields and the last three research fields gradually

started to widen. One interesting pattern to note here is that the trajectory of citation patterns between Biology and Medicine and between Chemistry and Physics were very much identical. As for Computer Science, despite having the least number of citations during the initial years, it progressively surpassed the citations received by Physics by the year 2009, and by 2019, the citations were almost equal to that of Chemistry. A conclusion that can be drawn from this citation pattern is that even though the subjects of natural science stand on top in terms of citations received per year, the ongoing pattern implies that Computer Science could potentially overtake all the other research fields, becoming the top research field with the highest number of citations per year.

5.4.5 Contribution of institutions towards publications:

Institutions are the entities that drive the creation of scholarly works. They support researchers by providing them with the necessary infrastructures and resources. Additionally, they also host their own academic works and share them with academic communities. OpenAlex lists about 8 such types of institutions, namely, Education, Health Care, Facility, Government, Company, Nonprofit, Other, and Archive.

The analysis of institutions is carried out in two parts. Initially, the contribution of each institution type is examined by calculating the total publications published by these institution types. Thereafter, the total publications per year by the top 5 institutions according to total publications are studied. The following subsections are dedicated to explaining these analyses.

5.4.5.1 Publications associated with different institutions:

The analysis of the publications by different institutions offers a perspective on the contribution of these institutions to the creation and dissemination of research works. The analysis of publications can also be considered a metric to rank an institution's global position. Thus, a higher number of publications can influence and attract researchers, students, and even funding agencies. Further, institutions with a higher number of publications might have a higher pool of researchers, potentially leading to a likelihood of producing quality outputs.

Implementation:

In order to perform this analysis, 3 data frames were created. The first data frame consisted of publications and their year of publication and is consistent with the data frame shown in Table 5.8. The second data frame consisted of publications and the institutions they were associated with. Similarly, the third data frame consisted of institutions and the type they belonged to. The schemas of the second and third data frames are shown in Tables 5.13 and 5.14. Initially, exploiting the first data frame, the publication year was filtered only to include years between 1992 and 2022. Thereafter, the second data frame was refined such that it did not contain any *Null* values for institutions. These two data frames were then joined together with *id* from the first data frame and *work_id* from the second data frame as the key columns. The combined data frame was again joined with the third data frame with columns *institution_id* from the combined data frame and column *id* from the third data frame as key columns. Finally, the resulting dataset was grouped according to the type of institution, and the total publications published by each of the institution types were counted. The result is shown in Figure 5.22. The snippet of the source code is displayed in Listing 5.21.

Column Name	Data Type	Description
work_id	String	The OpenAlex ID for publication
institution_id	String	The OpenAlex ID for institutions

Table 5.13: Data frame with publication ID, and institution ID.

Column Name	Data Type	Description
id	String	The OpenAlex ID for institutions
type	String	The type of institution

Table 5.14: Data frame with institution ID and type of institution.

```
#reading first parquet file
read_works = spark.read.parquet("D:\
    open_alex_parquet\works.parquet")
#reading second parquet file
read_authorship = spark.read.parquet("D:\
    open_alex_parquet\works_authorships.parquet")
#reading third parquet file
read_institutions =spark.read.parquet("D:\
    open_alex_parquet\institutions.parquet")
# dropping duplicates if any
filter_works = read_works.dropDuplicates(["id"])
filter_authorship = read_authorship.dropDuplicates
    (["work_id","institution_id"])
filter_institution = read_institutions.
    dropDuplicates(["id"])
#selecting only the required columns
works = filter_works.select("id","publication_year
    ")
authorship = filter_authorship.select("work_id","
    institution_id")
institution = filter_institution.select("id","type
    ")
#filtering the publication years between 1992 to
    2022
```

```
16 select_years = works.select("id","publication_year
       ").filter((col("publication_year").between
       (1992,2022)))
17 #filtering to exclude NULL values from column "
       institution_id"
18 filter_null_values = authorship.select("work_id","
       institution_id").filter((col("institution_id").
       isNotNull()))
19 #joining the dataframes
20 joined_works_authorship = select_years.join(
       filter_null_values,select_years.id ==
       filter_null_values.work_id,"inner")
21 #performing second join
22 joined_works_institution = joined_works_authorship
       .join(institution,joined_works_authorship.
       institution_id ==  institution.id,"inner")
23 #counting the total publications and displaying
       the result
24 joined_works_institution.groupBy("type").count().
       orderBy(col("type").desc()).show()
```

Listing 5.21: Source code for Porportion of publications associated to different institutions.

Discussion:

The graph depicted in Figure 5.22 portrays that almost 70% of the total publications were associated with Education institutions. This dominance can be attributed to the fact that most of the research works are conducted utilizing the facilities provided by the educational institutions. As highlighted by Slowe [41], there are substantial roles that institutions play that help further nourish and boost publications.

In contrast, other institutions, such as Government institutions, had comparatively fewer publications associated with them because their publications comprise mostly official and administrative documents. Similarly, only 0.29% of the publications were associated with Archives, largely because their publications mainly include historical records. One of the main reasons why Education

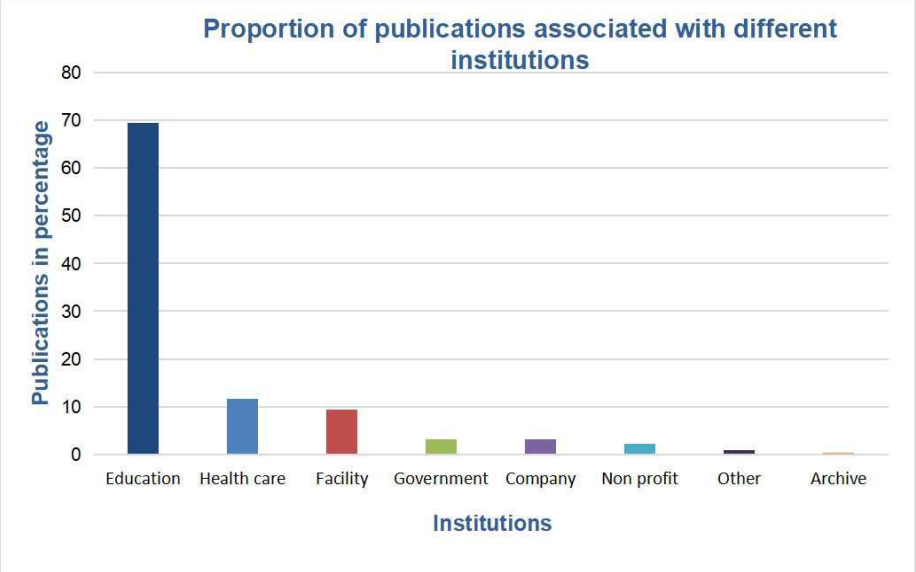


Figure 5.22: Porportions of publications associated with different institutions

institutions were the most productive is because of the higher number of researchers they have as compared to any other sector.

5.4.5.2 Total publications associated with dominant institutions per year:

Institutions play a very crucial role in the creation and distribution of publications. The analysis of yearly publications by institutions sheds light on the contribution of institutions to the growth and advancement of the world of publication. Further, a comparative analysis is conducted to determine what proportion of the total publications were published by the institutions.

Implementation:

In order to examine the yearly publications by the top 5 institutions, three data frames were constructed. The data frames are identical to the ones exploited in sub-section 5.4.5.1.

Initially, the first data frame was filtered only to include publications between 1992 and 2022. Subsequently, the second data frame was refined such that the

institution did not contain any *Null* values. Thereafter, the third data frame was filtered to only include the top 5 institutions among the 8 institutions previously studied in Section 5.4.5.1. The data frames were then joined together with column *id* from the first data frame and column *work_id* from the second data frame as the key columns. The combined data frame was then grouped according to the type of institution and the publication year, and the total publications were counted for each year. Since there was a huge difference in the total publications associated with different institutions, each individual institution with their total publications per year was displayed separately. The final results of the analysis are displayed in Figures 5.23, 5.24, 5.25, 5.26, and 5.27. The snippet of the source code is displayed in Listing 5.22.

```
#reading first parquet file
read_works = spark.read.parquet("D:\
    open_alex_parquet\works.parquet")
#reading second parquet file
read_authorship = spark.read.parquet("D:\
    open_alex_parquet\works_authorships.parquet")
#reading third parquet file
read_institutions =spark.read.parquet("D:\
    open_alex_parquet\institutions.parquet")
# dropping duplicates if any
filter_works = read_works.dropDuplicates(["id"])
filter_authorship = read_authorship.dropDuplicates
    (["work_id","institution_id"])
filter_institution = read_institutions.
    dropDuplicates(["id"])
#selecting only the required columns
works = filter_works.select("id","publication_year
    ")
authorship = filter_authorship.select("work_id","
    institution_id")
institution = filter_institution.select("id","type
    ")
#filtering the publication years between 1992 to
    2022
```

```
select_years = works.select("id","publication_year
    ").filter((col("publication_year").between
    (1992,2022)))
#filtering to exclude NULL values from column "
    institution_id"
filter_null_values = authorship.select("work_id","
    institution_id").filter((col("institution_id").
    isNotNull()))
#making a list of top 5 institutions
list_institution =["education","facility","
    healthcare","company","government" ]
#filtering to include only the top 5 institutions
filter_institutions = institution.select("id","
    type").filter((col("type").isin(
    list_institution)))
#joining the dataframes
joined_works_authorship = select_years.join(
    filter_null_values,select_years.id ==
    filter_null_values.work_id,"inner")
#performing second join
joined_works_institution = joined_works_authorship
    .join(filter_institutions,
    joined_works_authorship.institution_id ==
    filter_institutions.id,"inner")
#counting the total publications and displaying
    the result
joined_works_institution.select("*").groupBy("type
    ","publication_year").count().orderBy(col("type
    "),col("publication_year").asc()).show()
```

Listing 5.22: Source code for total publications associated to institutions per year.

Discussion:

Comparing the graph displayed in Figures 5.23, 5.24, 5.25, 5.26, and 5.27, it can be seen that the total publications were steadily rising every year for all types of institutions. Interestingly, the graphs representing the total publications per year for different institution types were very similar to one another. However,

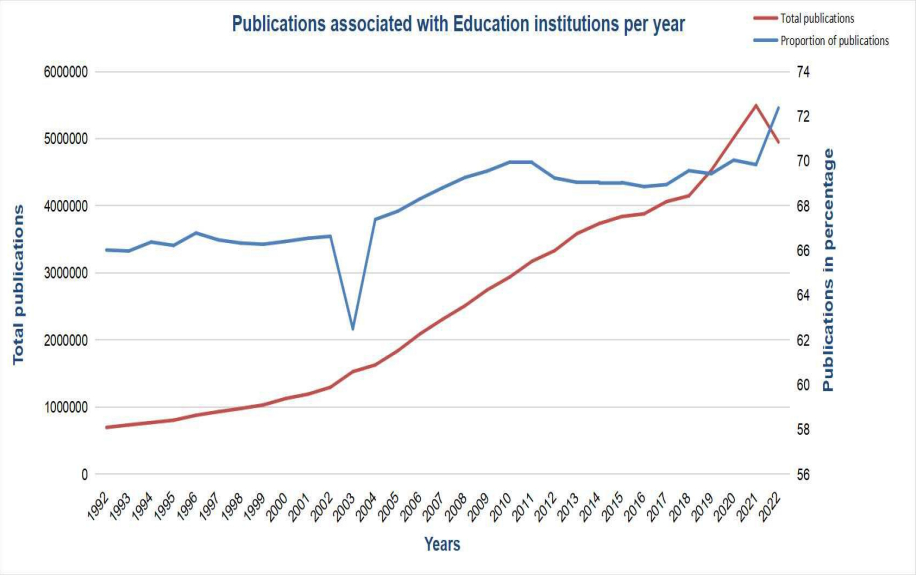


Figure 5.23: Total publications and proportion of overall publications associated with Education institutions per year

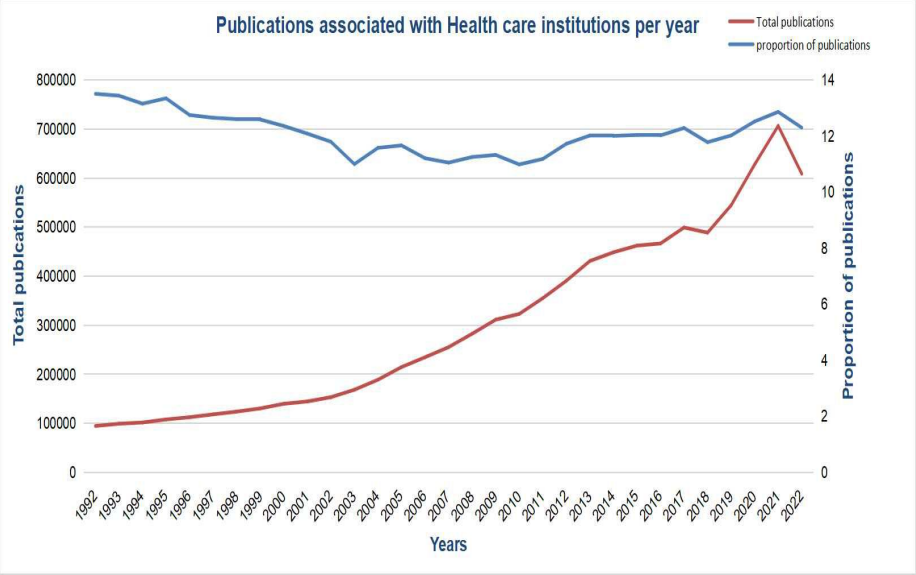


Figure 5.24: Total publications and proportion of overall publications associated with Health care institutions per year

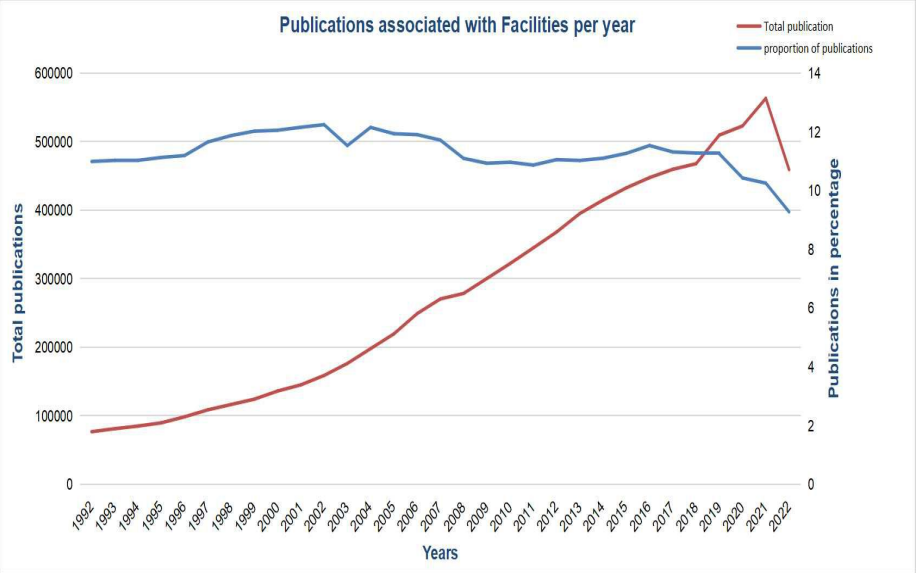


Figure 5.25: Total publications and proportion of overall publications associated with Facilities per year

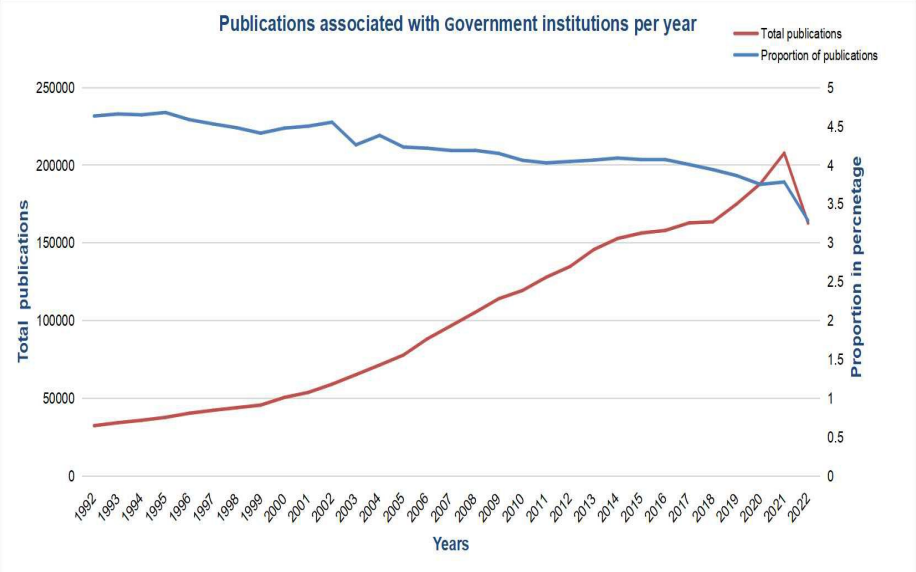


Figure 5.26: Total publications and proportion of overall publications associated with Government institutions per year

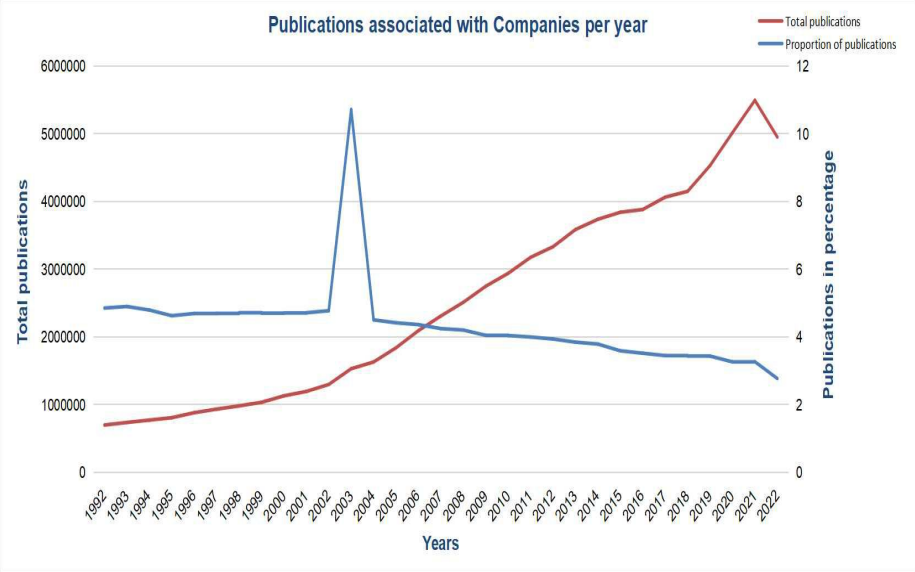


Figure 5.27: Total publications and proportion of overall publications associated with Companies per year

the difference lay in the proportion of overall publications per year. Although the total publications associated with these different institutions were steadily increasing, fluctuations can be seen in the proportion of overall publications per year.

The outcomes from the graphs imply that Education institutions stand out as the only institutions that experienced a rising proportion of overall publications, increasing from almost 66% in 1992 to almost 72% in 2022. As for the Companies, the proportion of overall publications was gradually declining except for a significant surge in 2003, from almost 5% in 2002 to almost 11% in 2003, which was the highest proportion of overall publications associated with Companies within the time period of 30 years. An interesting thing to note here is that in the year 2003 when the proportion of overall publications rose for Companies, every other institution faced a sharp downfall. Similarly, for Facilities, the proportion was relatively stable at almost around 12% until 2019, after which the proportion started to decline. The Government institutions also experienced diminishing proportions of overall publications from almost 5% in 1992 to almost 3% in 2022. Likewise, even though the Health institutions were facing a similar decrease in the proportion, they did manage

to increase the proportion gradually after 2010; however, the percentage was still less compared to the initial year(almost 14% in 1992 to about 12% in 1992).

In conclusion, the Education institutions had the highest number of publications associated with them both in terms of total publications per year and proportions of overall publications per year. The sudden increase and sudden decrease in the total publications and proportion of overall publications during the recent years, i.e., 2021 - 2022, for all the institutions, could be ambiguous due to the potential data update issues in the OpenAlex repository. Nonetheless, based on the results of previous years, it is safe to assume that the proportion of publications associated with institutions in recent years is also considerably high.

5.4.6 Contribution of sources towards publications:

This section is dedicated to the exploration of different types of sources and the role they play in hosting the publication. OpenAlex lists 4 types of sources, namely, Journals, Conferences, Ebook platforms, and Repositories. Understanding the publication pattern in these diverse sources helps researchers get an insight into the choice of platform for hosting their publications.

This analysis is performed in two parts. Firstly, the total publications in various types of sources are analyzed. Secondly, the publications per year hosted by the sources are examined. However, unlike the other time series analysis where the publication pattern was analyzed for the years between 1992 and 2022, for the sources, only the data from the last 10 years, i.e., 2012 to 2022, is available.

5.4.6.1 Proportion of publications hosted by different sources:

The study of the proportion of publications hosted by different sources reflects the diversity of the distribution of publications. It offers perspective on how widely the publications are distributed amongst various types of sources. A higher number of publications on a type of source could indicate greater reputability and a broader readership.

Implementation:

To conduct this analysis, a data frame was created that contained all the sources, their types, and the number of publications they host. The schema of the data frame is shown in Table 5.15. Initially, the sources were grouped based on the type they belonged to. Then, the total number of publications they hosted was summed in order to determine the total number of publications for individual source types. The result of the analysis is displayed in Figure 5.28. The snippet of the source code is displayed in Listing 5.23.

Column Name	Data Type	Description
id	String	The OpenAlex ID for sources
type	String	The type of source
works_count	Long	Total number of publications hosted by sources

Table 5.15: Data frame with source ID, type of source, and the total number of publications hosted by sources.

```
#reading the parquet file
read_sources = spark.read.parquet("D:\
    open_alex_parquet\sources.parquet")
# dropping duplicates if any
filter_sources = read_sources.dropDuplicates(["id"
    ])
#selecting only the required columns
sources = filter_sources.select("id","type","
    works_count")
#counting the total publications and displaying
    the result
```

```
8 sources.groupBy("type").sum("works_count")
```

Listing 5.23: Source code for proportion of publications hosted by different source.

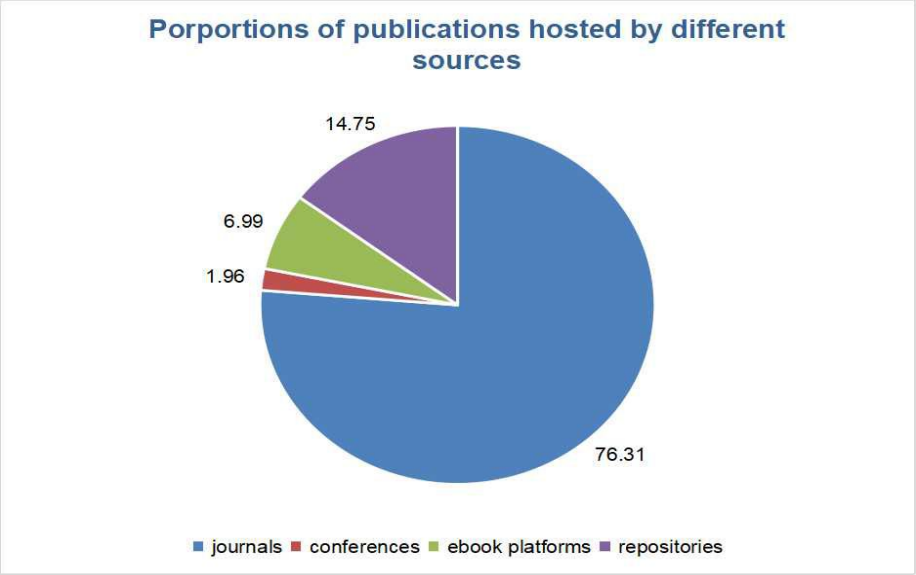


Figure 5.28: Proportion of publications hosted by different sources

Discussion:

According to the pie chart displayed in Figure 5.28, Journals were the most popular sources that hosted almost 77% of the total publications. They are followed by Repositories with 14.75%, Ebook platforms with 6.99%, and Conferences with only about 1.96%. Notably, the three types of sources (Conferences, Ebook platforms, and Repositories) collectively hosted less than a quarter of total publications. Thus, it can be concluded that Journals are the most favored choice of authors for hosting their works.

5.4.6.2 Proportion of publications hosted by different sources per year:

Understanding the total publications per year hosted by different sources helps track the trends and shifts in the distribution of publications across different types of sources. This metric provides a perspective through which investors and funding agencies can direct their investments toward the sources that are most utilized. At the same time, publishers and researchers can utilize this metric to adapt their publication strategy by publishing the work where the demand and impact are maximum.

Implementation:

Two data frames were constructed for this analysis. The first data frame consisted of the sources and the type they belong to and is consistent with the data frame displayed in Table 5.15. The second data frame comprised the sources, their year of publication, and the total number of publications they hosted. The schema of the second data frame is shown in Table 5.16.

To begin with, both the data frames were joined together with column *id* from the first data frame and column *source_id* from the second data frame as the key columns. The resulting dataset was then grouped based on source type and then by year of publication. The column *works_count* was then summed in order to calculate the total publications per year for different types of sources. Finally, the publications hosted by different sources each year were divided by the total publications hosted during respective years to calculate the proportion of publications each year. Since there was a huge difference between the proportion of publications for different sources, each individual source with their total publications per year was displayed separately. The final results of the analysis are displayed in Figures 5.29, 5.30, 5.31, and 5.32. The snippet of the source code is displayed in Listing 5.24.

Column Name	Data Type	Description
source_id	String	The OpenAlex ID for sources
year	Integer	The year of publication
works_count	Long	Total number of publications hosted by sources

Table 5.16: Data frame with source ID, year of publication, and the total number of publications hosted by sources.

```
#reading first parquet file
read_sources = spark.read.parquet("D:\
    open_alex_parquet\sources.parquet")
#reading second parquet file
read_sources_years = spark.read.parquet("D:\
    open_alex_parquet\sources_counts_by_year.
    parquet")
# dropping duplicates if any
filter_sources = read_sources.dropDuplicates(["id"
    ])
filter_sources_years = read_sources_years.
    dropDuplicates(["source_id"])
#selecting only the required columns
sources = filter_sources.select("id","type")
sources_years = filter_sources_years.select("
    source_id","type","works_count")
#joining the dataframes
combined = sources.join(sources_years,sources.id
    == sources_years.source_id ,"inner")
#counting the total publications and displaying
    the result
```

```
14 combined.groupBy("type","year").sum("works_count")
     .orderBy(col("type"),col("year").asc())
```

Listing 5.24: Source code for total publications hosted by different sources per year.

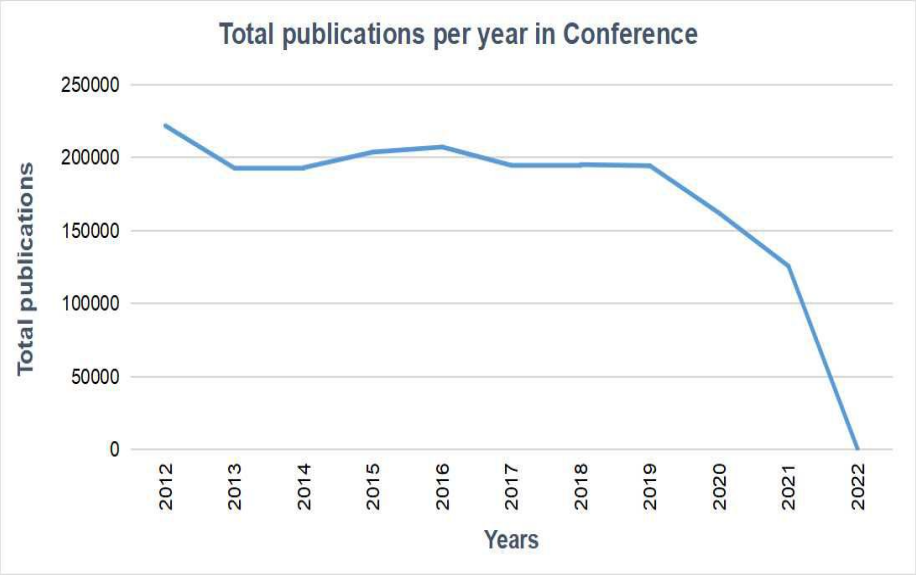


Figure 5.29: Total publications hosted by Conference per year

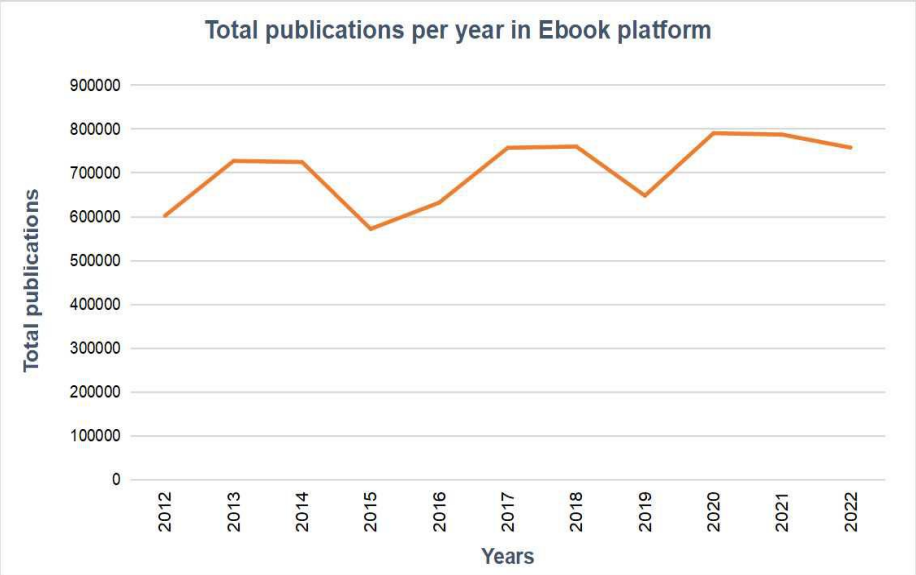


Figure 5.30: Proportion of publications hosted by Ebook platform per year

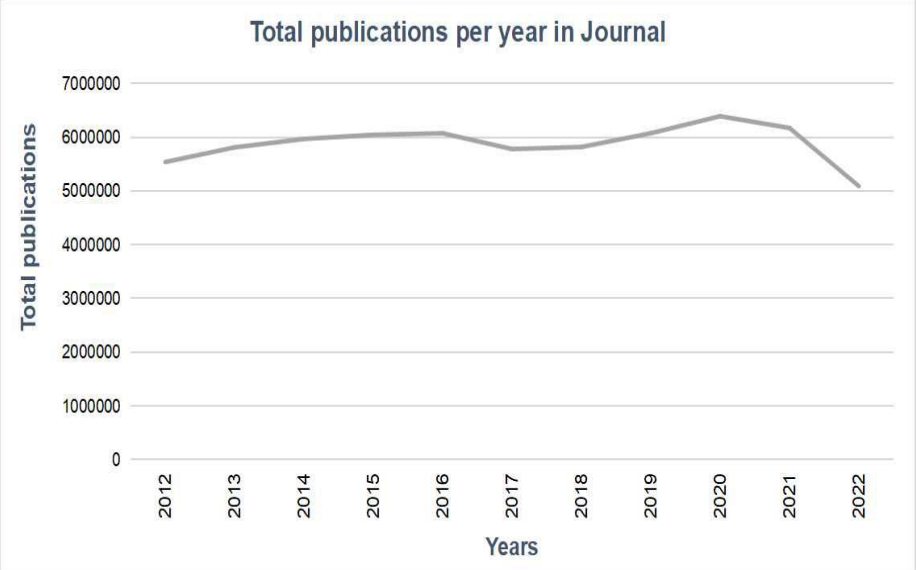


Figure 5.31: Proportion of publications hosted by Journal per year

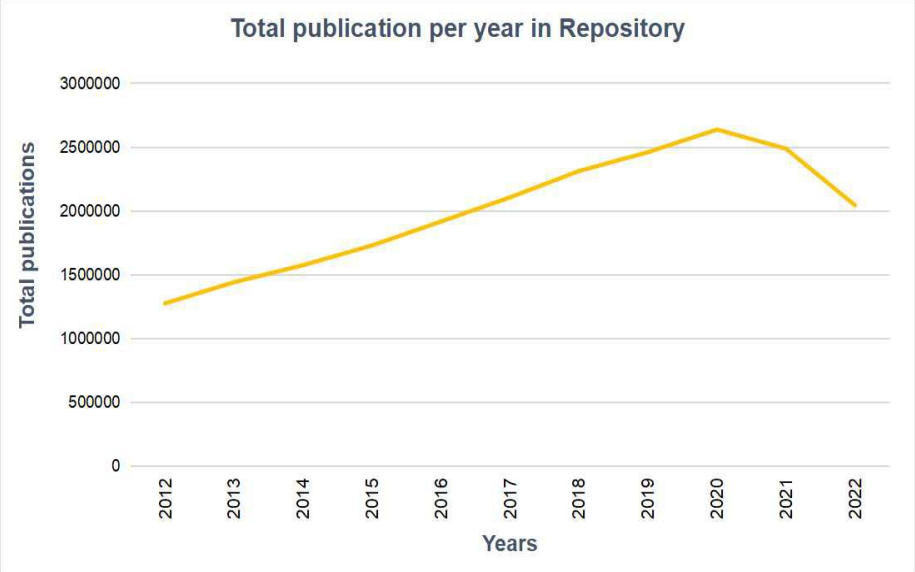


Figure 5.32: Proportion of publications hosted by Repository per year

Discussion:

A comparative analysis of graphs displayed in Figures 5.29, 5.30, 5.31, and 5.32 indicate an interesting finding. Despite the fact that Journals hosted the largest proportion of publications, as discussed in Section 5.4.6.1, the proportion of publications per year in Journals was steadily declining. Similarly, publications per year in Conferences displayed almost no change between the years 2013 and 2019, followed by a significant decrease. However, even though the publications per year in Journals were declining, it still had the highest proportion of publications over the years compared to other source types.

Likewise, publications in Ebook platforms displayed an interesting pattern with a sudden increase and decrease and almost a steady proportion of publications in between. On the other hand, the growth rate of publications in Repositories demonstrated a consistent rise.

In conclusion, the authors and the publishers seem to vary their interest in the type of sources to host the publications. The overall analysis reveals that Repositories have been gaining popularity over the years. While conflicting decisions about publishing in Ebook platforms are portrayed by the graph, the choice of publishing in journals and Conferences was decreasing. The increasing

publication rates across different source types can be attributed to digital revolutions and open-access movements that have made creating and distributing publications easier. The gradual downward shift in total publications in recent years would be due to the lack of updates in the OpenAlex.

5.4.7 Contribution of publishers towards publications:

Publishers play a pivotal role in the realm of publication. They are the intermediaries that publish the publications, thus acting as a bridge between authors and readers. Publishers are responsible for maintaining the quality of publications, handling copyrights, and providing editorial services. Accessing the effectiveness of the publishers in their contribution towards publication can be crucial for authors in making decisions regarding the choice of publishers for their publications.

For this thesis work, the analysis of publishers is carried out to determine the top 5 dominating publishers based on the total number of publications distributed.

One important point to consider is that just like explained in the previous section (see Section 5.4.6), similar limitations apply to publishers. The data is available for only the years between 2012 and 2022 is available. The analysis is done based on data provided within this timeframe.

5.4.7.1 Dominant publishers based on total publications distributed:

Analyzing the dominant publishers according to the total publications they have distributed provides information about publishers with considerable influence in disseminating the publications. It helps researchers understand which publishers have the highest impact in their domain field. This analysis could also form a base for further analysis regarding the global reach of the top publishers.

Implementation:

A data frame with all the publishers, their display names, and the total number of publications they have distributed was constructed to determine the top 5 publishers based on their distribution of total publications. The schema of the data frame is shown in Table 5.17.

The data frame was grouped according to the publisher ID and display name. The number of publications distributed was then summed to calculate the total number of publications distributed by the individual publisher. The result of the analysis is displayed in Table 5.18. The snippet of the source code is displayed in Listing 5.25.

Column Name	Data Type	Description
id	String	The OpenAlex ID for publishers
display_name	String	The title of publishers
works_count	Long	Total number of publications distributed by publishers

Table 5.17: Data frame with publisher ID, name of publishers, and the total number of publications distributed by publishers.

```
#reading the parquet file
read_publishers = spark.read.parquet("D:\
    open_alex_parquet\publishers.parquet")
# dropping duplicates if any
filter_publishers = read_publishers.dropDuplicates
    (["id"])
#selecting only the required columns
publishers = filter_publishers.select("id","
    display_name","works_count")
#counting the total publications and displaying
    the result
publishers.groupBy("id","display_name").sum("
    works_count").orderBy(col("sum(works_count)").
    desc()).show()
```

Listing 5.25: Source code for top 5 publishers based on total publications distributed.

Publisher	Total Publication
RELX Group	20364936
Johns Hopkins University	210951
University of Toronto	189747
Czech Academy of Sciences	62525
Academy of Sciences of the USSR	34516

Table 5.18: Total number of publications distributed by publishers.

Discussion:

From the data displayed in Table 5.18, it is clear that RELX Group was the top publisher, distributing almost 20.5 million publications. Interestingly, there was a distribution gap of over 20 million between the top publisher and the remaining 4 publishers. Nevertheless, the result set seems very uncertain because the USSR, which was dissolved during the early 1990s, is still the dominant publisher as per the data in the OpenAlex repository. A comparison with the findings presented in the recent article by Nishikawa-Pacher [42] on the 100 largest scientific publishers, based on journal count, reveals that only the first two publishers, Elsevier by RELX Group and John Hopkins University, align with those rankings. Although the total publications in this research work encompass all the publications and not just journals, the majority of the publications are hosted by the journals as discussed in section 5.4.6.1. The higher emphasis on journals should, therefore, assist journal publishers in attaining the top position. However, the research results obtained from the

OpenAlex dataset indicate otherwise. Therefore, it can be inferred that there might be incomplete information about publishers in the OpenAlex repository.

Due to the vague results generated regarding the publishers. Further analysis, such as publications per year distributed by the publishers or any other analysis regarding the publishers was not carried out.

5.4.8 Calculation of h-index

The h-index is the measure that assesses the impact and productivity of authors. The metric involves a thorough analysis of the author's publications and citation counts received by the publications. It is determined by first arranging the author's publication in descending order and finding the rank of the publication where the citation count is equal to or more than the position of the publication. This metric is exploited to gain an understanding of the author's influence in the publication realm.

In the interest of optimizing processing time and resource efficiency, this study focuses exclusively on the h-index within the field of Computer Science. The objective is to analyze the impact of researchers and their contributions within this specific domain.

5.4.8.1 Average h-index of authors in Computer Science per year:

The average h-index of authors in Computer Science provides a meaningful perspective into the collective impact and productivity of authors within this research field. It helps measure the overall influence of the authors, reflecting both the quantity and quality of their research works. A higher average h-index indicates that, on average, authors within Computer Science have made significant contributions to the field. This metric can also be used to assess the overall influence of research in Computer Science, thus aiding strategic decision-making regarding funds and resource allocation.

Implementation:

To perform this analysis, a data frame was constructed, which consisted of the publications, their authors, the research field they belong to, the year of publications, and the number of citations they received thus far. The schema of the data frame is displayed in Table 5.19.

The data frame was initially filtered to include only the years for which the h-index is to be calculated, and the column *concept_id* was filtered to include only Computer Science. Thereafter, the dataset was partitioned according to the author, ordered according to citation counts in descending order, and a row number was provided. Following that, the h-index was calculated using the condition that if the citation count is greater than or equal to the row number, the h-index is the row number; otherwise, it is 0. Finally, the dataset was filtered only to include the h-index higher than 0 and then grouped according to the *concept_id* to calculate the average h-index. The result of the analysis is displayed in Figure 5.33. The snippet of the source code is displayed in Listing 5.26.

Column Name	**Data Type**	**Description**
work_id	String	The OpenAlex ID for publications
author_id	String	The OpenAlex ID for authors
concept_id	String	The OpenAlex ID for research fields
publication_year	Integer	The year of publication
cited_by_count	Long	The total number of citations received

Table 5.19: Data frame with publication ID, author ID, research field ID, year of publication, and citation counts.

```
#reading the parquet file
read_h_index = spark.read.parquet("D:\
    open_alex_parquet\index.parquet")
# dropping duplicates if any
filter_h_index = read_h_index.dropDuplicates(["
    work_id","author_id"])
#selecting only the required columns
h_index = filter_h_index.select("work_id","
    author_id","concept_id","publication_year","
    cited_by_count")
#filtering to include only the required years and
    selecting computer science as research field
selecting_subject = h_index.select("*").filter((
    col("publication_year").between(0,2020))  & (
    col("concept_id")=="https://openalex.org/
    C41008148"))
#creating a window by partioning according to
    authors and ordering by number of citations
    received
windowSpec = Window.partitionBy("author_id").
    orderBy(col("cited_by_count").desc())
# adding a new column with row numbers
row_num=selecting_subject.withColumn("row_number",
    row_number().over(windowSpec))
#calculating h-index
h_index_count=row_num.withColumn("h-index", when((
    row_num.cited_by_count >= row_num.row_number),
    row_num.row_number ).otherwise(lit(0)))
#calculating average h-index and displaying the
    result
h_index_count.select("*").filter(col("h-index")>0)
    .groupBy("concept_id").avg("h-index").show(
    truncate=False)
```

Listing 5.26: Source code for average h-index of authors in computer science per year.

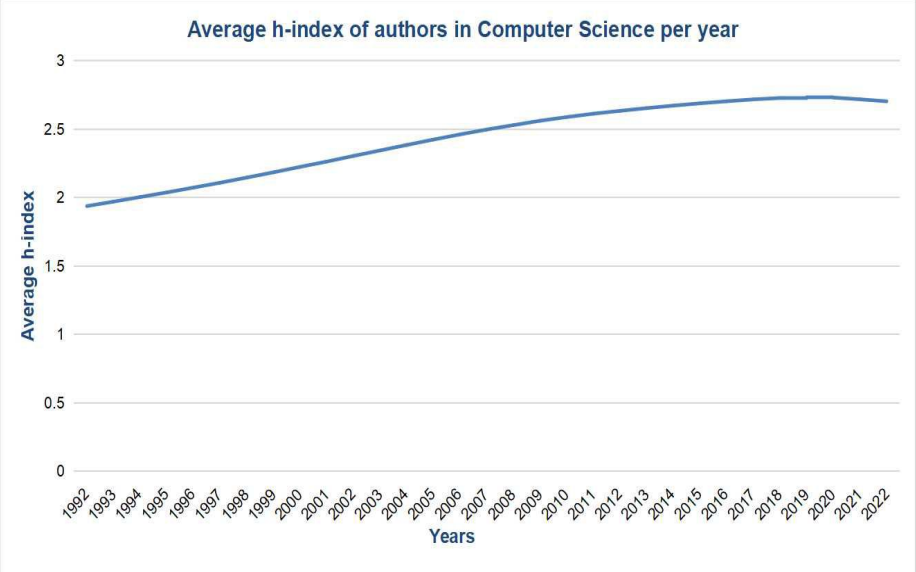


Figure 5.33: Average h-index of authors in Computer Science per year

Discussion:

According to the graph displayed in Figure 5.33, the average h-index for authors in Computer Science has demonstrated a positive trend, showing a consistent increase over the period 1992 - 2022. This increase might be attributed to the impactful publications every year and increasing citation rates. On the other hand, self-citation could also be one of the reasons for the increasing h-index. When authors consistently cite their own papers, it can influence the h-index. The slight decrease in the average h-index during the recent years(2021-2022) is due to the missing information in the OpenAlex repository.

Although the h-index is used as a metric to assess the author's impact, authors can themselves manipulate the metric through means of self-citations. Therefore, the h-index should be interpreted with caution, considering both self-citations and external citations.

6 Conclusion

In conclusion, this thesis work explored the world of bibliographic data within the OpenAlex repository, utilizing the Apache Spark tool. Through the exploration of the dataset and application of bibliographic metrics, valuable insight regarding the patterns, trends, and impact of the scholarly publication has been discovered.

The study began with the significance of Bibliometric Analysis and how various bibliographic metrics can be utilized to uncover trends. Throughout the study, various dimensions of the bibliographic data have been analyzed and Apache Spark has allowed for the efficient processing of such massive data.

6.1 Summary

In this section, the key takeaways of the findings have been summarized.

- **Publications:**

  Over the analyzed period from 1992 to 2022, there has been a noticeable increment in the overall publication, from about 2.2 million in the year 1992, reaching a highest of more than 10 million in the year 2020. When compared to the publications during the period 1981 to 2001 and 2002 to 2022, a significant upward trend in all types of publications was discovered. Notably, within this rise, there is also a gradual surge in openly accessible publications, with open-access publications rising from almost 2 hundred thousand in 1992 to almost 4 million in 2022, signifying the shift towards openness and greater accessibility of scholarly publications.

- **Authors:**

  A steady increase was found in the number of authors. The total number of authors, as per the result increased from about 2.4 million in the year 1992 to more than 15 million in the year 2022. The growth was consistent with the overall rise in the publication. Interestingly, the trend towards multi-authored publications has been increasing since the past decade, denoting a collaborative trend in publication.

- **Citations and self-citations:**

  Another major finding of the thesis work was regarding the citations and self-citations. It was found that alongside the increase in publications, the citations received by the publications were also on the rise, from almost 10.4 million in 1992 to about 116 million in 2022, indicating the recognition and influence of scholarly works. Furthermore, the impact of accessibility on citations could be clearly seen with the growing number of citations on openly accessible publications. In addition, multi-authored publications were likely to receive more citations than single-authored publications, highlighting the influence of collaboration in research.

  Similarly, the self-citation counts, although fluctuating, an overall increasing trend was observed; meaning an increase in the likelihood of authors to self-reference their own previous publication. The proportion of self-citations rose from less than 20% in 1992 to reaching a maximum of almost 25% in 2018. Interestingly, the co-authors of the publications were found to self-cite the publication more compared to the main authors, thus representing a collaborative approach to boost the visibility of the publications.

- **Interconnections between publications, authors, and citations:**

  To determine the relationship between publications, authors, and citations; four metrics were calculated "average publications per authors", "average authors per publications","average citations per publications", and "average citations per authors". Each of the results displayed an increasing trend. The individual results can be summarized as follows:

  - The increase in the average publications per authors depicts the broader contributions of authors to publications.
  - The increase in the average authors per publications reveals the rise in collaborative research efforts.

  - The increase in the average citations per publications suggests the growing influence of publications.
  - The increase in average citations per authors indicate that the authors are gaining more recognition for their contribution.

- **Research fields:**

  Two distinct results emerged during the analysis of research fields. When considering the total publications, Computer Science emerged as the most dominating field, showcasing a massive number of over 66 million publications. On the other hand, the count of the total citations revealed that Biology was revealed to be the most dominating. Further, the analysis of total publications per year and total citations per year on the dominating research field in the respective domain revealed the trend of publication and influence within the research fields. These results indicate that the volume and impact of scholarly publications must be analyzed separately to get a clearer insight.

- **Contribution of institutions, sources, and publishers:**

  The study of the contribution of institutions, sources, and publishers provided a comprehensive overview of how knowledge is shared. Education institutions contributed majorly to publications, with almost 70% of the total institutional publications associated with educational institutions, while, Journals as a source hosted a substantial proportion, almost 77% of publications. The contribution of each type of source and institution provides a greater understanding of the dissemination of knowledge throughout the years. However, the analysis of publishers generated a vague result putting a halt to the further analysis.

- **H-index:**

  Finally, the calculation of the average h-index of authors in Computer Science has also showcased a notable increase. An average h-index of less than 2 was recorded in the year 1992 while the index grew to more than 2.5 in the year 2022. The upward trend not only indicates the productivity of authors but also the impact of their works.

Each of the findings mentioned above answers Research Questions 2 to 6, mentioned in Section 1.2 of this thesis work. The findings in the sections "Publications"; "Authors"; "Citations and self-citations (only the results of citation analysis) " and "Interconnections between publications, authors, and

citations", collectively address Research Question 2. The results emphasizing analysis of self-citation discussed in the section "Citations and self-citations" answer Research Question 3. Similarly, the findings in the section "Concepts" respond to Research Question 4. Finally, the findings in the sections "Contribution of institutions, sources, and publishers"; and "H-index" address the Research Questions 5 and 6 respectively. Meanwhile, Research Question 1 focuses on the broader aspect involving data preprocessing and the overall analysis of bibliographic data using Apache Spark.

6.2 Limitations

The analysis of bibliographic data within the OpenAlex repository using Apache Spark has provided valuable insight. However, it is also important to acknowledge the limitations of this study.

- **Data completeness:**

  Although OpenAlex is comprehensive, it may have limitations regarding the completeness of the data. The inadequate update of the dataset might have resulted in most of the graphs displayed in this study, exhibiting a steep downward or upward trajectory.

- **Loss of data:**

  In order to boost the processing efficiency of the analysis, the format of the dataset was converted from JSON to Parquet, which might have resulted in the loss of data during the conversion process.

- **Dataset dependency:**

  The foundation of this thesis relies on the dataset within the OpenAlex repository. Therefore, utilizing a different repository might yield a different result while performing a similar analysis, depending on the structure and richness of the data within the repository.

- **Scope of the analysis:**

  The emphasis of the analysis was mostly on the general aspect, for instance, authorship analysis and citation analysis. While it includes portions where only specific concepts are covered, a more detailed analysis, for instance, an analysis of publication for a certain region or an analysis of authors belonging to a certain country, has not been covered in this study.

- **Temporal constraints:**

  The study focuses largely on the time period between the years 1992 to 2022. Expanding the time frame will generate a different outcome.

- **Methodological limitation:**

  The entire analysis was conducted exclusively utilizing the unique identification (ID) provided by OpenAlex to its entities. However, discrepancies such as the same ID corresponding to a different name/title or vice versa have not been addressed in this thesis.

6.3 Future Works

The findings in this thesis work unveiled valuable insight into the dynamics of bibliographic data. The analysis can be further extended to explore the citation network which would offer a more comprehensive perspective on citation behavior. A more localized study can be conducted to study for instance publications and citation patterns for a specific country. Investigating the impact of funders in shifting the bibliographic dynamics could add more value to the analysis. Additionally, the machine learning tools within Apache Spark could be utilized to predict the trend in publication.

As the bibliographic data continues to grow, future research works could emphasize exploring new dimensions within bibliographic data, refine the process of conducting bibliometric analysis, and use different technologies to derive more insight from bibliographic data.

Bibliography

[1] Pubrica Academy. Journal publication process for research paper, Mar 2022. URL https://pubrica.com/academy/publication-support/journal-publication-process-for-research-paper/. Accessed on January 15, 2024.

[2] Wikipedia. H-index, Dec 2023. URL https://en.wikipedia.org/wiki/H-index. Accessed on January 15, 2024.

[3] Jason Priem, Heather Piwowar, and Richard Orr. Openalex: A fully-open index of scholarly works, authors, venues, institutions, and concepts. *arXiv preprint arXiv:2205.01833*, 2022.

[4] OpenAlex API Documentation. URL https://docs.openalex.org/download-all-data/snapshot-data-format. Accessed on September 18, 2023.

[5] Michael Fire and Carlos Guestrin. Over-optimization of academic publishing metrics: observing goodhart's law in action, gigascience, 8, giz053, 2019.

[6] Jacalyn Kelly, Tara Sadeghieh, and Khosrow Adeli. Peer review in scientific publications: benefits, critiques, & a survival guide. *Ejifcc*, 25(3):227, 2014.

[7] Richard Walker and Pascal Rocha da Silva. Emerging trends in peer review—a survey. *Frontiers in neuroscience*, 9:169, 2015.

[8] AIJR. Paper publishing process, Feb 2021. URL https://aijr.org/paper-publishing-process/. Accessed on January 15, 2024.

[9] Eugene Garfield. Is citation analysis a legitimate evaluation tool? *Scientometrics*, 1:359–375, 1979.

[10] Purdue University. Library: Citation analysis: What is citation analysis? URL https://library.pfw.edu/c.php?g=16182&p=88939. Accessed on January 15, 2024.

[11] Jorge Hirsch. An index to quantify an individual's scientific research output. *Proceedings of the National academy of Sciences*, 102(46):16569–16572, 2005.

[12] Lutz Bornmann and Hans-Dieter Daniel. What do we know about the h index? *Journal of the American Society for Information Science and technology*, 58(9):1381–1385, 2007.

[13] Alfio Ferrara and Silvia Salini. Ten challenges in modeling bibliographic data for bibliometric analysis. *Scientometrics*, 93(3):765–785, 2012.

[14] João Paulo Romanelli, Maria Carolina Pereira Gonçalves, Luís Fernando de Abreu Pestana, Jéssica Akemi Hitaka Soares, Raquel Stucchi Boschi, and Daniel Fernandes Andrade. Four challenges when conducting bibliometric reviews and how to deal with them. *Environmental Science and Pollution Research*, pages 1–11, 2021.

[15] Google cloud. What is big data?, . URL `https://cloud.google.com/learn/what-is-big-data`. Accessed on January 27, 2024.

[16] Seref Sagiroglu and Duygu Sinanc. Big data: A review. In *2013 international conference on collaboration technologies and systems (CTS)*, pages 42–47. IEEE, 2013.

[17] Google cloud. What is apache spark?, . URL `https://cloud.google.com/learn/what-is-apache-spark`. Accessed on January 30, 2024.

[18] Databricks. Apache spark. URL `https://www.databricks.com/spark/about`. Accessed on January 30, 2024.

[19] Shubhada Prashant Nagarkar and Rajendra Kumbhar. Text mining: an analysis of research published under the subject category 'information science library science'in web of science database during 1999-2013. *Library Review*, 64(3):248–262, 2015.

[20] Barbara Stefaniak. Use of bibliographic data bases for scientometric studies. *Scientometrics*, 12(3-4):149–161, 1987.

[21] Cuong Huu Nguyen, Loc Thi My Nguyen, Trung Tran, and Tien-Trung Nguyen. Bibliographic and content analysis of articles on education from vietnam indexed in scopus from 2009 to 2018. *Science Editing*, 7(1):45–49, 2020.

[22] Geoff Krause and Philippe Mongeon. Measuring data re-use through dataset citations in openalex. In *27th International Conference on Science, Technology and Innovation Indicators (STI 2023)*. International Conference on Science, Technology and Innovation Indicators, 2023.

[23] Eric Schares and Sandra Mierz. Using openalex to analyse cited reference patterns. In *27th International Conference on Science, Technology and Innovation Indicators (STI 2023)*. International Conference on Science, Technology and Innovation Indicators, 2023.

[24] Thomas Scheidsteger and Robin Haunschild. Comparison of metadata with relevance for bibliometrics between microsoft academic graph and openalex until 2020. *arXiv preprint arXiv:2206.14168*, 2022.

[25] Chenyue Jiao, Kai Li, and Zhichao Fang. How are exclusively data journals indexed in major scholarly databases? an examination of the web of science, scopus, dimensions, and openalex. *arXiv preprint arXiv:2307.09704*, 2023.

[26] Heather Piwowar, Lea Maria Ferguson, Antonia C Schrader, and Nina Weisweiler. Open science factsheet no. 9 based on the 64th online seminar: Openalex. 2022.

[27] Aliakbar Akbaritabar, Tom Theile, and Emilio Zagheni. Global flows and rates of international migration of scholars. Technical report, Max Planck Institute for Demographic Research, Rostock, Germany, 2023.

[28] José Luis Ortega and Lorena Joaquina Delgado Quirós. Retractions, retracted articles and withdrawals coverage in scholarly databases. In *27th International Conference on Science, Technology and Innovation Indicators (STI 2023)*. International Conference on Science, Technology and Innovation Indicators, 2023.

[29] MultiTech. Explain types of data file formats in big data through apache spark, Sep 2020. URL `https://informationit27.medium.com/explain-types-of-data-file-formats-in-big-data-through-apache-spark-669f812c75e4`. Accessed December 3, 2023.

[30] Dinesh Chopra. Unveiling the battle: Apache parquet vs csv-exploring the pros and cons of data formats, May 2023. URL `https://medium.com/@dinesh1.chopra/-the-battle-apache-parquet-vs--exploring-the-pros-and-cons-of-data-formats-b6bfd8e43107`. Accessed on December 3, 2023.

[31] Marek Kwiek. What large-scale publication and citation data tell us about international research collaboration in europe: changing national patterns in global contexts. *Studies in Higher Education*, 46(12):2629–2649, 2021. doi: `10.1080/03075079.2020.1749254`. URL `https://doi.org/10.1080/03075079.2020.1749254`.

[32] Peder Olesen Larsen and Markus von Ins. The rate of growth in scientific publication and the decline in coverage provided by science citation index,

Sep 2010. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2909426/.

[33] Jevin West, Jennifer Jacquet, Molly King, Shelley Correll, and Carl Bergstrom. The role of gender in scholarly authorship. 2013. URL https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0066212.

[34] Zehra Taşkın, Abdülkadir Taşkın, Güleda Doğan, and Emanuel Kulczycki. Factors affecting time to publication in information science - scientometrics, Feb 2022. URL https://link.springer.com/article/10.1007/s11192-022-04296-8#citeas.

[35] Ariel Rosenfeld and Shir Aviv-Reuven. Publication patterns' changes due to the covid-19 pandemic: A longitudinal and short-term scientometric analysis. URL https://pubmed.ncbi.nlm.nih.gov/34188333/.

[36] Dr. Andrew Plume and Dr. Daphne van Weijen. Publish or perish? the rise of the fractional author.... *Research trends*, 1(38):5, 2014.

[37] Fakhri Momeni, Philipp Mayr, Nicholas Fraser, and Isabella Peters. What happens when a journal converts to open access? a bibliometric analysis. *Scientometrics*, 126(12):9811–9827, 2021.

[38] Dorte Henriksen. The rise in co-authorship in the social sciences (1980–2013). *Scientometrics*, 107(2):455–476, 2016.

[39] Helmut Arthur Abt. Citations to single and multiauthored papers. *Publications of the Astronomical Society of the Pacific*, 96(583):746, 1984.

[40] John Ioannidis. A generalized view of self-citation: Direct, co-author, collaborative, and coercive induced self-citation. *Journal of psychosomatic research*, 78(1):7–11, 2015.

[41] Sarah Slowe. The role of the institution in scholarly publishing. *Emerging Topics in Life Sciences*, 2(6):751–754, 2018.

[42] Andreas Nishikawa-Pacher. Who are the 100 largest scientific publishers by journal count? a webscraping approach. *Journal of Documentation*, 78(7):450–463, 2022.