BERGISCHE
UNIVERSITÄT
WUPPERTAL

Fakultät für Elektrotechnik,
Informationstechnik und Medientechnik

Lehrstuhl für Data & Knowledge Engineering

Master-Thesis

Analyzing the State of Computer Science Research with the DBLP Discovery Dataset

Leon Lennart Küll
1549540

Informatik

Data Analytics

Wuppertal, den 10. November 2022

Betreuer	Dr. Terry Ruas	Erstgutachter	Prof. Dr.-Ing. Bela Gipp
	Jan Philip Wahle M.Sc.	Zweitgutachter	Dr. Terry Ruas

Eidesstattliche Erklärung

Hiermit erkläre ich, dass ich die von mir eingereichte Abschlussarbeit (Master-Thesis) selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt sowie Stellen der Abschlussarbeit, die anderen Werken dem Wortlaut oder Sinn nach entnommen wurden, in jedem Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

Wuppertal, den 10. November 2022 ____________________________

(Unterschrift)

Einverständniserklärung

Hiermit erkläre ich mich damit einverstanden, dass meine Abschlussarbeit (Master-Thesis) wissenschaftlich interessierten Personen oder Institutionen zur Einsichtnahme zur Verfügung gestellt werden kann.

Korrektur- oder Bewertungshinweise in meiner Arbeit dürfen nicht zitiert werden.

Wuppertal, den 10. November 2022 ____________________________

(Unterschrift)

Acknowledgements

I would like to express my deepest gratitude to the many people who have supported me throughout my thesis. Without them, this thesis would not have been possible.

First and foremost, I am incredibly grateful for my supervisors Dr. Terry Ruas and Jan Philip Wahle, who continually provided invaluable guidance throughout the entire time I worked on this thesis. I deeply appreciate their expertise and constructive feedback, which helped me tackle any issues that arose during the thesis.

I also want to thank Prof. Dr. Bela Gipp for offering to write my thesis at his chair and giving me the opportunity to complete a part of my research at the National Institute of Informatics (NII) in Tokyo. Thanks should also go to Dr. Norman Meuschke, who helped me organize and review my application for the DAAD scholarship at the NII.

Moreover, I would like to thank my hosting professor at the NII Prof. Dr. Akiko Aizwa for her warm welcome and helpful insights when developing my thesis project, even though I could not visit the NII in person, due to traveling restrictions the COVID-19 pandemic imposed.

I express special thanks to Dr. Saif M. Mohammad for his comments on the project during its development and for essentially shaping how the project turned out. Many thanks also go to my two fellow students Tom Neuschulten and Alexander von Tottleben for developing the prediction endpoint and the shared efforts to integrate it into the project.

Finally, I want to thank my family and friends for their continued support and encouragement during this very intense academic year.

Related Publications

The content of this master's thesis was created as part of an ongoing research project led by my supervisors Dr. Terry Ruas and Jan Philip Wahle. Parts of this project I also worked on (and thus parts of my thesis) were already published or are planned to be submitted to computer science conferences, which I list in the following:

The DBLP Discovery Dataset and its creation (Sections 4.1.2 to 4.1.3), and its implementation (Section 5.1.4).

> Wahle, Jan Philip; Ruas, Terry; Mohammad, Saif, and Gipp, Bela (June 2022). "D3: A Massive Dataset of Scholarly Metadata for Analyzing the State of Computer Science Research". In: *Proceedings of the Language Resources and Evaluation Conference*. Marseille, France: European Language Resources Association, pp. 2642–2651. URL: `https://aclanthology.org/2022.lrec-1.283`

The Computer Science Insights system (Section 4.3.2), its motivation (Section 1.1), and its architecture (Section 5.1).

> Ruas, Terry; Wahle, Jan Philip; Küll, Lennart; Mohammad, Saif M., and Gipp, Bela (Oct. 13, 2022). *CS-Insights: A System for Analyzing Computer Science Research*. DOI: `10.48550/arXiv.2210.06878`. URL: `http://arxiv.org/abs/2210.06878` (visited on 10/18/2022). Submission planned for EACL'23 (System Demonstrations)

In the following the wording "we" is used rather than "I" as the ongoing research project is a collaborative effort, and I worked closely together with my supervisors.

Abstract

The number of scientific publications continues to rise exponentially, especially in Computer Science (CS). However, our ability to analyze those publications does not follow the same speed, which prevents us from finding and understanding implicit patterns hidden in their metadata (e.g., venues, document types). Current solutions are limited by restricting access behind a paywall, offering no features for visual analysis, limiting access to their data, only focusing on niches or sub-fields, and/or not being flexible and modular enough to be transferred to other datasets.

In this thesis, we conduct a scientometric analysis to uncover those implicit patterns hidden in CS metadata and to determine the state of CS research. Specifically, we investigate trends of the quantity, impact, and topics for authors, venues, document types (conferences vs. journals), and fields of study (compared to, e.g., medicine). To achieve this we introduce the Computer Science Insights (CS-Insights) system, an interactive web application to analyze CS publications through multiple perspectives. The data underlying this system is the DBLP Discovery Dataset (D3), which contains metadata from 5 million scholarly publications in CS and their metadata. We create D3 with data from DBLP, the largest open-access bibliography for scientific papers and articles in CS, and enrich it with further metadata (e.g., abstracts, citations). CS-Insights offers dedicated dashboards with multiple visualizations for all main features of D3 (e.g., publications, authors, venues, and citations) and multiple filters for more fine-grained analysis. Both D3 and CS-Insights are open-access, and CS-Insights can be easily adapted to other datasets in the future.

The most interesting findings of our scientometric analysis include that i) there has been a stark increase in publications, authors, and venues in the last two decades, ii) many authors only recently joined the field, iii) the most cited authors and venues focus on computer vision and pattern recognition, while the most productive prefer engineering-related topics, iv) the preference of researchers to publish in conferences over journals dwindles, v) on average, journal articles receive twice as many citations compared to conference papers, but the contrast is much smaller for the most cited conferences and journals, and vi) journals also get more citations in all other investigated fields of study, while only CS and engineering publish more in conferences than journals.

Contents

1 Introduction

Chapter 1 first introduces the problem this thesis tries to solve by explaining the context and motivation (Section 1.1). We establish the goals of our research through the main research objective and its corresponding tasks and research questions (Section 1.2) and list our contributions to CS research (Section 1.3). Lastly, we outline the remainder of this thesis (Section 1.4).

1.1 Problem Presentation and Motivation

In the last few decades, we have seen an exponential rise in the number of digital scientific publications, while our ability to analyze them does not follow the same speed, preventing us from uncovering implicit patterns among its main features (e.g., authors, venues) (Bornmann et al. 2021). Analyzing these large amounts of publications, and possibly any type of data, is hard, mainly due to its storage and processing challenges. There are already existing solutions to mitigate this problem, but all of them show inherent limitations. Researchers can use tools or repositories that already implement data storage, crawling, and processing, like Google Scholar[1], Semantic Scholar[2] or DBLP[3], to find papers or authors and view their metrics, but these solutions lack details in other areas (e.g., venues) and options for analysis with visual components. Other solutions also provide visualizations (e.g., Scopus[4], Web of Science[5]), but are not open-access and are only available behind paywalls, which is prohibitive to those who would benefit the most from their resources (e.g., institutions in developing countries). Therefore, researchers focus on specific research areas, e.g., NLP Scholar (Mohammad 2020c) for Natural Language Processing (NLP). Areas without such tools rely on data repositories (e.g. arXiv[6]) or general tools (e.g., VOSViewer (van Eck and Waltman 2010)), which also only have a limited set of options for analysis and visualizations.

Analyzing the entire research landscape would be prohibitive (Google Scholar alone has more than 389m records (Gusenbauer 2019)), so we focus efforts on a specific field of research and conducting a case study on that field, with the goal of our methodology also applying to other areas. We decide CS is a great candidate for this, for two main reasons. First, the presence of CS in solving or facilitating other field-related problems is undeniable (e.g., plagiarism detection (Wahle et al. 2021) or media bias (Spinde et al. 2021)). Advancements in CS are also responsible for many benefits, e.g., faster systems, more accurate results, and efficient tools. Today there is hardly any area not affected by the vast possibilities of CS. Consider how difficult it would be to test, develop, and research new vaccines without access to tools of informatics (e.g., public

[1]https://scholar.google.com/
[2]https://www.semanticscholar.org/
[3]https://dblp.org/
[4]https://www.scopus.com/
[5]https://www.webofscience.com/
[6]https://arxiv.org/

repositories (Kousha and Thelwall 2020), artificial intelligence (Aggarwal et al. 2022)). Second, CS is a massively growing field, especially when compared to other fields, which becomes apparent when we look at the submissions on arXiv (Figure 1.1). We can not only see that the number of submitted CS papers increased over the last 10 years (from less than 10k papers a year in 2011 to 60k in 2021) but also how CS is taking up a much larger percentage of the total submissions (from 10% in 2011 to over a third of the submissions in 2021, the most of all research fields). While arXiv is not a peer-reviewed repository, the same increase in CS submissions is also visible for repositories consisting of peer-reviewed publications (e.g., DBLP and Web of Science, as shown by Wahle et al. (2022) and Fiala and Tutoky (2017), respectively), and sub-fields of CS (e.g., NLP as shown by Mohammad (2020b)). Their studies also reveal that the number of authors has increased significantly over the last few decades.

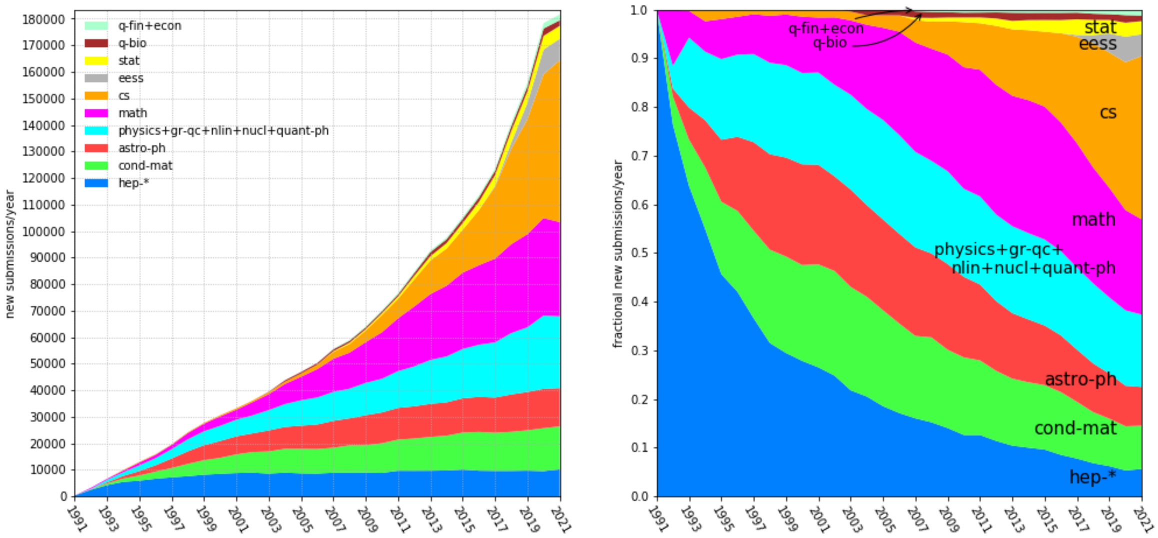


Figure 1.1 Submission rate statistics of arXiv 1991-2021; updated 1 January 2021 (arXiv 2022).

As a result, we see CS as a promising environment for developing a system to help understand its publications in an automated and democratic way and conducting an analysis with it, that can answer questions like: How fast is computer science research growing? How many authors are actively publishing in their field? What topics are prevalent in specific venues? Answering these questions helps other researchers and organizations make more informed decisions in their research and publications. Researchers can explore particular topics of interest for individual authors and venues, discover influential publications, or find important venues to inform their own research. Conference organizers and research organizations (e.g., ACL[7]) can track how their policy changes affect broad publication trends over time or compare the research output of authors and venues. For example, with a scientometric analysis one can track citation gaps across authors and venues; and in the future, uncover the influence of big technology companies, highly-funded universities, and governments.

[7]https://www.aclweb.org/

1.2 Research Objective

With our goal of an analysis of the state of computer science research, we define the main research objective as follows:

Analyze the state of computer science research by inspecting its different core components and uncovering implicit patterns.

From the main objective, we distill the following research tasks:

RT1 Review scientometric studies in the area of CS and tools/resources for scientometric analyses that already exist.

RT2 Collect, clean, organize, store, and publish data from scientific publications in CS.

RT3 Develop a system that uses the collected data to create visualizations to facilitate quantitative analyses.

RT4 Analyze CS through its core components (e.g., publications, authors, venues, and document types), evaluate the findings, and compare them to that of previous research.

We also derive research questions from our main research objective to narrow down which specific questions we have to answer to determine the state of CS research. Thus, the research questions will guide our analysis in the later parts of this thesis and determine which experiments we will perform during RT4:

RQ1 How many publications, authors, and venues are in our dataset? How do the numbers change over time? How many authors and venues are currently active?

RQ2 How are the citations and publications distributed across authors and venues? How do the distributions change over time?

RQ3 What are the most prominent authors and venues? Are there preferences for topics? Do the topics change over time?

RQ4 How do incoming and outgoing citations evolve over time? How do their distributions differ?

RQ5 How do conferences and journals compare in their number of publications and citations over time? How do the top venues and topics differ? Do top authors prefer conferences or journals?

RQ6 How do the most prominent fields of study differ from CS in topics and preference for conferences or journals?

1.3 Contributions

Through our goals, this thesis provides multiple contributions to the CS research community:

- We propose the DBLP Discovery Dataset (D3), a large and carefully curated dataset of CS publications metadata, with metadata of 5m CS publications from DBLP, enriched with additional metadata like abstracts and citation counts.
- Also, we develop the Computer Science Insights (CS-Insights) system, a modular analysis platform, which allows its users to perform quantitative analyses on the core components of CS research.
- We use the dataset and system to conduct a scientometric analysis of the publications, authors, venues, document types, fields of study, and their topics and citations, according to our research questions (RQ1 to RQ6).
- Both the dataset and system are open-access and open-source to allow everyone, regardless of background, wealth, or institutional affiliation to use and reproduce our research. References for the dataset and system are given at the beginning of Section 4.1 and Section 4.3.2, respectively.

1.4 Outline

Chapter 1 introduced the current problem in CS research and research in general, which this thesis will work on (Section 1.1), proposed a solution to address the problem (Section 1.2), and listed our specific contributions (Section 1.3).

Chapter 2 introduces fundamental knowledge the reader needs to understand this thesis, including some technical aspects and concepts (e.g., topic modeling and scientometrics) we use in this thesis.

Chapter 3 addresses RT1, by investigating what scientometric studies other researchers perform and what data sources and tools they use (Section 3.1). We then review those data sources and tools and determine there is currently no free solution that can perform a quantitative analysis on CS research and thus show which gap CS-Insights and our research fill (Section 3.2).

Chapter 4 presents our methodology, which is composed of RT2 and RT3. We begin by explaining how we acquire our data from DBLP and how we enrich it with more metadata from full-texts (Section 4.1). Next, we cover how we store the data and make it accessible through an Application User Interface (API) (Section 4.2). The rest of the chapter is spent on detailing the User Interface (UI) and its features, i.e., dashboards, filters, and visualizations (Sections 4.3.1 to 4.3.2). We also include some showcases to demonstrate how the system can be used (Section 4.3.3)

Chapter 5 goes over some implementation details that relate to RT2 and RT3. We provide an overview of the CS-Insights system, its components and their architecture (Section 5.1), and some details on our measures for quality assurance (Section 5.2).

Chapter 6 addresses RT4 by conducting an extensive analysis, that follows the research questions given earlier in the chapter. We split each research question into multiple smaller experiments, each of which receives a discussion, that evaluates the findings and relates them to other experiments and previous research. Before starting with the experiments, we state the general setup of how we conduct them (Section 6.1). We then group the experiments by the main attribute they investigate (e.g., publications, authors, document types) and for each experiment include a figure or table that contains the results we then discuss (Sections 6.2 to 6.7). In the end, we provide a summary of the chapter that shows we cover each aspect of each research question (Section 6.8).

Chapter 7 presents the final considerations of this thesis. We list our contributions and most interesting findings and conclude CS is a very active and growing field, whose characteristics and trends a scientometric analysis with CS-Insights can uncover (Section 7.1). Finally, we present the limitations and future work of this thesis (Section 7.2).

Appendix A includes supplementary figures of the UI and some of its visualizations, which are referenced throughout this thesis.

2 Fundamentals

This chapter shortly introduces some basic concepts related to technical aspects (Section 2.1), topic modeling (Section 2.2), and scientometrics (Section 2.3).

2.1 Technical Aspects

The CS-Insights system uses a MongoDB[1] database to store its data (Section 4.2). MongoDB is a document-oriented database, and not a relational database (e.g., MySQL). The key difference is the way data is stored in a document-oriented database: data is stored in documents, while relational databases use tables with rows and columns. Thus, the terminology also changes, which Table 2.1 highlights. In document-oriented databases the schema is not fixed, so adding and removing fields is easier, which allows for quicker iterations during development. The documents in MongoDB use JSON's key-value pairs, but add support for more features (e.g., dates) (Bradshaw et al. 2019, pp. 3, 16–17; Győrödi et al. 2015).

MySQL	**MongoDB**
Database	Database
Table	Collection
Index	Index
Row	Document
Column	Field
Join	Lookup
Primary key	Primary key
Group by	Aggregation

Table 2.1 Differences of terms between MySQL and MongoDB; adapted from Győrödi et al. (2015).

We then provide access to this data through a REST API on our server (Section 4.2.2), i.e., an API, which is based on the four principles of Representational State Transfer (REST): identification of resources, manipulation of resources through representations, self-descriptive messages, and hypermedia as the engine of application state (Fielding 2000, p. 82). This means REST uses HTTP verbs (e.g., GET, POST, DELETE) to transfer data as representations in a well-defined media type (e.g., JSON, XML). The resources are identified through unique resources identifiers (URIs), e.g., `shop/products/` to perform operations on all products or `shop/products/42` for operations on product 42 (Kopecký et al. 2014). Create, Read, Update, Delete (CRUD) operations (Bradshaw et al. 2019, p. 14) can then be performed on our data through the REST API.

2.2 Topic Modeling

Topic models are statistical models, that cluster a group of words into meaningful "topics" from any unstructured text or text corpus (e.g., emails, book chapters, blog posts). Each document in a corpus is treated as a "bag of words", i.e., the location of words in the document, syntax, and narrative of the document are ignored. The models then use these bags to determine the co-occurrence of specific words across the corpus of bags and generate the distribution of words that refers to each topic. Topic modeling is an automated approach, so researchers only have to define the number

[1] https://www.mongodb.com/

of topics the corpus is supposed to be sorted into (Mohr and Bogdanov 2013). The idea behind this is, e.g., documents about cats often include "cat" and "meow" and documents about dogs "dog" and "woof", which show these words occur together in specific documents and relate to the same topic.

Latent Dirichlet Allocation (LDA) (Blei et al. 2003) is a generative probabilistic model and the most used topic modeling approach. It assumes each document contains multiple themes/topics the authors want to discuss. The document is then generated by the authors by repeatedly selecting a topic and a word from that topic and placing it in the bag of words of the document until the document is completed. Selecting the next topic is based on the distribution of topics across the documents and selecting the next word is based on the distribution of words across the selected topic. LDA then tries to infer the intents of the authors when generating the document by reverse-engineering the two distributions that are used to draw the topics for a document and words from a topic. Once both distributions are determined the most probable words for a specific topic can be used by humans to imagine the actual topics, as LDA cannot generate topic labels (Mohr and Bogdanov 2013).

The most probable words of specific topics can also include common words that are present across all topics, as they appear in many different scenarios (e.g., "paper" or "present" for scholarly articles). These generic words also appear in the list of most frequent terms of the entire corpus and explain little about the topic's contents or the corpus. Chuang et al. (2012) develop the ranking measure "saliency", which is supposed to filter a corpus's list of most frequent words and rank words higher that only appear in a few topics, and words lower that appear in many topics. They first define the distinctiveness of a word, which measures how informative the word is for determining the generating topic, e.g., the word "brain" would be informative, while "paper" would not. Saliency is then computed by weighing the distinctiveness of a word against the overall probability of that word in the corpus. The list of the most salient words would then rank "paper" lower than the list of the most frequent words, making it easier to find differences between topics.

2.3 Scientometrics

Scientometrics is the study of quantitative aspects of science and technology, i.e., exploration and evaluation of scientific research. It covers measuring the quality of research and its impact, tracking and understanding citations, mapping and visualizing scientific fields, and using these measures for policy and management decisions (e.g., by institutions). Quantitative measures employed are, e.g., the impact factor for venues (average amount of citations per publication per year of that venue) or the h-index for authors (h papers of the author have at least h citations). Bibliometrics is similar to scientometrics and uses statistical methods to analyze publications and books, e.g., citation graphs. Both scientometrics and bibliometrics are sub-fields of informetrics, which covers the study of all information as a whole, regardless of form or origin (Mingers and Leydesdorff 2015). An example of a scientometric study in

CS is Coşkun et al. (2019), which analyzes the trends over time regarding countries, document types, institutions, author collaboration, keywords, and journals. Fiala and Tutoky (2017) conduct a bibliometric study investigating the quantity and the impact of publications according to document types, languages, disciplines, countries, institutions, and publication sources. Both use similar approaches (e.g., by examining the distribution of document types or most publishing institutions and countries) even though one is called a scientometric study and one a bibliometric study, which shows the closeness of bibliometrics and scientometrics when analyzing only scientific publications. We will further explore the findings of both studies in Section 3.1.1.

3 Related Work

This chapter provides an overview of related (scientometric) studies and the existing resources that can be used to make these studies easier. We start by investigating previous scientometric studies in CS and their findings, where we also show what data and tools the authors use for their analyses (Section 3.1.1), and shortly look into unique approaches in NLP (Section 3.1.2). Then, we present the data sources those studies use and other available data resources to show their limitations, and which gap CS-Insights fills (Sections 3.2.1 to 3.2.2). We also show a few general tools that can aid researchers in scientometric studies and highlight their differences from CS-Insights (Section 3.2.3). Finally, we present selected tools from NLP, as some of these tools are very similar to what CS-Insights tries to achieve (Section 3.2.4).

3.1 Scientometric Studies

This section covers previous scientometric studies in CS (Section 3.1.1) and shortly explores scientometric studies in NLP (Section 3.1.2). We show many researchers rely on paid-access data and most researchers do not use any specific tool to automate their studies.

3.1.1 Scientometric Studies in Computer Science

In this subsection, we review the previous work of other researchers on CS research to see which analyses are done, prove useful, and should also be provided in CS-Insights. We cover broad studies of CS, studies on topics and terms, conferences vs. journals, and lastly some studies on the differences between CS and other research fields.

Broad Studies on Computer Science Research

First, we look into two broad studies on CS research that investigate many areas of CS research (e.g., publications, authors, venues, and citations). Coşkun et al. (2019) use data from the Web of Science core collection to perform a scientometrics-based study of CS and Information Science research by looking at two periods (2008-2013 and 2014-2019) with 57,347 and 96,219 documents respectively and comparing the results to discover trends. They look into the document types and find that there are slightly fewer conference papers in the second period, but the overall amount of documents increases, as there are more journal articles and documents from other types. Conference papers make up most documents in both periods, but the gap between papers and articles closes over time. The top journals come mostly from engineering and other technical sub-areas (e.g., from IEEE or IEICE), which also reflects in the top research areas, as they also show a focus on technical and engineering-related issues. Lastly, they use networks from VOSViewer[1] to investigate the recurrence of keywords, which reveals a shift to current issues, such as privacy, security, IoT, and big data.

Fiala and Tutoky (2017) investigate the quantity and impact of 1.9m papers in CS based on document type, language, discipline, country, institution, and publication source from 1945 to 2014 available in Web of Science. They investigate the distribution of document types, which shows that proceedings papers make up the biggest part of the collection, but articles have more than 7x the number of citations. The number of articles shows a steady rise over time, except for one large drop in 2007 which the authors attribute to papers published in two book series being classified differently from 2007 onward. Similarly, the amount of proceedings papers rises over time, except for a drop between 2010-2011, because multiple conferences are not indexed in those years. The distribution of document languages shows that 99% of all documents are in English. Considering all seven subject categories in Web of Science, “Artificial Intelligence” has the most papers and citations, while “Interdisciplinary Applications” has the most citations per paper. In the top 20 sources (i.e., venues) the “Lecture Notes in Computer Science” have the most papers, the “Journal of Computational Physics” has the most citations, and the “IEEE Transactions of Information Theory” has the most citations per paper. Fiala and Tutoky (2017) also compare the top 20 keywords for the entire time frame against those before 1995 and periods of five years after 1995 and find unique keywords in each period except 2005-2009. They also investigate the top 20 cited references (#1 being “INFORM CONTROL” from Zadeh, L.A.) and papers (#1 being “Fuzzy sets” from Zadeh, L.A.). The distribution of citations shows most citations are two years old, followed by three years, and one year, while 52.2% of papers remain uncited and less than 1% get over 100 citations.

The approach of our analysis in this thesis (Chapter 6) is inspired by that of Coşkun et al. (2019) and Fiala and Tutoky (2017), as we also conduct a broad study of CS research (i.e., looking into publications, authors, venues, and citations). While both conduct their analyses manually, we develop and use our system (CS-Insights), which can generate visualizations easily and intuitively. CS-Insights can replicate most of their analyses and show the top publications, authors, or venues, the distribution of document types and citations, and how they all change over time. Our topic modeling component can also determine the most salient terms for specific periods, venues, or authors. We also extend the studies of Coşkun et al. (2019) and Fiala and Tutoky (2017) by using a much larger dataset (CS-Insights has 5m publications and the updated version of D3 6m[2]), diving deeper into the areas they analyzed (e.g., distribution of citations and papers across authors and venues), and investigating more areas (e.g., the differences between CS and other fields of study; see the rest of this section). Future researchers can also use CS-Insights to verify and extend our research even further.

[1] https://www.vosviewer.com/
[2] https://zenodo.org/record/7069915

Other broad Studies

Some works allocate a part of their analysis to detail differences between institutions or countries over time or which are the most productive affiliations (Coşkun et al. 2019; Fiala and Tutoky 2017; Xia et al. 2021). The current version of our dataset does not include any data on the affiliations, so we leave the inclusion and investigation of institutions and countries to future work (Section 7.2). For this reason, we also leave out the many studies that focus entirely on analyzing the state and trends over time of CS research for specific countries (Uddin et al. 2015; Supriyadi 2022; Faiz 2020) or institutions in general (Zurita et al. 2020). The country-specific studies focus on the output and performance of publications, authors, and institutions. For example, Uddin et al. (2015) compare the performance stats (e.g. publications, citations) of Mexico against the world over time, investigate top countries, institutions, publication sources, and authors, the number of authors per paper, and collaboration patterns of authors, institutions, and countries. Zurita et al. (2020) only rank the institutions based on citations in seven sub-fields of CS. This shows their approaches mirror Coşkun et al. (2019) and Fiala and Tutoky (2017) or cannot be replicated with our data, so we do not cover their analyses and results any further. The authors of the studies on countries and institutions use different data sources, e.g., Scopus (Supriyadi 2022; Faiz 2020), or Web of Science (Uddin et al. 2015; Zurita et al. 2020). For evaluation Faiz (2020) uses SciVal[3], but most researchers use no tool for their evaluation or do not specify it.

Some studies also include analyses with networks on authors, citations, or terms (Coşkun et al. 2019; Uddin et al. 2015). While we look into some of these areas, we do not leverage any networks and thus cannot perform any analyses, which require networks or graphs. In the future, we intend to also conduct analyses with networks (Section 7.2).

Studies on Topics & Terms

Other studies focus more on emerging terms and which areas are researched currently. Tattershall et al. (2020) apply a stock-market-inspired burst detection algorithm to DBLP data (2.6m documents between 1988 and 2017) to find "bursty terms", i.e., the fastest-rising topics in the history of CS research. They find historic peaks for "Java", "e-commerce", and "Smartphone", and that "word embeddings" and "deep learning" are still rising. Terms like "neural network" and "virtual reality" have two peaks, while other terms like "novel" are linearly increasing. Most terms show a life cycle of popularity and their classifier can predict with an accuracy of 80%, whether a term will rise or fall in popularity. Xia et al. (2021) use a different approach and leverage data from Scopus (75m documents from 1996 onward) and its classification system of subject areas. They cluster the publications based on direct citation references and then evaluate the prominence of each topic using citations, views, and the impact of recent years. The authors investigate the top 20 frontiers in

[3] https://www.scival.com/

CS and find "Object Detection; CNN; IOU" to be the most prominent one, followed by "Bitcoin; Ethereum; Blockchain", while the most prominent frontier in NLP is "Sentiment Classification; Named Entity Recognition; Entailment", followed by "Sentiment Classification; Opinion Mining; Product Review". We do not analyze the evolution of single terms or the most prominent topics as a whole, but using CS-Insights's topic modeling component, we can determine the most salient terms and most prominent topics for venues, authors, and fields of study, and how they change over time.

Studies on Conferences vs. Journals

Another popular area is the comparison of conferences and journals in CS research (Franceschet 2010; Vrettas and Sanderson 2015), which is tied to the characteristics of CS research itself. Most researchers in CS focus their publications on conferences and not journals, unlike other research fields (e.g., medicine), where researchers use journals as the primary way to publish their findings (Vrettas and Sanderson 2015; Vardi 2009; Franceschet 2010). Vrettas and Sanderson (2015) argue this is the reason why many studies compare conferences and journals in CS, as CS is an outlier among the research fields in this regard.

Rahm and Thor (2005) analyze the citation frequencies between two conferences and three journals in the database field from DBLP over 10 years (1994-2003), with citation information from Google Scholar. They conclude, that the conferences have a higher citation impact than the journals. In a later publication, Rahm (2008) finds that conferences still have a higher impact than journals, again using select high-quality conferences and journals between 1996 and 2004. On the other hand, Franceschet (2010) finds that journals have a higher impact in CS than conferences. He uses data from DBLP, Google Scholar, and Web of Science to look into the top authors based on different measures (e.g., number of publications), which shows researchers in CS publish more in conferences than journals. His study on the most popular topics and nations with the highest scientific impact with separate entries for journals and conferences both show that journals receive significantly more citations and thus have a higher impact. Vrettas and Sanderson (2015) conclude the differences between the findings of Rahm and Thor (2005) and Franceschet (2010) are due to different data sources, as Rahm and Thor (2005) use DBLP and Franceschet (2010) uses Web of Science. While Franceschet (2010) also uses Google Scholar and DBLP data, his conclusion that journals have a higher impact is based solely on publication and citation information from Web of Science. Vrettas and Sanderson (2015) also examine conferences (195,513 papers) and journals (108,600 papers) themselves, by using data from Microsoft Academic Search[4] and aligning it with venue rankings from the Australian government's research assessment Excellence in Research for Australia (ERA). They find the difference between citations of journals and conferences in CS is marginal. Aligning the venues by the ERA ranking, the high-ranked conferences

[4] The service was retired in 2012.

get, on average, more citations than the high-ranked journals. Incidentally, this aligns with Rahm and Thor (2005), who also compare reputable conferences and journals.

In our research, we are also taking a venue-based approach like Rahm and Thor (2005) and Vrettas and Sanderson (2015), by looking at the BibTeX entries of publications, which determines whether the publication is from a journal (i.e., "article") or a conference (i.e., "inproceedings"). We extend their work by covering the number of citations and publications, topics, changes over time, top venues, and preferences of top authors. Other researchers use more author-based approaches to analyze publication patterns of authors and affiliations (Kim 2019; Kumari and Kumar 2020).

Studies on Comparisons between CS and other Research Fields

Vrettas and Sanderson (2015) show that there is a prevalence of conferences in CS, as 76% of ranked conferences in the ERA assessment across all fields are from CS. Michels and Fu (2014) also show other research fields prefer journals by listing the distributions of publications among journals and conferences for 27 research fields in Web of Science in 2009. Yet, Šubelj and Fiala (2017) show growth in journal publications in CS and attribute this to a rising number of new journals, rather than each journal publishing more. This is again different from other research fields, as the rise in journal articles in physics is due to the existing journals publishing more (Šubelj and Fiala 2017).

We also investigate the differences between CS and other research fields (e.g., engineering, medicine), which most other papers only briefly mention if they cover them at all. The focus of our analysis is the difference in preferences for journals and conferences regarding citations and publications, and their topics.

3.1.2 Scientometric Studies in Natural Language Processing

In NLP there are two series of publications on scientometric studies, that are worth noting, as they go into more detail, than Fiala and Tutoky (2017) and Coşkun et al. (2019).

Mohammad (2020b) uses the NLP Scholar dataset and visualization to examine the state of NLP research. He discovers an increase in papers and authors in the last two decades and that authors are also publishing more papers yearly. The number of workshop and conference papers in the dataset (and thus NLP) is also many times larger than that of journals. He also finds that the number of publications is higher in alternate years, due to biennial conferences. In another paper Mohammad (2020a) investigates the citations of NLP literature, where he finds that journals have the highest average and median citation count, even though they make up only 2.5% of the papers. Top-tier conferences (i.e., ACL, EMNLP, NAACL, COLING, EACL) combined ranked second, before the other conferences, workshops, etc. The same results are observed when only recent years are considered. In his work, he also bins the publications according to citation count and finds 6.4% do not have any citations and about 56% have 10 citations or more. Lastly, Mohammad (2019) investigates the topics of NLP research, by analyzing the top unigrams and bigrams that occur in the titles of the papers and their development over time, number of citations and papers, and average and median citations. He finds, that "language" is the most occurring unigram and "machine translation" the most occurring bigram. The diversity of title unigrams was lower in the 1980s compared to recent years and "neural" has been the most occurring unigram in titles per year since 2017.

Mariani et al. (2019a) perform an extensive analysis of the publications, authors and their collaboration, venues, citations and references, and their trends over time using the NLP4NLP corpus. The authors find the number of papers, authors, and references increased more in the last two decades than before 2000. They also list the most productive authors based on their amount of publications and show that publications are higher in alternate years. Their analysis shows that recent papers (2015) and old papers (1974 and before) get the least amount of citations on average. New papers have not had enough time to accumulate citations yet, and it also becomes apparent there are only very few publications in their data for the 1960s and 1970s, which could explain the low average citations. In their second paper using the NLP4NLP corpus Mariani et al. (2019b) investigate topics and terms over time and how their occurrences develop. Mariani et al. (2019b) find, e.g., that topics like "hidden markov models" and "speech recognition" dropped in frequency in the last few years, while "annotation" and "dataset" were rising. "wordnet" and "support vector machine" were rising for a while, but also dropped in frequency the last few years. They also show that single terms like "bigram" and "trigram" were also less frequent, while "ngram" saw a rise and then stagnated in the last few years.

Our case study on CS takes multiple aspects from the analysis conducted by Mohammad (2020b) and Mariani et al. (2019a) that are not present in the studies on CS, e.g., the number of authors over time, or citation binning. We use our automated system CS-Insights, similar to Mohammad (2020b), who uses his NLP Scholar visualization for his analysis. Mariani et al. (2019a) on the other hand use multiple different tools to process the data and generate the visualizations. Finally, the analysis we perform includes the trends of papers, authors, venues, topics, citations, and references in CS, but delves deeper into each analysis and adds aspects like discrepancies between conferences and journals, or research fields, which are missing or only shortly mentioned in the scientometric studies on NLP mentioned in this subsection.

3.2 Resources

For this thesis we use “aggregators” as a broad term to refer to data resources such as digital libraries, repositories, and search engines, that aggregate scientific publications, and index them and their metadata, but might also include additional features to work with the data beyond a search function. These features can include, e.g., varying degrees of visualization options or filters to refine the search. This section covers broad aggregators, that comprise publications of multiple disciplines (Section 3.2.1) and aggregators that are specialized in specific disciplines or fields (Section 3.2.2). We also show tools that help to conduct scientometric studies in general (Section 3.2.3), and which resources are available in NLP (Section 3.2.4), as some resources are similar to what we try to achieve with CS-Insights.

3.2.1 Broad Aggregators

Google Scholar (estimated to have over 389m records (Gusenbauer 2019)) and Semantic Scholar (over 206m papers[5]) are freely accessible web-based search engines for scholarly literature. They include different records, such as peer-reviewed publications and pre-prints. The search engines focus on searching and finding publications, authors, and their metrics (e.g., h-index, number of papers, citations), but lack details on venues and publishers. Their filter options are also limited, as both do not have a filter for the access type or number of citations, and Google Scholar also cannot filter by authors or venues. Additionally, neither Google Scholar nor Semantic Scholar offers an interactive platform to browse their databases, preventing users from exploring features not explicitly available on their website (e.g., the fields of study). While their web interfaces are freely available, Google Scholar does not provide any API or means to download their data. Some studies from Section 3.1 use Google Scholar for their analysis, but only to a very limited degree, e.g. to find the h-index of specific authors (Franceschet 2010), find citations for papers of a handful of venues (Rahm and Thor 2005), or get citations for a smaller sub-field in CS (Mohammad 2020b). Semantic Scholar offers their data through an API or as a bulk download, but both options to access their data require an access key[6]. Their API also allows 100 requests per 5 minutes without a key, for testing purposes. Two datasets are offered by Semantic Scholar: S2AG (Semantic Scholar Academic Graph) (Ammar et al. 2018), which includes all data that makes up the knowledge graph that powers Semantic Scholar (over 206m papers); and S2ORC (Semantic Scholar Open Research Corpus) (Lo et al. 2020), which includes a subset of open-access papers from S2AG and their metadata (136m papers), enriched with abstracts and full-texts. Google Scholar and Semantic Scholar are both limited by not offering any quantitative analysis options and not providing easy access to their data, both of which CS-Insights overcomes. While Semantic Scholar’s S2ORC dataset allows easy access, it is not regularly updated,

[5] https://www.semanticscholar.org/
[6] https://www.semanticscholar.org/product/api

which limits its use to investigate more current trends. Other researchers leverage Google Scholar, but they only use it for its h-index or citation metrics and in much smaller quantities than we require to analyze CS.

Some researchers (Vrettas and Sanderson 2015; Bornmann et al. 2021) use Microsoft's academic search engines in their scientometric studies, but Microsoft retired their services Microsoft Academic Search in 2012[7] and its successor Microsoft Academic in 2021[8], so both will not be covered further.

Two large web-based paid-access platforms are Web of Science (over 171m records[9]) and Scopus (over 87m records[10]). They expand on the capabilities of Google Scholar and Semantic Scholar (e.g., the search engine and citation index), by each offering more filters (19 in Web of Science and more than 12 in Scopus) to refine the search (e.g., publisher, field of study, affiliation, keyword). The (refined) search results can then be exported and downloaded for further analysis, which other resources (e.g., Google Scholar, Semantic Scholar) do not offer. For that reason, many scientometric studies use the data from Web of Science (Coşkun et al. 2019; Fiala and Tutoky 2017; Franceschet 2010) or Scopus (Xia et al. 2021; Bornmann et al. 2021). Especially researchers analyzing the research output of affiliations (specific countries or institutions) tend to choose Web of Science (Uddin et al. 2015; Zurita et al. 2020) or Scopus (Faiz 2020; Supriyadi 2022), as other resources might not have data on affiliations (e.g., DBLP). Both platforms also allow a basic analysis of the (refined) search results by generating visualizations. Web of Science can group the results by 21 attributes (similar attributes as for the filters, e.g., authors, publication years, document types), and then visualize the results as a treemap, bar chart, and grid with a configurable number of entries. Scopus can also group its (refined) search results and visualizes the results either as a line chart (for grouping by year or source), bar chart (author, affiliation, or country/territory), or pie-/ringchart (document type, subject area) while showing a list with the top entities on the left. These visualizations allow analyzing distributions, trends, and comparing entities (e.g., authors, venues), but are not used by any studies to our knowledge. Web of Science and Scopus provide many features to their users, but their main limitation is they are paid-access only, which prohibits researchers without the necessary funds (e.g., from developing countries) from accessing the services. CS-Insights intends to take their analysis component and make it available to everyone, extending the kind of quantitative analysis that can be done with Web of Science and Scopus. We also conduct a case study with CS-Insights using CS to show what it is capable of and that researchers do not have to conduct all their scientometric analysis manually, as most currently do. However, we do not offer any ways to search for papers, as there are already multiple open-access solutions for that (e.g., Google Scholar, Semantic Scholar, DBLP).

[7] https://web.archive.org/web/20170105184616/https://academic.microsoft.com/FAQ
[8] https://www.microsoft.com/en-us/research/project/academic/
[9] https://clarivate.com/webofsciencegroup/solutions/web-of-science/
[10] https://blog.scopus.com/posts/scopus-roadmap-whats-new-in-2022

3.2.2 Specialized Aggregators

The specialized aggregators still have varying degrees of specialization and size. We start detailing the larger and broader ones and then move to the smaller and even more specialized ones.

Two large open-access repositories are arXiv (over 2.1m scholarly articles[11]) and DBLP (Ley 1997) (over 6.3m publications[12]). arXiv stores pre-prints from sciences and some related fields so its contents are not peer-reviewed but it offers multiple ways to download its latest data. The Computing and Research Repository (CoRR) (Halpern 2000) is the section of arXiv that focuses on CS and has multiple categories (e.g., Artificial Intelligence, Computation and Language, and Databases). DBLP, on the other hand, entirely focuses on CS publications, including both peer-reviewed publications and some pre-prints, and their downloadable data gets updated monthly. Both arXiv and DBLP do not offer a citation index or options for analyses with visualizations. CiteSeerX[13] (over 10m records[13]) is a digital library, which also focuses on papers in Computer and Information Science. It crawls its data from publicly available websites and thus is fully open-access and provides all its data for download. Their copyright only covers up to 2019 and an exemplified search for "machine learning" only returns papers from 2017 or earlier, so we conclude CiteSeerX is not further updated. No studies we cover use data from CiteSeerX, but multiple studies use DBLP (Rahm and Thor 2005; Franceschet 2010; Tattershall et al. 2020; Kim 2019) and arXiv (Sharma et al. 2021), due to their open-access nature and being up to date. More studies use DBLP than arXiv which we explain with arXiv consisting solely of pre-prints, while DBLP covers mostly peer-reviewed publications. As DBLP, arXiv, and CiteSeerX overcome the paid-access issues of Web of Science and Scopus, and include an easy data download, unlike Google Scholar and Semantic Scholar, they are a step in the right direction. Unfortunately, they do not provide any visual analysis, but this is the gap CS-Insights fills, as we leverage data from DBLP for our case study (Section 4.1.1) and thus expand the features of DBLP, to provide a free system to perform a scientometric analysis of CS. To our knowledge, no one else has created an open-source and open-access (visual) analysis system for CS yet.

Some publishers like IEEE and ACM also have their own platforms, i.e., IEEE Xplore[14] (over 5.7m items[14]) and the ACM Digital Library[15] (over 550k articles[16]), for their own publications and those of their partner publishers. Both platforms offer a search, citation index, and some filters. As both publishers focus on CS publications, and in the case of IEEE also engineering, the contents of their platforms reflect that. Downloading some of the articles and papers with full-text requires paid-access and they provide little to no options to analyze the search results. No study we

[11] https://arxiv.org/
[12] https://dblp.org/
[13] https://citeseerx.ist.psu.edu/
[14] https://ieeexplore.ieee.org/
[15] https://dl.acm.org/
[16] https://libraries.acm.org/digital-library

looked into uses data from IEEE Xplore or the ACM Digital Library, which is not surprising, considering Web of Science and Scopus offer better features to get data for a scientometric study in general, and repositories like DBLP are fully open-access with a larger number of CS publications.

For completion, we also want to mention some of the many small aggregators, that focus on specific areas in CS or offer additional features, e.g., linking code or tables. Zeta Alpha[17] is a discovery and recommendations engine for papers, trends, and code in AI and data science. Papers With Code[18] is a free and open resource of machine learning papers, code, datasets, methods, and evaluation tables. NLP Index[19] focuses on NLP GitHub repositories with papers. 42Papers[20] aggregates high-quality CS and Artificial Intelligence (AI) papers and enabled its users to share them with each other. None of the four mentioned aggregators offer any citation counts or analysis, except Zeta Alpha, which maps the search results into a two-dimensional semantic space and links them in a graph using VOSViewer. Like IEEE Xplore and the ACM Digital Library, these four small aggregators were also not used in any studies we covered in Section 3.1, as these are more for niches in CS and not CS as a whole. They do not provide any features that compare to CS-Insights and their data is also not of interest to us, as DBLP provides data more suited for our case study.

There are also other aggregators for other areas, e.g., PubMed[21] for medicine, but those also do not provide features similar to CS-Insights (i.e., analysis based on visualizations), so these will not be covered, as our case study is on CS.

3.2.3 General Tools

Besides aggregators, there are also some tools not specific to any research field, that researchers can use to perform scientometric studies. These tools do not focus on a search function, instead, most focus on visualizations and analyses. In this subsection, we present a few such tools, their features, and major differences from CS-Insights.

SciVal builds on Scopus's data to visualize research performance for authors, institutions, and countries. It allows researchers to benchmark author and institution performance and analyze research trends based on different metrics, including publication and citation metrics from Scopus and additional metrics (e.g., topics, authors, and research areas). Both Scopus and SciVal belong to Elsevier, and thus SciVal is paid-access only like Scopus. The main difference between SciVal and CS-Insights is, that SciVal focuses on determining research performance, while CS-Insights allows for a scientometric analysis of broad trends in CS research (e.g., of its publications, authors, and venues).

Some tools use network-based approaches for analysis, which we present two of in

[17] https://search.zeta-alpha.com/
[18] https://paperswithcode.com/)
[19] https://index.quantumstat.com/
[20] https://42papers.com/
[21] https://pubmed.ncbi.nlm.nih.gov/

this paragraph. CiteSpace[22] (Chen 2004) is a free Java application, that visualizes trends and patterns in scientific literature. It visualizes the co-citation network of a knowledge domain to make it easy to locate pivoting points, turning points, and cluster centers. Chen (2004) uses his tool to find the two revolutions in the superstring field in theoretical physics. He later added features to visualize emerging trends and abrupt changes and uses them to show their effectiveness in mass-extinction research (1981-2003) and terrorism research (1990-2003) (Chen 2006). CiteSpace can directly work with data from Web of Science and includes interfaces to work with data from PubMed, arXiv, ADS, and NSF Award Abstracts. Similarly, VOSViewer (van Eck and Waltman 2010) visualizes bibliometric networks. It can leverage relations of citations, bibliographic coupling, co-citations, or co-authorships. Additionally, it offers text mining capabilities to extract important terms from scientific publications and to visualize them in a co-occurrence network. Zeta Alpha (mentioned in Section 3.2.2) uses VOSViewer to create their graphs linking publications in a semantic space and Coşkun et al. (2019) use it for some of their graph-based analyses. Both tools presented in this paragraph focus on analysis with networks, which CS-Insights and our case study on CS do not cover. CiteSpace and VOSViewer provide useful insights into research areas or domains that do not have specialized tools (e.g., the tools NLP has; see next subsection) available, but they do not help us with a quantitative analysis as CS-Insights does.

3.2.4 Resources in Natural Language Processing

In this subsection, we detail resources available for the research field of NLP, as some resources strongly correlate to what we want to achieve with CS-Insights.

The most prominent resource in NLP probably is the ACL Anthology[23], which consists of nearly 80k open-access papers from the area of computational linguistics and NLP. It is used in many other resources and studies, e.g., the NLP4NLP corpus (Mariani et al. 2019a), which includes papers from the ACL Anthology and some other venues with a focus on NLP (e.g., ISCA, IEEE, ICASSP, TASLP, LRE); and the NLP Scholar dataset (Mohammad 2020b), which combines the data from the ACL Anthology with citation information from Google Scholar. Our analysis also shows over 99.3% of the papers in the ACL Anthology are also in DBLP (Wahle et al. 2022), which is the dataset we use for our study. We consider the ACL Anthology for NLP, what DBLP is for CS: the largest available open-access dataset for the respective area.

Researchers interested in investigating trends in NLP can use the interactive visualization NLP Scholar[24] (Mohammad 2020c), which is built with Tableau[25] and uses the dataset with the same name (Mohammad 2020b). The NLP Scholar visualization features a bar chart for papers per year and citations per year, a list with the most cited papers and authors, a boxplot of citations, and a treemap of the venues with

[22] http://cluster.cis.drexel.edu/~cchen/citespace/
[23] https://aclanthology.org/
[24] http://saifmohammad.com/WebPages/nlpscholar-demo-basic.html
[25] https://www.tableau.com/

the most published papers. It offers filters for the year of publication, authors, the number of citations, and paper title unigram or bigram. CS-Insights shares certain similarities with NLP Scholar, as it also allows its users to perform a quantitative analysis of its underlying data, which is also our goal, just with another focus on the data. Thus, we expand on the capabilities of NLP Scholar and add more filters and aggregation options, while also covering a broader field by investigating CS. Due to us building a dedicated system and not relying on Tableau, we also have a more scalable solution, that allows processing larger datasets.

In NLPExlorer[26] (Parmar et al. 2020) users can explore NLP papers, venues, authors, and topics with an LDA (Blei et al. 2003) topic modeling approach. The tool curates five topics, each with multiple subcategories. These topics and subcategories can be explored by searching and then selecting a specific venue or author. NLPExlorer also shows the paper and citation distribution over years, but selecting a paper or topic only shows its metadata. Another tool, called DRIFT[27] (Sharma et al. 2021), tracks research trends and developments over the years. The available analysis methods include keyword extraction, word clouds, predicting trends using productivity, tracking bi-grams, finding the semantic drift of words, tracking trends using similarity, and topic modeling. Sharma et al. (2021) perform a case study on the cs.CL corpus[28] from arXiv, which is the subset of CS papers that covers Computation and Language (i.e., NLP), but users can also upload their own corpus. CS-Insights also includes a topic modeling component, but with a different focus than NLPExlorer, as we generate the topics automatically from the terms used in the titles and abstracts and additionally offer a comparison based on the most used terms. Our implementation and visualization also allow for more customizability and exploration than DRIFT.

[26] http://nlpexplorer.org
[27] https://gchhablani-drift-app-t0asgh.streamlitapp.com/
[28] https://arxiv.org/list/cs.CL/current

4 Methodology

The goal of this thesis is to analyze the state of CS research with a scientometric study. In the previous chapter (Chapter 3), we showed that current solutions to analyze the state of CS research are limited, so we build our own (CS-Insights) to answer our research questions and achieve our goal. This chapter details how we build the CS-Insights system to analyze CS research, which takes three steps. We first require a large dataset specialized on CS publications and their metadata, so we reason the data source we pick (Section 4.1.1) and explain how we acquire the data (Sections 4.1.2 to 4.1.3). Second, we need to store the data in a way that makes it easy to extract the information we need again for our study (Section 4.2.1) and then provide ways to manage and interact with the data more efficiently (Section 4.2.2), which also makes it available for other researchers. Lastly, we create an interactive system that queries the data user-friendly and visualizes it in different plots. We explain details about its design (Section 4.3.1) and interface (Section 4.3.2) and show some examples of how the system can be used (Section 4.3.3).

4.1 Data Acquisition

The first step is to extract a large collection of data which will allow us to answer our research questions. For this, we must first decide on a source for our data (Section 4.1.1). We then detail how we extract the data we need and enrich it with more metadata (Sections 4.1.2 to 4.1.3). This section describes how we create the original version of the DBLP Discovery Dataset (D3) (Wahle et al. 2022), which we use in this thesis and is available on zenodo[1]. A new version is also available[2], but that is for future work (Section 7.2) as it was not available in time for this thesis.

4.1.1 Data Source

We decide to use a preexisting data source (DBLP) over building a new one from scratch as this has multiple benefits, including the source most likely already taking care of issues in matching papers, authors, venues, etc. Publications without a Digital Object Identifier (DOI) or link might be hard to join because of small differences in the title (e.g., due to hyphens), or there might be multiple versions due to pre-prints. The authors might have different spellings (e.g., “Christopher D. Manning”, “Christopher Manning”, “Chris Manning”), or there might be multiple authors with the same name (e.g., “Yang Liu”[3]) (Ley 2009; Ammar et al. 2018). Venues also might be abbreviated (e.g., “IEEE Transactions on Information Theory” to “IEEE Trans. Inf. Theory”), or the venues are just mentioned with their code (e.g., “EMNLP” instead of “Conference

[1] https://zenodo.org/record/6477785

[2] https://zenodo.org/record/7069915

[3] Search on Google Scholar for “Yang Liu”: https://scholar.google.com/citations?view_op=search_authors&mauthors=yang+liu

on Empirical Methods in Natural Language Processing”). The codes can also change, e.g., all codes in the ACL Anthology changed in 2020 (Mohammad 2020b).

There are multiple reasons for picking DBLP as a data source over the other available aggregators covered in Section 3.2. All broad aggregators we mentioned in Section 3.2.1 could not be utilized. Google Scholar does not offer easy access to its data (no standardized API and rate limitations for webpage crawling), while Web of Science and Scopus are not open-access. Semantic Scholar’s contents are also proprietary (Gusenbauer 2019). They do offer the S2ORC dataset (Lo et al. 2020), which consists of a subset of their data, but it is not regularly updated and received its last update in 2020[4]. Of all the specialized aggregators we covered (Section 3.2.2), we find DBLP the most fitting. It is the largest open-access repository of CS publications and their metadata, which also gets updated monthly (Wahle et al. 2022). DBLP also takes care of most of the mapping issues mentioned in the last paragraph (Ley 2002; Ley 2009). Other specialized aggregators all have one or more drawbacks compared to DBLP, because they are smaller (Zeta Alpha, Papers With Code, NLP Index, 42Papers), not peer-reviewed (arXiv), or not updated anymore (CiteSeerX). IEEE Xplore and the ACM Digital Library are tied to their respective publishers and thus might be focused too much on specific areas and not give enough variety for document types or venues. Additionally, some previous scientometric studies already used DBLP data (Rahm and Thor 2005; Franceschet 2010; Tattershall et al. 2020; Kim 2019), so we feel DBLP is a good choice for our data source.

4.1.2 Primary Information from DBLP

DBLP offers open-access to their data in multiple ways. Researchers can use the search, that is available through their website, or the search API for publications[5], authors[6], and venues[7]. DBLP also provides monthly updated XML dumps of their data[8]. We retrieve the full release of all currently available data, as we are interested in the state of CS at a large scale over time, and extract all records from January 1st, 1936 to December 2nd, 2021, which includes DBLP’s monthly release from December 1st, 2021. In the future, we can also use DBLP’s monthly releases to keep CS-Insights up-to-date automatically, i.e., download the latest release each month to add all new entries and update already existing entries (Wahle et al. 2022). An overview of the attributes, including examples of the data we retrieve from DBLP, will be provided in Table 4.1 in Section 4.2.

The largest actors in research are the publications, authors, and venues. DBLP directly supplies them, so we can extract them using a limited amount of additional work:

[4] `https://github.com/allenai/s2orc`
[5] `https://dblp.org/search/publ/api`
[6] `https://dblp.org/search/author/api`
[7] `https://dblp.org/search/venue/api`
[8] `https://dblp.org/xml/release`

Publications Most entries in DBLP are indexed publications with their respective metadata; other examples include webpages and author information. DBLP classifies documents according to their BibTeX entry types (e.g., article, inproceedings). We transform all records into a standard JSON format based on the document type of the publications and map authors and venues to uniquely identified entities.

Authors DBLP handles multiple authors with the same name using an iterative four-digit counter in their data and when aggregating the data it distinguishes those authors automatically using different heuristics (Ley 2009). In case authors cannot be clearly distinguished, DBLP uses disambiguation pages[9]. Authors with multiple names are mapped in DBLP's author records (Ley 2009), which are sparse and rarely contain other informative features besides an URL to the personal webpage of the authors. We use the unique ids of the authors to map them to their publications. The author's current affiliated institution is not available in its own field and might only be entered in the "note" field (Ley 2009).

Venues For almost all publications, DBLP provides a venue code, by using the abbreviation of the venue or its acronym (e.g., "IEEE Trans. Inf. Theory" instead of "IEEE Transactions on Information Theory" or "EMNLP" instead of "Conference on Empirical Methods in Natural Language Processing"). We map them to their publications with their unique ids, like the authors. In DBLP's data, the venue name is stored in different fields, depending on the document type, i.e., conferences use the field "booktitle", while journals use the field "journal". These are also two different fields in our dataset (D3), so when extracting information from the dataset, one has to be careful to consider both fields and merge their contents, as both fields are never used in the same record.

Other Fields DBLP also contains some other fields, which we copy without any modification and directly store in the publication entries. The two most notable fields are the *type of paper* (contains information about the BibTeX type of the publications) and the *publishers* (e.g., Springer, IEEE, and ACM; but the data is very scarce, as less than 10% of publications have publishers annotated).

4.1.3 Secondary Information from Full-Texts

The full-texts of publications contain valuable information about author affiliations, content, and references currently not present in DBLP or other resources (e.g., NLP Scholar). We leverage the different fields DBLP provides for DOIs and links to the publications (e.g., "url" or "ee", meaning electronic edition (Ley 2009)), to crawl the publications and retrieve their corresponding PDF files, which include the full-text. We then use GROBID[10] (Lopez 2022) to parse the PDFs and extract abstracts, affiliations, and references. GROBID stands for **G**ene**R**ation **O**f **BI**bliographic **D**ata and is an

[9] Disambiguation page for "Yang Liu": `https://dblp.org/pid/51/3710.html`
[10] `https://github.com/kermitt2/grobid`

open-source machine-learning library for extracting, parsing, and converting PDF documents into structured XML documents. Table 4.1 in Section 4.2 also includes the attributes we extracted from the full-texts with example values.

Abstracts For abstract extraction, we use GROBID's CRF Wapiti (Lavergne et al. 2010) engine, which achieves an F1-score (using Levenshtein Matching with a minimum distance of 0.8) of 92.85% when drawing 1943 PubMed papers[11]. With this model, we retrieve 3,980,144 abstracts which are 81.33% of the documents in the dataset. GROBID disregards the remaining documents because of poor quality or because there is no accessible document that could be parsed. We directly add the extracted information to the records we get from DBLP.

Affiliations We extract the author names and affiliations with the same engine we use for extracting the abstracts. To create author–affiliation pairs, we match author names from extracted affiliations to author names in DBLP using the Levenshtein distance. Using name matching to create author–affiliation pairs is also robust in practice, which we demonstrate by performing two small bootstrap and permutation tests (Dror et al. 2018). In the first test, we randomly draw 20 samples of $n = 100$ publications and evaluate how often author names extracted from the PDFs do not match those in DBLP. To draw more challenging samples in the second test, we took the first $n = 100$ publications from a ranked list in which the average Levenshtein distance between authors' names increased. Both tests show less than 5% of names are mismatched ($p < 0.001$).

While our approach to creating authors–affiliations pairs proves to yield great quality results, GROBID has issues properly extracting and parsing affiliations in the first place. There are duplicates, incorrect, incorrectly structured, or missing affiliations. Considering the issues GROBID has to parse the affiliations, we decide the incorporation of institutions will be left to future versions of CS-Insights (Section 7.2). The information about the countries is also left for future versions, as DBLP does not have that data and we cannot derive the country information from the institutions, due to the issue listed in extracting them.

Citations Google Scholar does not provide large-scale access to their data, i.e., it does not have a standardized API and limits access for crawling, so we cannot use Google Scholar to retrieve citations. Other services also cannot be used, as they have the same issues (e.g., Semantic Scholar), or their data is not open-access (e.g., Web of Science and Scopus). We instead calculate citations within DBLP ourselves, by building a citation graph from the bibliographies of full-texts similar to the ACL Anthology Reference Corpus (Radev et al. 2009). To parse the documents' bibliographies, we use GROBID's BidLSTM-CRF engine, which obtains an F1-score of 87.73% for the PubMed samples (using Levenshtein Matching with a minimum distance of 0.8)[11]. We add two fields to each publication entity to create our citation links. One for the incoming citations (i.e., for each document that cites the publication) and one for

[11] `https://grobid.readthedocs.io/en/latest/Benchmarking-pmc/`

the outgoing citations (i.e., for each reference in the bibliography of the publication). From this, we receive two lists of document ids, which we can use to construct a citation graph.

When measuring the number of citations that come from outside D3 using the Semantic Scholar API we receive the result that 21.15% of citations are from papers outside of D3 (i.e., other research fields than CS) (Wahle et al. 2022). During that step, we match our data to Semantic Scholar using the DBLP-id, which also yields us the entries for the fields of study.

4.2 Data Storage & API

In the previous section, we acquired the data we need, so the next step is to store it in a way, that makes it easy and efficient to retrieve again. To store our data we decide to use MongoDB, which allows great performance and scalability (Bradshaw et al. 2019, p. 6; Győrödi et al. 2015). One of our future goals is to make CS-Insights available to work with other datasets, which requires flexibility of the schemas and is one of the benefits of using a non-relational database (Bradshaw et al. 2019, p. 3; Győrödi et al. 2015). In this section, we first explain how we design the database schema (Section 4.2.1) and how we manage the data and make it readily available through an API (Section 4.2.2).

4.2.1 Database Schema

The current database schema is shown in Table 4.1. It differs from the original schema shown by Wahle et al. (2022) (Table A.1), as we had to make some changes for performance increases and due to some attributes we could not use for any analyses.

We put the data from the crawler into our database with a Python script. The script ignores any attributes in the data that we are not interested in (e.g., pages). Some attributes are also unused (i.e., empty fields), as we do not intend to analyze them currently, but maybe in the future (Section 7.2). Those attributes are marked with an asterisk (*) in Table 4.1. Citation references (i.e., ids for incoming and outgoing citations) are empty, as those are taking up most of the space, and we are not interested in any network analysis. So far, we only use the citation references to extract the citation counts. We also leave the affiliations empty, as explained in Section 4.1.3, and many fields for authors or venues, as we do not have that data yet, but might want to investigate them further in the future.

Another change we make is the duplication and denormalization of the author and venue names, so they are also directly available in the publications collection. When first testing the system, we quickly realized, that a normalized solution that requires lookups of author and venue names using the unique ids does not work, as it is too slow. A denormalized schema helps, as it is quicker to read, but takes longer to write (Bradshaw et al. 2019, pp. 211–212). In the normalized schema, we would have to do a `$lookup` operation across the entire dataset, which takes 10-15 minutes from our

Attribute	Example
publication	
id	62cc663aeba63d1b526e0689
title	NLP Scholar - An Interactive ...
abstractText	As part of the NLP Scholar ...
yearPublished	2020
authors	[Saif M. Mohammad]
authorIds	[62bf2884e9832d137d41fb5e]
venue	ACL (demo)
venueId	62bf273022ce6513861ee199
publisher	ACL
typeOfPaper	inproceedings
fieldsOfStudy	[Computer Science]
*inCitations	[]
inCitationsCount	9
*outCitations	[]
outCitationsCount	34
openAccess	true
dblpId	conf/acl/Mohammad20b
doi	https://doi.org/...
pdfUrls	[]
url	db/conf/acl/acl2020-d.html#Mohammad20b
author	
id	62bf2884e9832d137d41fb5e
fullname	Saif M. Mohammad
number	0001
orcid	0000-0003-2716-7516
*timestamp	-
*email	-
*dblpId	-
venue	
id	62bf273022ce6513861ee199
names	[ACL (demo)]
*acronyms	[]
*venueCodes	[]
*venueDetails	[]
*dblpId	-
***affiliation**	
*id	4eb3...f094
*name	National Research Council Canada
*country	Canada
*city	Ottawa
*lat	-
*lng	-
*dblpId	-

Table 4.1 Database schema currently used in CS-Insights. Unused attributes are marked with an asterisk (*).

experience. Instead, we denormalize the author and venue names and copy them into the publication collection, so the queries only take 10-15 seconds. CS-Insights revolves around analyzing (i.e., reading data) and only rarely writes data (once a month max.), so increased write times are not an issue.

Due to the limitations of the database, we do not perform any further denormalization, even though it could reduce response times even more (i.e., copying the data of all publications into the referenced[12] entries in the author and venue collection). MongoDB has a document size limit of 16MB (Bradshaw et al. 2019, p. 207), which means each publication, venue, and author can only have 16MB of information. The largest venue in our dataset ("IEEE Access") has around 55,000 publications, which means all of its 55,000 publications would need to be saved in the same document with further denormalization. 16MB is not enough for this, as this would leave less than 300 bytes for each publication including its abstract. Implementing this approach would be possible, but would need a lot more schema engineering and the response times at the time were also satisfying, so we saw no need for drastic changes in the schema.

4.2.2 API: Data Management and Usage

With our schema ready, the next step is to be able to put the data into the database and efficiently retrieve it again. For this, we create the backend of the CS-Insights system, which serves as a REST API with endpoints, which we can query to manage the data in the database and get the results back. Each endpoint serves one function, e.g., read from a collection, update a document in the collection, or aggregate results for analysis.

Data Management

To properly manage our data in the database using the REST API, the API needs to enable the basic CRUD operations to create, read, update, and delete documents in our database. We leverage the library express-restify-mongoose (ERM)[13] to automatically generate endpoints for the CRUD operations of all collections (i.e., papers, authors, venues, and affiliations) using our already defined schemas. The library also allows customization of the queries to a limited degree, by supporting sorting, skipping, and limiting returned documents, populating documents with documents from other collections using ids, selecting specific attributes, and having some filter capabilities. Using a library saves us a lot of work, time, code, and maintenance.

[12] Referenced through the author and venue ids.
[13] `https://florianholzapfel.github.io/express-restify-mongoose/`

User Management

Access to CS-Insights requires a user account, which everyone can register without cost, so we can mitigate misuse and better manage our limited server resources. To manage the user accounts we add two routes, one to register a new account and one to log in with an existing account. All endpoints except those for login and register require authentication with an account, either a normal user account or an administrator account. User accounts enable access to the endpoints that aggregate results and are used by the frontend to conduct analyses (We discuss these endpoints in the next paragraph). All endpoints for data management discussed in the previous section are only available to administrators, so users cannot modify our data or retrieve parts of the data they are not supposed to retrieve (i.e., abstracts we are not allowed to distribute further due to copyright laws).

Aggregations for Visualizations

Lastly, we have endpoints that perform aggregations for our visualizations, so the endpoints return exactly what is needed for the visualizations (Section 4.3). These aggregations are performed by directly querying MongoDB. We do not use ERM for this as there were multiple downsides to this approach:

- ERM does not have the option to aggregate results, i.e., there is no equivalent of the `$group` stage, which we need for most of our queries.
- It makes it easier to test the aggregation endpoints and their complex functionality, which make up the largest part of the functionality of the backend.
- There are some small issues, e.g., filters not properly working on populated documents.

Overall ERM works well for simple tasks in data management but fails for the complex aggregations we require for our visualizations. Additionally, we cache the results of any aggregation queries to make repeated queries faster.

4.3 Interactive Visualization(s)

Humans can better understand data if it gets visualized (e.g., in bar charts, line graphs, or scatterplots), than if it is just presented as numbers and text (Shneiderman et al. 2018, p. 552). We build an interface, which creates interactive visualizations to display our data intuitively, and is integrated into the frontend of our CS-Insights system. The first subsection goes over the design decisions of the prototype (Section 4.3.1), and the second subsection covers the interface of the CS-Insights system (Section 4.3.2), which is also part of our submission to arXiv/EACL (Ruas et al. 2022). Lastly, we showcase how the interface of the frontend can be used with some examples (Section 4.3.3).

4.3.1 Prototype Design

First, we create a prototype to decide the layout of the interface, its features, and which visualizations we want to include because directly implementing the frontend without a plan would take more time in the end. In this subsection, we only cover those basic decisions as many parts of the prototype and finished frontend are identical, and we already explain the finished product in more detail in Section 4.3.2. An example of the prototype can be seen in Figure A.1.

We use Figma[14] to design our prototype and go through four iterations before deciding on the final prototype. All four prototypes we create can be navigated, and change their visualizations based on what page is currently selected. During the development of the prototypes, we decide our goal is to provide researchers with a system to investigate CS research themselves and come up with their own questions they might want to answer. We want other researchers to explore what *they* want and not what *we* want, i.e., we do not want to answer our questions from the LREC paper again and simply reconstruct its plots (Wahle et al. 2022). As a result, we offer multiple dashboards with various visualizations and filter options to give a broad overview of all aspects of CS, as shown in the next subsection. The selection of plots is inspired by NLP Scholar, as it has proven successful at giving insights into NLP and providing a broad overview of the trends of the publications, authors, venues, and citations in NLP.

4.3.2 User Interface

CS-Insights offers web-based interactive visualizations to explore CS publications through their metadata, such as venues, authors, and abstracts. Figure 4.1 shows an example of the frontend's interface, which we reference throughout this section. The interface is composed of three main parts: A. *Dashboards*, B. *Filters*, and C. *Visualizations*. *Dashboards* control which *visualizations* are shown and which attribute of the publication metadata is currently visualized. *Filters* allow users to select which publications are visualized, by defining criteria the publication metadata has to match. For a better understanding of the interface, one can generally make the following analogy: CS-Insights's interface follows a similar structure as a SQL query, in which a dashboard (A) acts as `GROUP` statement and the filters (B) as `WHERE` clause.

A demo for CS-Insights is also available online[15]. To generate the visualizations, the frontend queries the aggregation endpoints in the backend, which requires authentication via a user account, as mentioned in Section 4.2.2. CS-Insights is still publicly available, as the account can be created through the interface without cost. At the time of writing this thesis, a demo account is available on the main GitHub page[16], which is also linked on the homepage of the UI[15].

[14] https://www.figma.com/
[15] https://cs-insights.uni-goettingen.de/
[16] https://github.com/gipplab/cs-insights-main

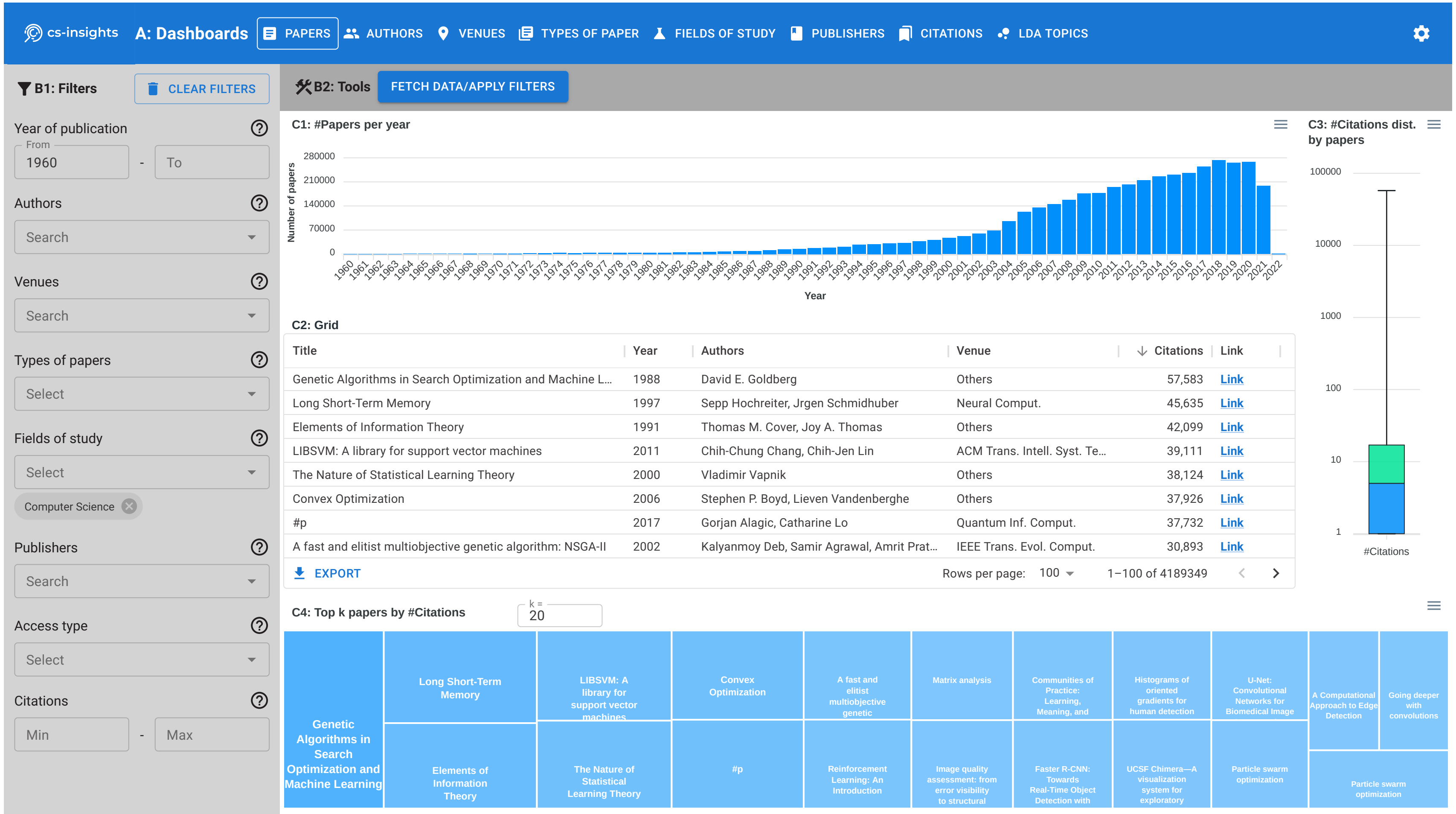


Figure 4.1 CS-Insights's UI with the *Papers* dashboard selected. A. Dashboards, B1. Filters, B2. Tools, C1. #Papers per year, C2. Grid, C3. #Citations distribution, and C4. Top k papers by #Citations.

Dashboards and App Bar

The *dashboards* (A) can be selected from the app bar at the top of the page, which is responsible for the navigation. Besides the dashboards, the app bar also shows the CS-Insights logo on the left, which brings the user back to the homepage if clicked on, and a gear on the right to log out again or view the *Account* page. There are currently eight different dashboards that users can select from the app bar. First is the *Papers* dashboard, which shows information about the publications, as seen in Figure 4.1. The dashboards for *Authors* (full name), *Venues* (where a paper was published, e.g., ACL, Communications of ACM), *Types of Paper* (according to their BibTeX entries), *Fields of Study* (high-level areas of research, e.g., CS, mathematics), and *Publishers* (responsible agency/institution for publishing, but >90% of publications leave this field blank in DBLP) share the same visualizations with the *Papers* dashboard (Figures A.2 to A.6). However, they aggregate the publications by the respective attribute in the publication metadata, i.e., on the *Venues* dashboard C1 would show the number of venues over time, C2 and C4 the top venues, and C3 the distribution of total citations venues have received. These five dashboards include a "metric switch" in the *Toolbar* (B2), which switches the metric from #Citations to #Papers. The *Citations* dashboard consists of a bar chart (C1) and boxplot (C3) each for both the incoming citations (i.e., when a paper gets cited by another publication) and the outgoing citation (i.e., the references in the bibliography of a paper) (Figure A.7). Lastly, the *LDA Topics* dashboard performs a topic modeling analysis based on LDA (Blei et al. 2003) with the titles and abstracts of the publications and shows their most frequent and salient (Chuang et al. 2012) terms (Figure A.8). It uses the topic modeling visualization (C5), which is exclusive to this dashboard and not shown in Figure 4.1. Both the *Citations* and *LDA Topics* dashboards directly use the metadata of the publications and do not aggregate it beforehand.

Filters

Filters (B1) are located in the sidebar on the left and can be configured to select a subset of publications to be visualized. Eight different filters can be applied for each available dashboard (A); six for textual values and two for numeric ones. Once one or more filter values are set, this modification has to be applied and a new data batch loaded through the "Fetch Data/Apply Filters" button (B2). The filters can all be cleared again using the "Clear Filters" button at the top right corner of the sidebar (B1).

All textual filters use auto-completion and regular expression; thus, the user is already presented with suggestions while typing, that match the typed string. The filters *Authors*, *Venues*, and *Publishers* require the user to stop typing for a predefined amount of time (currently set to one second) before suggestions are loaded, as these filters query the backend for the suggestions. For the filters, *Types of papers*, *Field of study*, and *Access type* pre-set values are presented in a drop-down menu, so any suggestions are immediately available, e.g., when clicking on *Types of papers* the suggestions “Article”, “Inproceedings”, “Book”, “Incollection”, “Phdthesis”, and “Mastersthesis” appear. Both numerical filters (i.e., year of publication and citations) function by restricting minimum and maximum values (the filters work inclusive of the selected values). To obtain more information about the filters and their match conditions (e.g., case sensitivity), the user can hover over the question mark icon (?) to the right of their respective filter heading.

Different filters work together through a logical `AND`, values on the same textual filter with a logical `OR`, and values on the same numerical filter with `AND`. For example, if the user decides to search papers from two specific authors in ACL from 2020, the following query would be built: `author=(Jan Philip Wahle OR Terry Ruas) AND (venue=ACL) AND (yearStart=2020 AND yearEnd=2020)`.

There is also a hidden feature to show the co-occurrence of authors or fields of study. If the user filters by a specific author on the *Authors* dashboard, the grid shows the co-authors of that specific author. Similarly, selecting a specific field of study using the filters on the *Fields of study* dashboard shows which other fields of study the selected field of study occurs with the most on the same publication.

Visualizations

CS-Insights uses five different visualization elements across its eight dashboards, which Figure 4.1 (C) shows four of, exemplified for the *Papers* dashboard: *#Papers per year* (C1), *Paper Details Grid* (C2), *#Citations distribution* (C3), and *Top k Papers by #Citations* (C4). Our topic modeling visualization (C5) is shown in Figure 4.6 in Showcase 3. The visualization elements C1-C4 can be exported in several formats (e.g., .csv, .svg, .png) using the three bars in the top right of each element, while C5 offers an “Export” button to export the entire visualization as an HTML file, which keeps all interactive elements intact. All five elements show a loading icon while fetching data from the backend. In the following, we use *[Dashboard]* as a placeholder for the main dashboard element, i.e., the name of the currently selected dashboard. For example “#*[Dashboard]* per year” means “#papers per year” on the *Papers* dashboard and “#venues per year ” on the *Venues* dashboard.

Bar Chart (C1) The ***#[Dashboard]* per year** shows the number of unique dashboard main elements per year. For example, in the *Venues* dashboard, the user can see the bar chart displaying the number of unique venues where the selected papers were published by year. Hovering over a bar reveals the exact number of entries for that year[17]. If the number of entries for a given year is 0, we grey out the year’s label to make it easier for the user to distinguish it from a very small number of entries. Publications without the year set are aggregated to “NA” on the left of the chart, should no filter for the year of publication be selected.

[17] We add a label with the exact value for each year for this thesis, which is always shown, to make the analysis in this thesis more comprehensible.

Grid (C2) The ***[Dashboard]* details grid** displays the available details for each dashboard in a table format. For example, in the *Papers* dashboard, the first column is the paper's title followed by its year of publication, list of authors, venue, number of citations, and link to the actual paper (if available). On the five aggregated dashboards, the grid would change slightly by including the name of the *[Dashboard]* (e.g., the name of the author or venue), the first and last year of publication, and the number of papers, citations, and average citations per paper, while also aggregating all *[Dashboard]* without the main dashboard element set to "Others". A link to the search on DBLP is also added on the *Authors* and *Venues* dashboard, and the grid can generally be sorted using any of the column headings. As some text values can be too long (e.g., paper titles), we abbreviate them for readability purposes, but hovering over them will still display the entire value.

Boxplot (C3) The **#Citations distribution by *[Dashboard]*** visualization shows the distribution of total citations for the selected dashboard main element, e.g., for authors, it will show the distribution of total citations authors have received. On the five aggregation dashboards (so not on *Papers*), the user can also select the number of papers as an alternative metric using the metric switch (B2). Hovering over the boxplot reveals the exact values for the minimum, first quartile, median, third quartile, and maximum. All boxplots are log scaled for better usability, as the maximum often is multiple magnitudes larger than the other values.

Treemap (C4) The **Top k *[Dashboard]* by #Citations** shows the top k elements based on the number of citations. As in C3, the *Papers* dashboard uses only the number of citations (#Citations) as a metric to generate its output, while the five aggregation dashboards also offer the option of showing the top k based on the number of papers (#Papers) in addition. The value of k can be adjusted freely using a text field that reloads the plot automatically. When the text in C4's boxes is too large, we collapse them for readability purposes. Similar to the other visualizations, one can also hover over the chart and show the entire name and value of the selected field.

Topic Modeling (C5) On the *LDA Topics* dashboard, the user can explore the most frequent and salient (Chuang et al. 2012) terms (stemmed words) of a given collection of documents through our topic modeling visualization, which is adapted from Sievert and Shirley (2014). Showcase 3 shows an example of C5 (Figure 4.6) and gives examples of the features of C5 and the *LDA Topics* dashboard. The output in this dashboard is divided into two parts: the semantic topic clusters (left) and the list of the most frequent and salient terms (right). Both parts are generated based on the text in the titles and abstracts of papers. When hovering over a cluster or clicking on it, the 30 most relevant terms of the selected cluster are shown on the right as red bars while continuing to show the overall frequency of those 30 terms in all clusters as blue bars. Once a cluster is selected, the user can adjust the relevance metric (Sievert and Shirley 2014) using a slider in the top right. When no cluster is selected, the plots consider all titles and abstracts to compose their list of terms. The user can also identify clusters associated with a term by hovering over the desired terms directly. Overlap of clusters indicates their semantic proximity.

4.3.3 Showcases

In this subsection, we show some examples to explain and explore what the CS-Insights system can do, so the reader can better understand how we are doing the experiments for our analysis later in this thesis (Chapter 6). During the showcases, we also touch on potential analyses, but we do not discuss the results in detail, as we already cover those in our actual analysis.

Showcase 1: Comparisons with Filters

The first showcase exemplifies the functionality of the filters, by comparing the number of conference papers and journal articles over the last few years. We use the bar chart (C1) on the dashboard *Papers*, as we are interested in the publications over time and set the *Year of publication* filter to 2010–2020 and the *Type of papers* filter to "Article" (Figure 4.2) and "Inproceedings" (Figure 4.3). In the two charts, we can make multiple observations: i) both journal and conference publications increased over time ii) journal publications increased more and overtook conference publications in 2020 iii) conference papers even had a drop in 2020, which we assume to be related to the COVID-19 pandemic. We further investigate the differences between journals and conferences in Section 6.6, which includes a more detailed look into their number of publications over time.

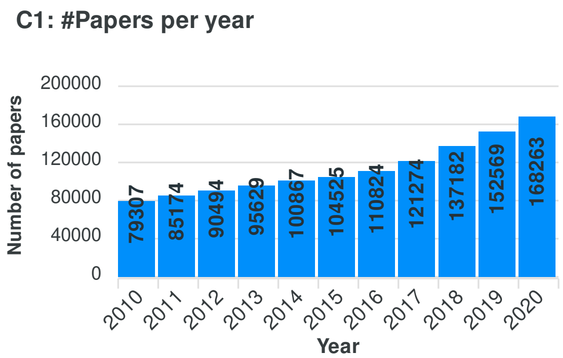


Figure 4.2 Number of journal articles per year between 2010 and 2020.

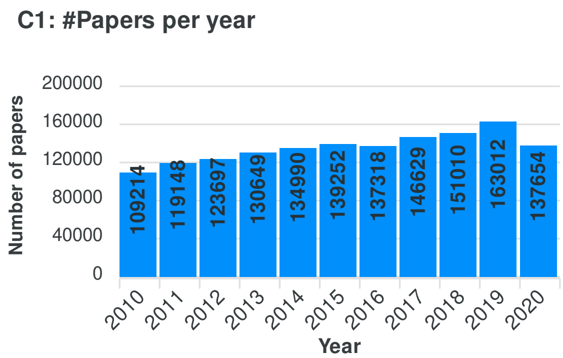


Figure 4.3 Number of conference papers per year between 2010 and 2020.

Showcase 2: Dashboard–Filter Interaction

We use the second showcase to explain the interaction of filters and dashboards and when the user has to use a filter, and when the corresponding dashboard. Our first example shows how to find the authors who published the most in CVPR (Computer Vision and Pattern Recognition). For this, we select the *Authors* dashboard, “CVPR” in the *Venue* filter, and click the heading “Papers” in the grid (C2) twice to sort by the papers descending[18]. The result (Figure 4.4) reveals that “Luc Van Gool” published the most papers in “CVPR” (122 papers, the first in 1991, the last in 2020).

C2: Grid

Author	First	Last	↓ Papers	Citations	Citations/Paper	Link
Luc Van Gool	1991	2020	122	15,194	124.54	Link
Xiaoou Tang	2004	2020	92	21,404	232.65	Link
Thomas S. Huang	1988	2020	88	15,456	175.64	Link
Ming-Hsuan Yang 0001	1998	2020	87	22,399	257.46	Link
Marc Pollefeys	1997	2020	85	7,449	87.64	Link
Xiaogang Wang 0001	2004	2020	84	21,490	255.83	Link
Trevor Darrell	1988	2020	79	31,249	395.56	Link
Shuicheng Yan	2005	2020	79	8,971	113.56	Link

EXPORT Rows per page: 100 1–100 of 17535

Figure 4.4 Authors that published the most in the CVPR conference.

The other way around, if we want to find out which venue a specific author published in the most, we swap what we select as the filter and what as the dashboard. We select the *Venues* dashboard, “Luc Van Gool” in the *Authors* filter, and again click the “Papers” heading twice to sort by paper descending. The result (Figure 4.5) shows that Luc Van Gool published the most in CVPR (122 papers), with the second-ranked venue being ECCV (63 papers).

C2: Grid

Venue	First	Last	↓ Papers	Citations	Citations/Paper	Link
CVPR	1991	2020	122	15,194	124.54	Link
ECCV	1994	2020	63	18,346	291.21	Link
ICCV	1990	2019	53	6,834	128.94	Link
Int. J. Comput. Vis.	1999	2021	38	20,210	531.84	Link
CVPR Workshops	2003	2021	31	1,051	33.90	Link
WACV	2009	2021	28	839	29.96	Link
BMVC	1993	2015	27	1,468	54.37	Link
IEEE Trans. Pattern Anal. Mach. Intell.	1995	2020	24	3,638	151.58	Link

EXPORT Rows per page: 100 1–100 of 166

Figure 4.5 Venues Luc Van Gool published the most in.

[18] An alternative way to sorting the grid is to switch the metric to “#Papers” and then looking at the first entry of the treemap.

Showcase 3: Topic Modeling

In our last showcase, we demonstrate the topic modeling component (C5). We want to investigate the topics of CVPR between 2000 and 2005 and select the *LDA Topics* dashboard, a fitting model in the toolbar (B2)[19], and the filters *Year of publication* 2000–2005 and *Venues* "CVPR". Figure 4.6 shows the results and Figure A.8 the entire dashboard). The terms "track", "imag", and "detect" are the three most salient, while "imag", "model", and "method" are the three most frequent. Clicking on topic cluster 1 or hovering over it reveals the adjusted distribution of terms for that topic considering their relevancy, where "model", "imag", and "approach" are the top three (Figure A.9). Hovering over the term "track" reveals, that it appears most in topic cluster 5 (Figure A.10). In Section 6.4 we are further investigating the topics of venues and how they change over time.

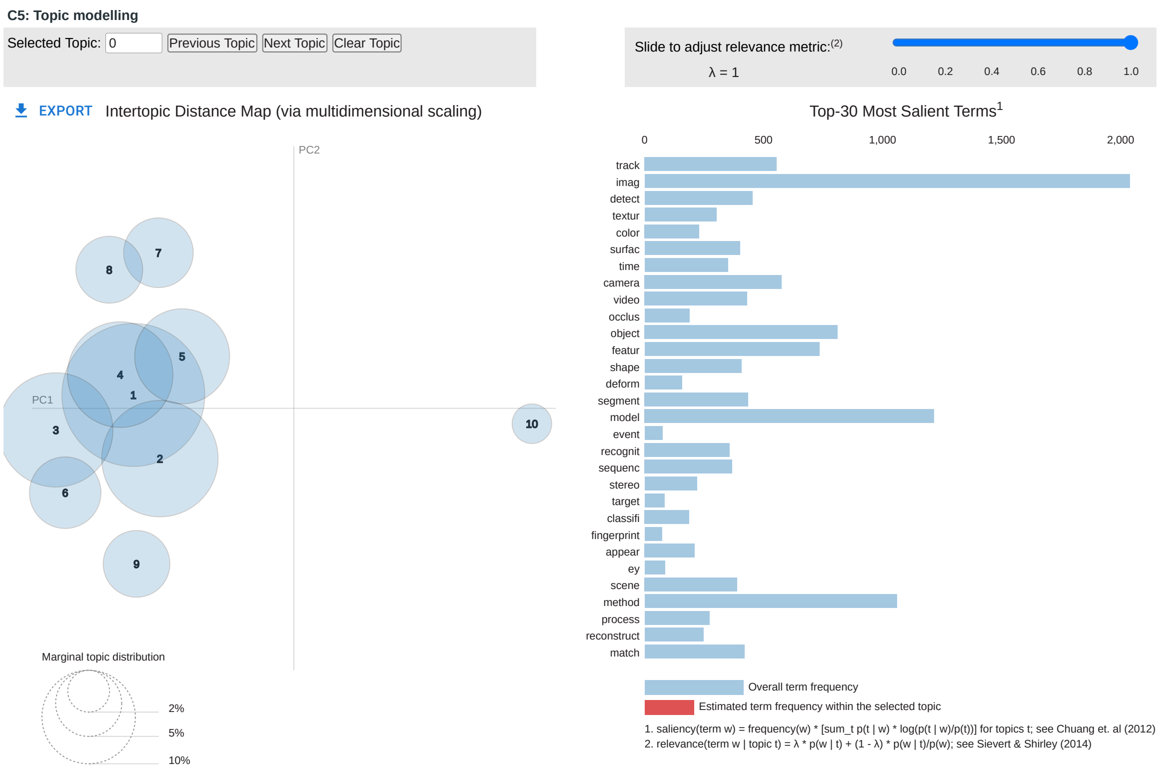


Figure 4.6 Topic modeling visualization (C5) for the CVPR conference 2000-2005.

[19] Currently, the user can select any of the available models, as they yield the same results.

5 Implementation

This chapter covers aspects regarding the implementation of CS-Insights. We detail the architecture of CS-Insights and its sub-components (Section 5.1) and our measures to improve the quality of our code (Section 5.2).

5.1 Architecture

The CS-Insights system consists of the four sub-components *frontend*, *backend*, *prediction endpoint*, and *crawler*, which we can see in Figure 5.1. Our system is available as a free web application without the need for any installation as it runs in any web browser[1], providing access to multiple users simultaneously. The entire code for all components is available online on GitHub and accessible through the main GitHub repository[2], so researchers can also run the code locally on their machine if they choose to. In this case, even though CS-Insights is split into multiple components, other researchers only have to interact with the main component[2], as all sub-components are managed from there automatically. To guarantee a flexible and modular setup, every sub-component in CS-Insights runs in its own docker container[3]. The architecture discussed in this section is also part of our submission to arXiv/EACL (Ruas et al. 2022).

5.1.1 Frontend

The frontend[4] is responsible for presenting the main components of our system (i.e., dashboards, filters, and visualizations), and through the frontend's interface, the user can filter, retrieve, and visualize the metadata of CS publications (see Section 4.3.2). We use TypeScript[5] and as web framework React[6] because it is open-access and has a large community support[7]. For charts we use ApexCharts[8] and for other UI components, we use Material UI[9].

[1] `https://cs-insights.uni-goettingen.de/`
[2] `https://github.com/gipplab/cs-insights-main`
[3] `https://www.docker.com`
[4] `https://github.com/gipplab/cs-insights-frontend`
[5] `https://www.typescriptlang.org`; a superset of JavaScript
[6] `https://reactjs.org/`
[7] `https://www.statista.com/statistics/1124699/worldwide-developer-survey-most-used-frameworks-web/`
[8] `https://apexcharts.com/react-chart-demos/`
[9] `https://mui.com/`

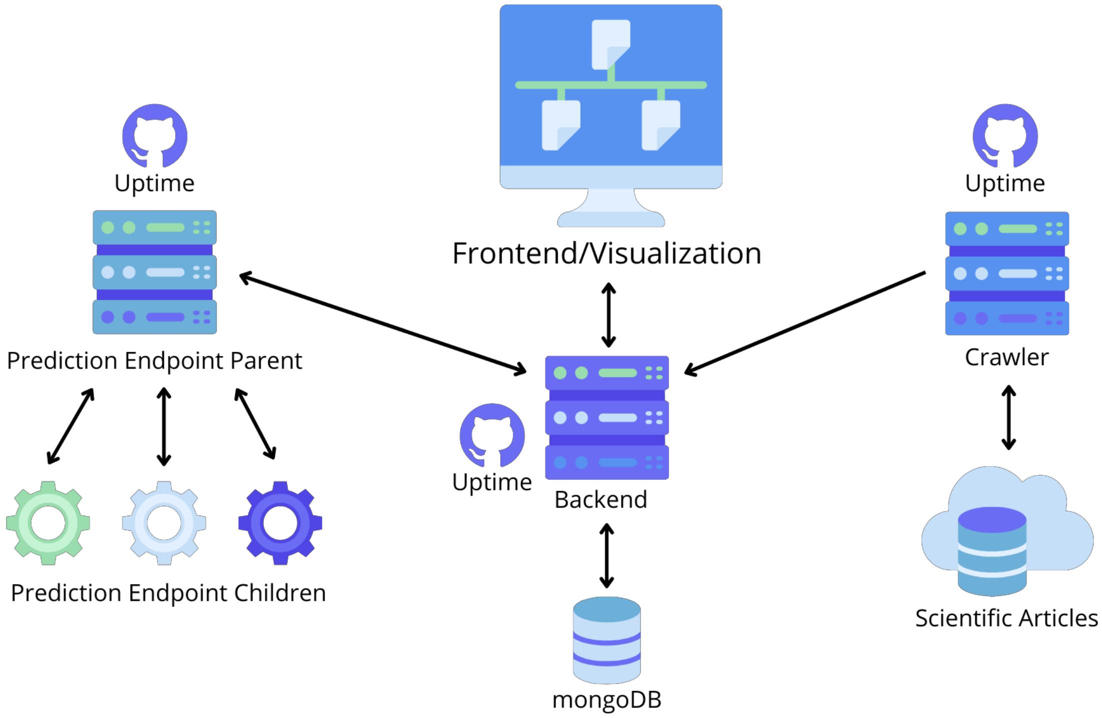


Figure 5.1 Overview of the CS-Insights system.

5.1.2 Backend

The backend[10] serves as REST API to access, retrieve, aggregate, and analyze data (see Section 4.2). It controls who can access data and how, and performs computationally expensive tasks (e.g., accumulating citations of all authors for each paper available). CS-Insights uses MongoDB as a database with mongoose[11] providing the object document mapping. We also use TypeScript with Node.js[12] as JavaScript runtime, and Express.js[13] to handle the HTTP(S) requests. The CRUD endpoints of its API are auto-generated from the mongoose models with ERM, while the endpoints the frontend uses are manually written (Section 4.2.2).

[10] https://github.com/gipplab/cs-insights-backend
[11] https://mongoosejs.com/
[12] https://nodejs.org/en/
[13] https://expressjs.com/

5.1.3 Prediction Endpoint

The prediction endpoint[14] is implemented in Python 3, is responsible for the training and prediction of the models in the *LDA Topics* dashboard, and is used to generate the semantic topics and their respective lists of the most frequent and salient terms. For topic modeling, we use gensim's[15] implementation of LDA (Blei et al. 2003). The visualizations are implemented using pyLDAvis[16], a port of LDAvis (Sievert and Shirley 2014). As the training and inference require processing thousands of documents, we create a dedicated service to maintain these models, distribute them on the available compute infrastructure, assign them to compute jobs, and consolidate all results. The endpoint consists of a manager parent node and a variable amount of child nodes, where the parent node takes requests through a REST API. It also abstracts topic model creation, training, and inference to distribute the computational load across different independent child nodes. To preprocess the text (i.e., titles and abstracts) before we create the topics we use `preprocess_string()`[17] from gensim, which removes HTML tags, punctuation, duplicate whitespaces, numbers, stopwords, and words with less than three characters, and stems the text.

5.1.4 Crawler

The crawler[18] is also implemented in Python 3 and creates our dataset D3 (see Section 4.1). In the future, the crawler can be used to keep CS-Insights and D3 up-to-date with the most recent publications, by running it monthly to add new publications and update existing ones (Section 7.2). We use aiohttp[19] to request the full-texts and GROBID (Lopez 2022) to extract the metadata from the full-texts, which is a resource-intensive process. Therefore, we implement a parallel and asynchronous routine to parse the latest release, retrieve the corresponding full-texts, extract their metadata, and align the information to DBLP. We split the dataset into equally sized chunks to work on mutually exclusive parts of the dataset with multiple processes without processing the whole repository at once. Then, we launch n processes to retrieve publications, where each process asynchronously requests the PDF link of a paper or, if not present, parses the HTML page of the paper to identify the PDF link and downloads it. To restrict requests to the same domain and respect server limits, we use semaphores and wait to respect the retry-after header whenever we receive an HTTP 429 ("Too many requests") response. In parallel to the n retrieval processes, we launch another n processes to work on full-texts of the previously downloaded chunks and extract their metadata. To reduce disk requirements, we delete the full-texts after

[14] https://github.com/gipplab/cs-insights-prediction-endpoint
[15] https://radimrehurek.com/gensim/models/ldamodel.html
[16] https://pyldavis.readthedocs.io/en/latest/readme.html
[17] https://radimrehurek.com/gensim/parsing/preprocessing.html
[18] https://github.com/gipplab/cs-insights-crawler
[19] https://docs.aiohttp.org/en/stable/

extraction and only keep their metadata. The uncompressed size of D3 is ≈18GB in JSON format and ≈21.5GB in CSV format (Wahle et al. 2022).

5.2 Quality Assurance

To ensure the quality of CS-Insights and its code we employ various measures as Table 5.1 shows. We use tests to make sure the logic of our components works

Component	Tests	Linting	Typing	Code Style
Frontend	-	ESLint[20]	TypeScript	Airbnb[21]
Backend	Mocha[22]	ESLint	TypeScript	Airbnb
Prediction Endpoint	pytest[23]	Flake8[24]	mypy[25]	Black[26]
Crawler	pytest	Flake8	Pyright[27]	Black

Table 5.1 Technologies used across the four components of CS-Insights for testing, linting, typing, and code styling.

correctly and linting for static code analysis to find potential problems (e.g., unused variables, unused imports). Checks for typing are added to avoid potential errors from dynamically typed languages, and a common code style is used for better readability. Tests, linting, typing, and code style are enforced by checking the code whenever a commit is pushed to one of our GitHub repositories. The frontend does not use tests, because normal tests make little sense in our case, where we have a “dumb” frontend, that only visualizes data and does not aggregate any data itself. UI tests were considered but postponed to future work due to the time constraints and little additional value considering the amount of work required.

Documentation for CS-Insights can be found in the README of the main project[28] and more detailed documentation will also be added soon[29]. Automatically generated documentation for the available endpoints of the backend[30] and prediction endpoint[31] is already available and will be available soon for the frontend[32].

We also track the uptime of all components from a separate GitHub repository[33] using upptime[34], which automatically creates GitHub issues, when a component goes down and closes the issue when it comes back up again.

[28] https://github.com/jpwahle/cs-insights/blob/main/README.md
[29] https://jpwahle.github.io/cs-insights/
[30] https://jpwahle.github.io/cs-insights-backend/
[31] https://jpwahle.github.io/cs-insights-prediction-endpoint/
[32] https://jpwahle.github.io/cs-insights-frontend/
[33] https://github.com/gipplab/cs-insights-uptime
[34] https://github.com/upptime/upptime

6 Analysis and Discussion

In this chapter, we conduct experiments to answer our research questions (Section 1.2) and discuss their results. We first explain the setup for our experiments including a data overview (Section 6.1), before starting with the experiments for our analysis. The experiments are split into multiple sections, each covering one attribute in our data. We section the experiments as follows: publications (Section 6.2; Experiments 1 to 2), authors (Section 6.3; Experiments 3 to 8)), venues (Section 6.4; Experiments 9 to 14), citations (Section 6.5; Experiments 15 to 17)), document types (Section 6.6; Experiments 18 to 24)), and fields of study (Section 6.7; Experiments 25 to 27)). Every experiment references the research question it relates to at the end of its title and includes a discussion of the results. Lastly, we close out the chapter with a short summary (Section 6.8).

Disclaimer Some numbers and results might differ from the paper published in LREC (Wahle et al. 2022), as the data we use is from a CSV export of D3. The limitations of our work are further explained in Section 7.2.

6.1 Setup

This section shortly explains the general setup of the experiments (Section 6.1.1) and the setup for specific groups of experiments (Section 6.1.2).

6.1.1 General Setup

We conduct all experiments in this chapter with CS-Insights and its underlying data from D3, by leveraging its various dashboards and filters in specific ways. Table 6.1 shows an overview of the data from D3, which spans from 1936 to 2022. The showcases in Section 4.3.3 already demonstrated some ways CS-Insights can be used to visualize this data and interact with it. For our experiments, we use all dashboards and filters except *Publishers* and *Access type*. We exclude *Publishers*, because the data is too sparse, and exclude *Access type*, as this is not a focus of our analysis. CS-Insights's dashboards also consist of multiple visualization elements, of which we use the bar chart (C1), grid (C2), boxplot (C3), and topic modeling component (C5) in multiple experiments. Only the treemap is not used, as the grid provides better formatted and readable results for our analysis and shows exact numbers. This results in many possible combinations of specific dashboards, filters, and visualizations, which are too many to detail here.

Attribute	Amount
Publications	4,893,540
Abstracts	3,980,144
Citations	97,053,288
Authors	2,730,729
Venues	14,268
Types of paper	7
Fields of study	20

Table 6.1 Number of unique entries for each field in D3.

We extract the results from the experiments with the integrated export functionality for C1-C4 in the format we see fit best for the corresponding experiment (as images or mostly .csv), and for the topic analysis (C5) by copying the list of the most salient terms. The results are then either directly included or are preprocessed first, by further aggregating the results to better highlight important aspects and making some results more comprehensible. In the analysis in this thesis, we make sure not to go further than two layers deep with our aggregation of results to avoid manual analysis, which would contradict the purpose of CS-Insights. For example, directly taking results from a query would be one layer, and aggregating results from multiple (e.g., five) queries into one table would be layer two.

6.1.2 Experiment Specific Setup

Many experiments share a specific setup, so to avoid repetitiveness during those experiments, we give details in this paragraph.

Citation and Paper Distributions over distinct Time Periods We investigate the distribution of citations and papers over eight distinct periods for authors (Experiment 5), venues (Experiment 11), incoming/outgoing citations (Experiment 17, and conferences/journals (Experiment 20) with the same setup. The first four periods (1960-1999) use a span of 10 years and the other four (2000-2019) use five years. This avoids the tables from getting too cluttered, compared to using 12 periods of five years, though we are aware of the issues different lengths of time spans can cause (Section 7.2). We leave out data before 1960, because the data is too sparse for meaningful analysis, from 2020, because we want to avoid the influence of the COVID-19 pandemic, whose effect becomes apparent in Experiments 1 and 19, and from 2021 because our data is not complete for that year (Section 7.2). The distributions of citations and papers consist of the first quartile, median, third quartile, maximum, and average. We leave out the minimums, as for the number of citations they are always 0 and for papers always 1.

Topic Modeling We use our topic modeling component to generate 10 topics and find the 30 most salient terms (stemmed words in a ranked list) for subsets of our data and then compare the lists for different subsets. For this, we put the 30 most salient terms of each subset into one table column each and then mark in bold which terms are unique to a column and in italics which are common across all columns. We use this approach to investigate the trends of the subsets over time, by selecting five distinct periods and generating the most salient terms for each period, but again only use data until 2019 to avoid the influence of the COVID-19 pandemic. This approach is applied to the most cited and productive authors (Experiment 8) and venues (Experiment 14), and conferences/journals (Experiment 24). Alternatively, we compare five distinct entities, i.e., the five most cited or most productive venues (Experiment 13) and the five most prominent fields of study (Experiment 27). The topic modeling experiments do not aggregate any data and directly use the publications. Our approach follows that of Fiala and Tutoky (2017), who also split their data into distinct periods and highlight terms unique to a period.

6.2 Publications

This short section focuses on experiments conducted directly on the publications of CS-Insights without any further aggregations and answers a part of RQ1 by looking into the number of publications per year (Experiment 1) and the most cited publications (Experiment 2).

Experiment 1: How does the number of publications change over time? (RQ1)

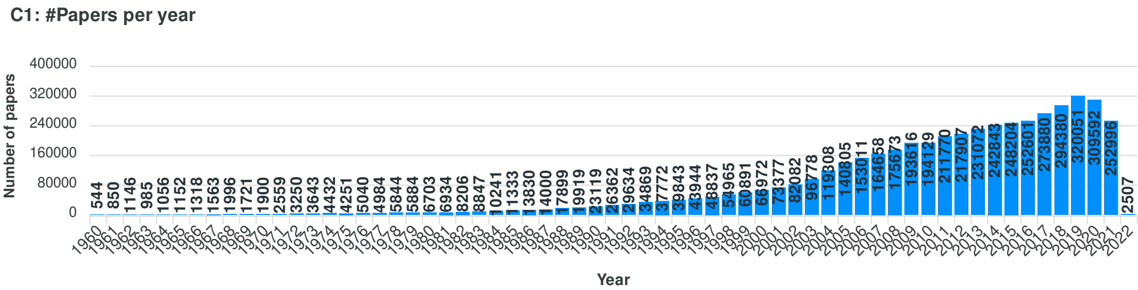


Figure 6.1 Number of publications per year starting in 1960. See Figure A.11 for the full span.

In total CS-Insights comprises 4,893,540 publications between 1936 and 2022, with 276 publications missing a year of publication. The first thing to notice is the exponential increase in publications in the 2000s and a peak around 2019 (**Figure 6.1**). Bornmann et al. (2021) also see an exponential increase in scientific publications overall (not just in CS) and find a doubling time of 14 years since 1952, while our data shows that in 14 years (2005 to 2019) the number of publications more than doubles. A doubling time of 12 years (2007-2019) might be more appropriate for recent years and an even smaller doubling time for earlier years (e.g., in 1990-1997 and 2000-2005 the number of papers also doubled). This highlights the boost in CS publications compared to other fields of study very well. We also observe an increase in publications during the end of the 1980s and 1990s, which we think is caused by the more widespread adoption of personal computers[1] and the internet[2], respectively. Similar increases between the 1980s and 2010s can be observed in Web of Science (Fiala and Tutoky 2017) and NLP (Mohammad 2020b; Mariani et al. 2019a), with Mohammad (2020b) also specifically mentioning the observations. The peak around 2019 is not visible

[1] https://www.statista.com/statistics/214641/household-adoption-rate-of-computer-in-the-us-since-1997/

[2] https://www.statista.com/statistics/189349/us-households-home-internet-connection-subscription/

in Fiala and Tutoky (2017), Mariani et al. (2019a), and Mohammad (2020b), as their data does not cover 2019. Mohammad (2020b) and Mariani et al. (2019a) also both observe a decrease in papers every second year which they both attribute to biennial conferences, but this trend is not visible in our data. We hypothesize the great difference in dataset size (CS-Insights is nearly 100 times larger) and broader coverage of topics (CS vs. NLP) cause biennial conferences to not affect the overall number of publications or the different biennial conferences compensate for each other. The drop in publications in 2020 can be explained by the COVID-19 pandemic.

Experiment 2: What are the most cited publications?

#	Title	Year	Authors	Venue	#Citations
1	Genetic Algorithms in Search Optimization and Machine Learning	1988	David E. Goldberg	*Others*	57,583
2	Long Short-Term Memory	1997	Sepp Hochreiter, Jrgen Schmidhuber	Neural Comput.	45,635
3	Elements of Information Theory	1991	Thomas M. Cover, Joy A. Thomas	*Others*	42,099
4	LIBSVM: A library for support vector machines	2011	Chih-Chung Chang, Chih-Jen Lin	ACM Trans. Intell. Syst. Technol.	39,111
5	The Nature of Statistical Learning Theory	2000	Vladimir Vapnik	*Others*	38,124
6	Convex Optimization	2006	Stephen P. Boyd, Lieven Vandenberghe	*Others*	37,926
7	#p	2017	Gorjan Alagic, Catharine Lo	Quantum Inf. Comput.	37,732
8	A fast and elitist multiobjective genetic algorithm: NSGA-II	2002	Kalyanmoy Deb, Samir Agrawal, Amrit Pratap, T. Meyarivan	IEEE Trans. Evol. Comput.	30,893
9	Reinforcement Learning: An Introduction	2005	Richard S. Sutton, Andrew G. Barto	IEEE Trans. Neural Networks	30,815
10	Matrix analysis	1985	Roger A. Horn, Charles R. Johnson	*Others*	29,323

Table 6.2 Top 10 most cited publications.

We observe different topics are covered by the most cited publications (**Table 6.2**), but machine learning appears to be the most prominent topic with multiple publications. Half of the publications in this list have *Others* as the venue, which in this experiment means the publications are books. The publication titled "#p" has the same name in DBLP, so this is not an error on our end, though Google Scholar shows it with its full name "Quantum invariants of 3-manifolds and NP vs #P". It also only has very few citations on Google Scholar[3] and Semantic Scholar[4] compared to 37,732 citations in CS-Insights. We believe this is due to an error in the matching of the citations (Section 4.1.3), because of the broken and very short title. Our top 10 is entirely different from the top 20 of Fiala and Tutoky (2017), which might be because of the different underlying datasets (DBLP vs. Web of Science). Only our #8 shows

[3]https://scholar.google.de/scholar?oi=bibs&hl=de&cluster=13152820713440837432
[4]https://api.semanticscholar.org/CorpusID:233455290

up as #17 on their list. There are no overlaps with the top 20 from Mariani et al. (2019a) and the top 15 from Mohammad (2020b), which we explain by them only using publications from NLP. We also note some author names are missing special characters (Section 7.2).

6.3 Authors

This section covers experiments that aggregate the data in CS-Insights by the authors and then analyzes the authors to answer parts of RQ1, RQ2, and RQ3. We start with general trends for the number of authors and their number of publications and citations (Experiments 3 to 5), before covering the most cited and most productive (i.e., most published) authors, and what topics/venues they publish in (Experiments 6 to 8).

Experiment 3: How many authors publish in CS-Insights per year? (RQ1)

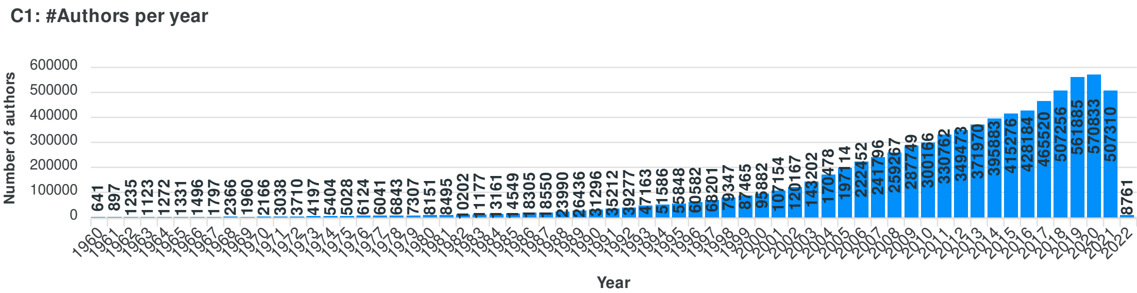


Figure 6.2 Number of unique authors per year starting in 1960. See Figure A.12 for the full span.

In total there are 2,730,729 authors in CS-Insights, while there are 54,604 publications without any authors listed. Similar to the number of publications (Experiment 1) there is a continuous growth in the number of authors (**Figure 6.2**), even in 2020, though the increase was smaller between 2019 and 2020 than between 2018 and 2019. The patterns from Experiment 1 are also reflected; there has been a major increase since the 2000s, which began as a spurt in the late 1980s, and a peak around 2020 (for publications 2019). Mohammad (2020b) and Mariani et al. (2019a) observe similar trends in NLP and Mohammad (2020b) concludes that attracting more researchers every year means the research field is in good health.

Experiment 4: How many authors published in the last x years? How many authors are new to CS? (RQ1)

Time span	1 year	2 years	3 years	4 years	5 years	Total
Years	2020	2019-2020	2018-2020	2017-2020	2016-2020	1936-2020
#Authors	570,833	875,697	1,081,137	1,243,146	1,378,348	2,579,224
%Authors	22.13%	33.95%	41.92%	48.20%	53.44%	100.00%
#Authors new	183,559	370,122	534,718	683,923	817,879	
%Authors new	7.12%	14.35%	20.73%	26.52%	31.71%	
Years (inv.)	1936-2019	1936-2018	1936-2017	1936-2016	1936-2015	
#Authors	2,395,665	2,209,102	2,044,506	1,895,301	1,761,345	

Table 6.3 Number of authors who were active (at least one publication) in the last x years compared to the total amount of authors. Also includes the number of new authors and their percentage considering the total number of authors (calculated using the inverted time span).

We include the number of authors that were active before the last x years at the bottom (inverted time span), to determine the number of new authors by calculating the difference between authors in the inverted time span and total authors (**Table 6.3**).

In 2020 over one-fifth of all authors published at least one paper, over a third in the last two years, and over half of all researchers in CS-Insights in the last five years. Over 30% of all authors in CS-Insights only joined CS in the last five years, which also infers that more than half of the authors that published between 2016 and 2020 were new authors. From this, it again becomes apparent that the field is massively growing but also very active and healthy. Similar trends for more and more new authors joining can also be observed in NLP (Mariani et al. 2019a). Wahle et al. (2022) give some more insights into the activity of authors in D3, which we cannot reproduce with CS-Insights's UI at the current time.

Experiment 5: How are the citations and papers distributed across authors? How do the distributions change over time? (RQ2)

It is not possible to compute the average number of authors per paper (or papers per author) purely from CS-Insights's UI, as it only supplies the unique number of authors per year and not the total number of authors per year, so authors who publish multiple papers in a year would only count once.

The overall citations in the investigated period show a median of 11 citations with the first quartile being at 2 citations and the third quartile being at 51 citations, meaning half of all authors received between 2 and 51 citations (**Table 6.4**). This is a difference from NLP (Mariani et al. 2019a), where 42% of authors have no citations, while our first quartile (25%) already has 2 citations. They conclude the high percentage of authors without citations is due to many citations coming from neighboring domains not covered in their dataset. As we include the entirety of CS and not only a sub-field, this issue is smaller in our work. For the citations, we also

Time span	#Citations Q1	Med.	Q3	Max.	#Papers Q1	Med.	Q3	Max.
1960-1969	0	3	19	20,532	1	1	2	39
1970-1979	0	2	16	19,871	1	1	2	78
1980-1989	0	3	24	57,729	1	1	2	171
1990-1999	0	7	44	48,209	1	1	3	256
2000-2004	1	12	61	52,250	1	1	3	214
2005-2009	3	17	70	53,063	1	1	3	360
2010-2014	3	14	55	41,367	1	1	3	399
2015-2019	2	8	30	89,647	1	1	3	586
1960-2019	2	11	51	133,020	1	2	4	1,332

Table 6.4 Distribution of the number of total citations and papers over authors per time period showing the first quartile, median, third quartile, and maximum. The upper block covers 10 years per time period and the lower block five years.

see a peak in 2005-2009, which slowly falls off for the earlier years and more quickly for the more recent years. While this trend might appear interesting, we later see in Experiment 15 that it mirrors the overall trend for citations quite well. The maximum for citations fluctuates but also shows a general trend upward, meaning singular recent papers get cited more, most likely due to the increase in publications (Experiment 1) and researchers not looking so far back for references (Fiala and Tutoky 2017).

For the number of papers, the distribution nearly stays the same but shows a slight increase since 1990. A general increase is visible in the maximum, which shows singular authors push to always publish more papers in the same span. The low median makes sense as most authors in our dataset (45.85%) only published one paper (Wahle et al. 2022). This is even more extreme in NLP, where Mohammad (2020b) shows 57.9% of all authors only published one paper in NLP. Considering the large number of authors in recent years (Experiment 3), and many of them being new contributors (Experiment 4), it is possible they have only published one paper so far, or quickly dropped out of the field again. Determining the exact reason requires more research in the future, e.g., by investigating the number of papers authors publish, similar to Wahle et al. (2022), who show only very few authors stay active in CS for a long time.

Experiment 6: Who are the most cited and most productive authors? (RQ3)

It is interesting to see that the most productive and most cited authors have no overlap (**Table 6.5**). We also note not only are the total citations higher for the most cited authors but also the average citations. Currently, it seems to be a quantity vs. quality matter, but subsequent experiments (Experiments 7 and 8) show the topics the authors mainly cover are different (and thus also the venues they publish in), which could cause this. Franceschet (2010) also looks into the top 10 authors based on publications by using DBLP data from 2010. In his work, Philip S. Yu was #1 with 547 publications

#	Author (#Citations)	First	#Papers	#Citations	Avg. Citations
1	*Others*	1936	54,604	676,548	12.39
2	Ross B. Girshick	2004	69	146,867	2,128.51
3	Anil K. Jain 0001	1974	662	123,682	186.83
4	Kaiming He	2009	66	114,330	1,732.27
5	Jitendra Malik	1987	231	109,821	475.42
6	Andrew Zisserman	1985	454	105,025	231.33
7	Li Fei-Fei 0001	2003	194	102,735	529.56
8	Luc Van Gool	1984	784	96,530	123.13
9	Jiawei Han 0001	1985	874	95,371	109.12
10	Trevor Darrell	1987	288	91,451	317.54
	Average	1991	402	109,535	648.19
#	**Author (#Papers)**	**First**	**#Papers**	**#Citations**	**Avg. Citations**
1	*Others*	1936	54,604	676,548	12.39
2	H. Vincent Poor	1977	1,649	74,467	45.16
3	Mohamed-Slim Alouini	1997	1,445	39,300	27.20
4	Lajos Hanzo	1989	1,382	35,887	25.97
5	*Wei Wang	1986	1,334	22,805	17.10
6	Philip S. Yu	1980	1,288	73,436	57.02
7	*Lei Zhang	1992	1,269	12,299	9.69
8	*Yu Zhang	1991	1,261	11,380	9.02
9	Victor C. M. Leung	1982	1,260	30,704	24.37
10	*Yang Liu	2001	1,247	8,778	7.04
	Average	1988	1,348	34,340	24.73

Table 6.5 Top 10 authors based on the number of citations received (top) and publications (bottom). The average is computed excluding *Others*. Asterisks (*) denote entries, which refer to disambiguation pages in DBLP and not singular authors.

and can still be seen in our list at #6, now with 1288 publications. The other authors who were in the top 10 in 2010 are not in the top 10 anymore but are now somewhere in the top 100, e.g., Elisa Bertino was #3 with 494 publications and is now #31 with 966 publications. As with the most cited publications (Experiment 2), there is no overlap to NLP-specific authors, both for the most cited and most productive authors (Mariani et al. 2019a). This makes sense once we look into the next experiments, which investigate the topics of the most cited and most productive authors and show NLP is not among the most researched topics.

Experiment 7: What are the preferred venues/topics of the most cited and most productive authors? (RQ3)

To determine the topics we look into the venues of the top 5 authors (ignoring *Others* and disambiguation entries) by citations and publications (**Table 6.6**). We are aware that the top 5 authors might not resemble every author in CS-Insights, but we believe they are a good approximation, and some valuable insights can be gained.

In the top and bottom halves themselves, there is not much difference between the venues the authors get most cited in and most published in, but there are obvious differences between the top and bottom halves. The top 5 venues for the most cited authors show a clear trend toward the topics of computer vision and pattern recognition, both for the venues these five authors got most cited in and most published in. Most venues are conferences, with the most reoccurring being CVPR (Computer Vision and Pattern Recognition), ICCV (International Conference on Computer Vision), and ECCV (European Conference on Computer Vision), but the journal IEEE Trans. Pattern Anal. Mach. Intell. also appears just as often as CVPR. On the other hand, the top 5 venues of the most productive authors appear to be more on the engineering side of CS and focus on signal processing, communication, and information theory. Most of the venues in this list are also IEEE journals, with a few conferences in between. The topics of the most cited and most productive authors also explain why there is no overlap with the top authors from NLP (Experiment 6); the most covered topics in CS-Insights do not include NLP.

Experiment 8: Do the topics of the most cited and productive authors change over time? (RQ3)

In **Table 6.7** the first period includes all publications before 2000 to reduce issues with data sparsity in the earlier years, similar to Fiala and Tutoky (2017), and starts with the year of the first publication, i.e., 1974 and 1977.

We observe the findings from the 30 most salient words align with the general topics of the venues the authors publish and get cited in the most (Experiment 7), and thus barely overlap. For the most cited authors, the three most salient terms over the whole period “imag”, “fingerprint”, and “textur” match the computer vision and pattern recognition focus of their preferred venues, which “object”, “motion”, and “track” also support. Similarly, the terms “object”, “imag”, “match”, “recognit”, and “fingerprint” appear in all five periods, again indicating a strong tie to the focus of the venues. In 2000-2004 terms like “face”, “biometr”, and “shape” first appear, indicating a greater focus on biometrics. The most salient term “cluster” (1974-2009) is replaced by “latent” in 2010-2014, which we hypothesize to be related to clustering algorithms with the term “latent” (e.g., Latent Dirichlet Allocation (LDA)) gaining more popularity. We also note the first appearance of “network”, “train”, and “cnn” in 2015-2019, which shows the rise of approaches leveraging neural networks. Fiala and Tutoky (2017) show the popularity of the term “neural network” already before 1999 and a decline in popularity for “neural network” after 2005. On the other hand, Tattershall et al. (2020)

Author (#Citations)	Venue (#Citations)	Venue (#Publications)
Ross B. Girshick	CVPR (53,294) IT. Pattern Anal. Mach. Intell. (53,179) ICCV (22,197) ACM Multimedia (13,371) ECCV (4,566)	CVPR (30) ICCV (12) IT. Pattern Anal. Mach. Intell. (9) ECCV (8) ACM Multimedia (2)
Anil K. Jain 0001	IT. Pattern Anal. Mach. Intell. (35,962) ACM Comput. Surv. (13,647) *Others* (10,794) ICPR (7,387) IEEE Trans. Inf. Forensics Secur. (5,375)	IT. Pattern Anal. Mach. Intell. (104) ICPR (75) Pattern Recognit. (49) ICB (41) IEEE Trans. Inf. Forensics Secur. (35)
Kaiming He	IT. Pattern Anal. Mach. Intell. (53,089) CVPR (22,051) ICCV (21,211) ECCV (16,580) ECCV (13) (1,309)	CVPR (29) ICCV (13) IT. Pattern Anal. Mach. Intell. (11) ECCV (10) ACM Trans. Graph. (1)
Jitendra Malik	CVPR (46249) IT. Pattern Anal. Mach. Intell. (32,564) ICCV (12,451) ECCV (8,048) SIGGRAPH (3,876)	CVPR (65) ICCV (41) ECCV (29) IT. Pattern Anal. Mach. Intell. (24) Int. J. Comput. Vis. (12)
Andrew Zisserman	CVPR (25,570) Int. J. Comput. Vis. (21,130) ICCV (16,604) BMVC (9,016) ECCV (8,992)	CVPR (70) BMVC (53) ECCV (45) ICCV (41) Int. J. Comput. Vis. (31)
Author (#Papers)	**Venue (#Citations)**	**Venue (#Publications)**
H. Vincent Poor	IEEE Trans. Inf. Theory (10,369) IEEE Trans. Signal Process. (7,207) IEEE Trans. Commun. (7,005) IEEE Trans. Wirel. Commun. (6,723) IEEE Signal Process. Mag. (5,384)	IEEE Trans. Inf. Theory (134) ISIT (132) IEEE Trans. Wirel. Commun. (123) IEEE Trans. Commun. (109) IEEE Trans. Signal Process. (99)
Mohamed-Slim Alouini	IEEE Trans. Commun. (8,637) IEEE Trans. Wirel. Commun. (6,993) Others (4,468) IEEE Trans. Veh. Technol. (2,241) ICC (2,022)	IEEE Trans. Wirel. Commun. (164) IEEE Trans. Commun. (138) ICC (105) GLOBECOM (84) IEEE Trans. Veh. Technol. (83)
Lajos Hanzo	IEEE Trans. Veh. Technol. (5,462) IEEE Commun. Surv. Tutorials (4,796) IEEE Trans. Commun. (3,665) Proc. IEEE (3,193) IEEE Trans. Wirel. Commun. (2,928)	IEEE Trans. Wirel. Commun. (164) IEEE Trans. Commun. (138) ICC (105) GLOBECOM (84) IEEE Trans. Veh. Technol. (83)
Philip S. Yu	KDD (9,702) IEEE Trans. Knowl. Data Eng. (8,442) Knowl. Inf. Syst. (5,494) SIGMOD Conference (5,260) ICDM (4,437)	IEEE Trans. Knowl. Data Eng. (94) ICDM (77) SDM (74) KDD (73) CIKM (71)
Victor C. M. Leung	IEEE Commun. Mag. (3,498) IEEE Trans. Veh. Technol. (2,801) IEEE J. Sel. Areas Commun. (2,087) IEEE Wirel. Commun. (1,830) IEEE Trans. Wirel. Commun. (1,814)	IEEE Trans. Veh. Technol. (89) ICC (88) GLOBECOM (73) IEEE Trans. Wirel. Commun. (65) WCNC (46)

Table 6.6 The top 5 venues for each of the top 5 most cited authors (top) and most productive authors (bottom) they got most cited in (left) and most published in (right). "IEEE Trans. Pattern Anal. Mach. Intell." is abbreviated with "IT. Pattern Anal. Mach. Intell."

Top 5 most cited authors						Top 5 most productive authors					
1974-1999	2000-2004	2005-2009	2010-2014	2015-2019	1974-2019	1977-2019	1977-1999	2000-2004	2005-2009	2010-2014	2015-2019
cluster	cluster	cluster	latent	*fingerprint*	imag	network	estim	data	**schedul**	*network*	**paper**
object	face	face	*fingerprint*	face	fingerprint	channel	**signal**	**cluster**	scheme	**code**	**simul**
invari	*recognit*	*fingerprint*	face	*match*	textur	problem	**handoff**	algorithm	relai	**graph**	result
textur	**featur**	biometr	*imag*	*object*	model	differ	**time**	queri	power	*channel*	*channel*
imag	biometr	minutia	*recognit*	latent	object	paper	**detect**	**index**	**sep**	mobil	problem
distanc	*fingerprint*	*imag*	*object*	segment	face	code	*network*	**pattern**	user	**social**	estim
curv	algorithm	algorithm	biometr	**human**	pose	data	**parallel**	**mine**	*channel*	**featur**	**present**
project	model	data	minutia	cluster	cluster	present	user	base	averag	**decod**	power
structur	data	*object*	segment	*imag*	motion	relai	cach	adapt	adapt	**spectrum**	*network*
match	document	**classif**	model	**task**	track	consid	algorithm	divers	combin	**sens**	**exist**
recognit	**person**	**deform**	user	pose	algorithm	model	**buffer**	**cdma**	divers	**predict**	**numer**
form	**select**	segment	*match*	**network**	descriptor	result	mobil	**equal**	**select**	**complex**	**compar**
camera	view	secur	templat	**text**	human	propos	**disk**	**system**	**propos**	**detector**	**noma**
fingerprint	*match*	shape	**qualiti**	**instanc**	learn	fade	**comput**	cach	data	error	**express**
segment	**facial**	*recognit*	retriev	shape	biometr	scheme	**multius**	*channel*	**transmit**	data	relai
descriptor	**writer**	**express**	detect	**box**	kernel	energi	**join**	**burst**	perform	**privaci**	**solut**
pose	system	model	**composit**	*recognit*	segment	express	queri	mobil	**spectral**	**cooper**	scheme
vision	*object*	system	**ag**	**spoof**	data	close	**quantiz**	**structur**	**effici**	**link**	**analyz**
estim	partit	templat	shape	**reader**	method	simul	**block**	*network*	distribut	**capac**	**demonstr**
view	templat	partit	secur	visual	plane	error	problem	perform	**transmiss**	**label**	**devic**
plane	user	**region**	**sketch**	detect	retriev	investig	**filter**	object	**fade**	**cognit**	**optic**
document	textur	pose	**orient**	identif	classif	user	**execut**	combin	**rate**	**aid**	**end**
determin	shape	retriev	**alter**	**depth**	match	select	*channel*	**wireless**	base	relai	**number**
algorithm	**ensembl**	**scene**	**dataset**	**map**	featur	develop	scheme	**probabl**	**antenna**	user	mobil
model	identif	**fusion**	**ridg**	**target**	train	numer	**processor**	scheme	**optim**	base	**term**
data	pattern	*match*	**method**	estim	correspond	decod	data	**segment**	**switch**	video	**achiev**
frame	**line**	**supervis**	learn	**transfer**	perform	order	web	averag	result	**applic**	video
pattern	learn	learn	**class**	**embed**	video	time	**node**	**gener**	**threshold**	power	**function**
motion	plane	**reconstruct**	**field**	**train**	bodi	power	**codec**	**interfer**	error	**radio**	**appli**
integr	*imag*	visual	**databas**	**cnn**	geometri	graph	**server**	web	*network*	object	distribut

Table 6.7 Top 30 most salient terms for the top 5 most cited and most productive authors in different time periods. Terms that only appear in one time period are **bold** and terms that appear in all five time periods are in *italics*.

find an ongoing increase for “neural network” and “Convolutional neural (CNN)”, and Xia et al. (2021) even show “Object Detection; CNN; IOU” is the #1 research frontier in CS, which matches more with our results. We assume the differences are due to the different approaches and underlying data.

For the most productive authors, the two most salient terms over the whole period “network” and “channel”, and the terms “code”, “data” and “relai” are related to the signal processing focus of their venues. There are also some general terms (e.g., “problem”, “present”, and “consid”) with high saliency, which is not the case for the terms of the most cited authors. The terms “network” and “channel” are also the two only terms, that appear in every period, but we believe the term “network” in this context is not related to neural networks and instead, physical networks considering the associated venues and other terms. We also see generally more unique terms per period compared to the most cited authors, e.g., “codec” before 2000, “wireless” in 2000-2004, and “privaci” in 2010-2014. The shift of topics to issues such as privacy, security, IoT, and big data, that Coşkun et al. (2019) find is thus only somewhat visible for our most cited and most productive authors.

6.4 Venues

This section covers experiments that aggregate the data in CS-Insights by the venues and then analyzes the venues to answer the remaining parts of RQ1, RQ2, and RQ3. We start with general trends for the number of venues and their number of publications and citations (Experiments 9 to 11), before covering the most cited and most productive venues, and what topics they cover (Experiments 12 to 14).

Experiment 9: How many venues publish in CS-Insights per year? (RQ1)

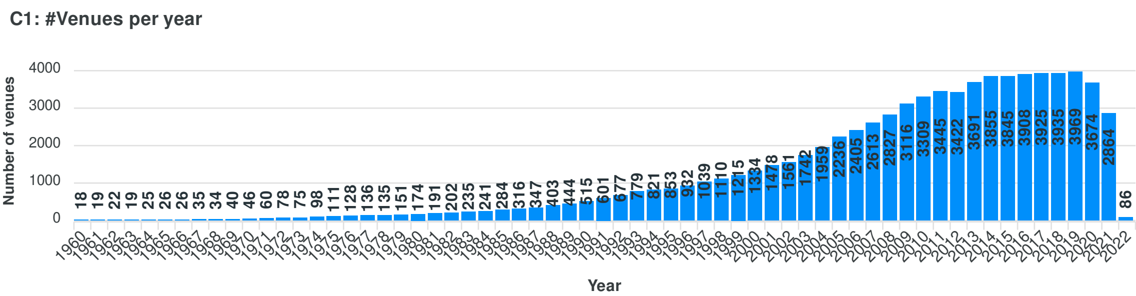


Figure 6.3 Number of unique venues per year starting in 1960. See Figure A.13 for the full span.

In total there are 14,268 venues in CS-Insights, while there are 14,212 publications without any venue listed. 12,683 publications without a venue (nearly 90%) are books, and another 1,131 are Ph.D. and master’s theses. For the number of venues we see an increase during the late 1980s and a large increase since the 2000s (**Figure 6.3**), similar to the publications (Experiment 1) and authors (Experiment 3). The number of venues seems to reach a plateau in 2014, unlike publications and authors, which show an even bigger increase since 2017. In 2020, the number of venues went down, which was likely caused by the COVID-19 pandemic and in-person events being canceled. This is also the reason why the number of publications goes down in 2020, as seen in Experiment 1.

Experiment 10: How many unique venues published in the last x years? How many venues are new? (RQ1)

Time span	**1 year**	**2 years**	**3 years**	**4 years**	**5 years**	**Total**
Years	2020	2019-2020	2018-2020	2017-2020	2016-2020	1936-2020
#Venues	3,674	4,670	5,338	5,890	6,456	14,044
%Venues	26.16%	33.25%	38.01%	41.94%	45.97%	100.00%
#Venues new	448	963	1,488	2,009	2,604	
%Venues new	3.19%	6.86%	10.60%	14.31%	18.54%	
Years (inv.)	1936-2019	1936-2018	1936-2017	1936-2016	1936-2015	
#Venues	13,596	13,081	12,556	12,035	11,440	

Table 6.8 Number of venues that were active (at least one publication) in the last x years compared to the total amount of venues. Also includes the number of new venues and their percentage considering the total number of venues (calculated using the inverted time span).

We show the number of venues that published in the last few years and how many are new, i.e., published for the first time (**Table 6.8**). Our goal is to investigate if new venues replace old venues or if the number of unique venues per year is actually stagnating in recent years (Experiment 9). We include the number of venues that were active before the last x years at the bottom (inverted time span), to determine the number of new venues by calculating the difference between venues in the inverted time span and total venues.

In the last five years, only 46% of all venues were active, while 40% of those were new venues. This is surprising, as we expected the bulk of venues to be recurring over the years, with some new ones every year. We think this has multiple reasons. First, up to 2020, there are 1,774 venues only for books, which only have one publication assigned nearly all the time. Second, some venues are listed on DBLP and in D3 with an increasing counter added to their name (e.g., “iConference (1)”, “TOOLS (48)”, “HCI (42)”), which causes a few more new venues and fewer older ones (see Section 7.2 for all limitations of our work). We try to remedy this by removing the counter with a script to fix the data from D3, but five venues with a total of around 100 occurrences are still in CS-Insights, so the total number of venues should be around 100 less. However, this does not explain the effect seen in this experiment, as the 1,774 book venues and 100 duplicate venues are spread over 1936-2020 and thus not enough to have an effect this large. We conclude the experiment shows an actual trend, that is not just caused by issues in the data. There are new venues in the last few years even though Experiment 9 shows a plateau in the number of unique venues per year, which implies the new venues compensate for an apparent loss of older venues. The number of new venues is smaller compared to the number of new authors (Experiment 4) in the last few years, which means the number of authors is faster growing than the number of venues. As mentioned in Experiment 1, in NLP the number of publications goes down every second year due to biennial conferences (Mohammad 2020b; Mariani et al. 2019a), but we can still not replicate the same trend in CS-Insights.

Experiment 11: How are the citations and papers distributed across venues? How do the distributions change over time? (RQ2)

Time span	#Citations					#Papers				
	Q1	Med.	Q3	Max.	Avg.	Q1	Med.	Q3	Max.	Avg.
1960-1969	0	111	1,699	60,481	4,950.42	3	41	231	2,179	171.26
1970-1979	0	75	851	157,404	4,128.27	15	36	138	2,133	140.22
1980-1989	0	117	1,125	184,718	4,269.73	15	35	136	5,276	137.43
1990-1999	1	296	2,319	318,304	5,023.76	18	44	156	8,821	160.09
2000-2004	47	711	3,200	237,082	4,947.81	19	59	155	5,716	145.40
2005-2009	92	553	2,819	479,768	4,725.48	16	51	169	6,881	167.19
2010-2014	91	419	1,797	492,469	3,549.21	15	54	166	7,724	170.56
2015-2019	49	243	1,094	722,350	2,523.54	17	66	195	25,371	211.79
1960-2019	37	239	1,494	1,609,420	7,029.40	14	33	193	42,071	318.30

Table 6.9 Distribution of the number of total citations and papers across venues per time period showing the first quartile, median, third quartile, maximum, and average. The upper block covers 10 years per time period and the lower block five years.

We see both the distribution of citations and papers have a higher average than the respective median for every period (**Table 6.9**), which implies that most citations and papers fall to a few venues and are not spread evenly. For the citations, the first quartile peaks in 2005-2009, which is also the period with the most citations (Experiment 15) and lines up with the authors (Experiment 5), while the median and third quartile peak in 2000-2004. We explain the difference in 2000-2004 with the distribution across venues being the most even (highest median over average ratio of all periods), and also 2005-2009 showing the highest increase in venues per year (3,000 unique venues in 2000-2004 to nearly 5,000 in 2005-2009), which causes the citations to split up more. The large drop-off in citations for the first quartile before 2000 also matches the citations of the authors (Experiment 5). If we consider the total period, half of all venues receive between 37 and 1494 citations in total. The maximum steadily increases for both citations and papers, showing some venues are publishing more papers every period and some also receiving more citations. This again matches the authors, who show a slight trend upward of the maximum of citations and a steady increase in publications (Experiment 5). The number of papers reveals that most venues publish only slightly more on average over the decades and most venues do not show the same increase as the venues that are responsible for the maximums. Overall we see a correlation between overall citations, authors, and venues and a small correlation between the number of papers for authors and venues.

Experiment 12: What are the most cited and most productive venues? (RQ3)

#	Venue (#Citations)	First	#Papers	#Citations	Avg. Citations
1	CVPR	1988	12,757	1,621,492	127.11
2	*Others*	1938	14,212	1,505,675	105.94
3	NeuroImage	1996	16,947	1,377,202	81.27
4	IT. Pattern Anal. Mach. Intell.	1975	6,559	1,337,060	203.85
5	IEEE Trans. Inf. Theory	1963	16,325	1,147,862	70.31
6	Commun. ACM	1958	12,742	948,274	74.42
7	IEEE Trans. Ind. Electron.	1990	12,777	802,571	62.81
8	ICRA	1984	25,017	790,170	31.59
9	IEEE Trans. Signal Process.	1990	13,328	762,802	57.23
10	IEEE Trans. Autom. Control.	1991	10,762	717,910	66.71
	Average	1982	14,143	1,101,102	88.12
#	**Venue (#Papers)**	**First**	**#Papers**	**#Citations**	**Avg. Citations**
1	IEEE Access	2013	54,961	452,363	8.23
2	ICASSP	1975	45,660	714,945	15.66
3	Sensors	2009	36,718	526,251	14.33
4	IGARSS	2002	29,421	119,000	4.04
5	ICRA	1984	25,017	790,170	31.59
6	ISCAS	1993	23,549	174,126	7.39
7	ICIP	1993	22,714	330,302	14.54
8	ICC	1984	22,296	303,031	13.59
9	Appl. Math. Comput.	1998	19,983	336,211	16.82
10	IROS	1988	19,561	422,430	21.60
	Average	1994	29,988	416,883	14.78

Table 6.10 Top 10 venues based on the number of citations received (top) and publications (bottom). The average is computed excluding *Others*. "IEEE Trans. Pattern Anal. Mach. Intell." is abbreviated with "IT. Pattern Anal. Mach. Intell."

Both top 10s have barely any overlap, except for the ICRA (IEEE Robotics and Automation Society) conference, which also has the highest average citations of the top 10 most productive venues, but the lowest of the top 10 most cited venues (**Table 6.10**). Some venues do appear in the other's top 20 though, e.g., NeuroImage at #14 and IEEE Trans. Inf. Theory at #16 in the top 20 venues with the most publications. In the top 20 of the most cited venues ICASSP (IEEE International Conference on Acoustics, Speech and Signal Processing) appears at #11, and Sensors at #19. The most cited venues include many venues we already saw as venues the most productive and most cited authors publish in (Experiment 7), e.g., CVPR, IEEE Trans. Pattern Anal. Mach. Intell., and IEEE Trans. Inf. Theory. This implies the topics of the most cited venues also align with computer vision, pattern recognition, signal processing, and communication. We investigate the topics further in Experiment 13. In a later experiment, we see engineering has a high preference for conferences, which explains why many of the most productive venues in this experiment are IEEE conferences with a focus on topics from the field of engineering (Experiment 26).

The most cited venues are dominated by journals, but the first place is taken by the CVPR conference. On the contrary, the most productive venues are mostly conferences, while the first and third places are taken by the open-access journals IEEE Access and Sensors, respectively. In general, the average citations are also lower for most productive venues compared to the most cited venues, and IEEE Trans. Pattern Anal. Mach. Intell. has the highest average citations (203.85) of all covered venues. When we compare the list to the most productive venues of Fiala and Tutoky (2017) we recognize some venues, e.g., IEEE Transactions on Information Theory, and Communications of the ACM. Most venues in their list appear to be journals, which is different from our list, where it is mostly conferences. Coşkun et al. (2019) only investigate the top 10 journals for two periods (2008-2013 and 2014-2019). We again see a little overlap, e.g., IEEE Access is #1 in the second period, and IEEE Transactions On Information Theory also appears. Their most productive journals are dominated by IEEE and IEICE journals, which we do not see in our most productive venues, but instead, we see five IEEE journals in our 10 most cited venues. The #1 of both works has fewer papers than our #10, and they use a different and smaller dataset (from Web of Science) which can explain the differences.

An experiment for the future would be to look into other measures of quality (e.g., impact factor) for these most prominent venues, considering the most cited venues are, on average, 12 years older and thus more established and possibly more prestigious. Another interesting experiment for future research is to look into open-access publications only, as both lists change drastically, with IEEE Access and Sensors leading the most productive venues and them being #3 and #1 respectively for the most cited venues, where they do not appear in the top 10 or 20 before at all.

Experiment 13: What are the most popular topics of the most cited and most productive venues? How do the venues' topics differ from each other? (RQ3)

While there are no terms that appear in all five of the most cited venues, we can use the unique terms (bold) to see how the venues differentiate from each other and what makes them unique (**Table 6.11**). For CVPR it is "camera", "deep", "detect" (computer vision), for NeuroImage "cortex", "brain", and "stimuli" (brain imagery), and for IEEE Trans. Pattern Anal. Mach. Intell. "face", "cluster", and "track" (facial recognition). IEEE Trans. Inf. Theory is more related to signal processing and communication, which is visible due to "code", "channel", "decode", and "signal",

Top 5 most cited venues					Top 5 most productive venues				
CVPR	NeuroImage	IT. PA. M. Int.	IT. Inf. Theory	Commun. ACM	IEEE Access	ICASSP	Sensors	IGARSS	ICRA
propos	**respons**	propos	**code**	algorithm	*propos*	*propos*	*propos*	**sar**	*propos*
result	**connect**	paper	**channel**	**program**	result	**speech**	sensor	*imag*	control
object	**ag**	result	network	**time**	control	*imag*	network	*propos*	*present*
network	data	**face**	**sequenc**	**softwar**	network	paper	measur	**surfac**	**object**
present	method	model	**decod**	develop	*imag*	model	paper	**area**	result
imag	function	learn	present	problem	data	result	*present*	method	**manipul**
method	imag	**recognit**	**capac**	**new**	*present*	*present*	provid	**radar**	learn
state	**suggest**	**cluster**	paper	provid	paper	network	*imag*	model	**robot**
motion	network	**label**	estim	**languag**	learn	estim	data	**classif**	paper
art	**activ**	object	**signal**	**need**	detect	**filter**	method	**soil**	estim
camera	process	train	algorithm	**inform**	model	**recognit**	featur	**land**	**task**
problem	model	work	**nois**	**technolog**	**power**	algorithm	work	*present*	*imag*
dataset	**left**	network	problem	**system**	**differ**	**nois**	**posit**	**chang**	**plan**
learn	**associ**	imag	**set**	**includ**	featur	**rate**	achiev	featur	design
featur	**cortex**	develop	**studi**	**number**	method	signal	detect	data	**grasp**
challeng	**visual**	achiev	**obtain**	point	algorithm	achiev	**monitor**	perform	provid
gener	result	surfac	**sourc**	**acm**	provid	problem	energi	**resolut**	problem
experi	**right**	**track**	function	**commun**	obtain	**train**	result	measur	sensor
segment	**particip**	import	**construct**	model	**comput**	**code**	node	**forest**	**map**
us	**diffus**	point	**perform**	**follow**	**predict**	provid	obtain	**sens**	time
deep	provid	motion	**sub**	**design**	achiev	**speaker**	**requir**	**cover**	**motion**
surfac	**brain**	segment	**user**	**servic**	energi	channel	**optic**	**retriev**	perform
demonstr	**area**	requir	given	**internet**	design	learn	**sensit**	**remot**	**actuat**
detect	**show**	**match**	**rate**	data	time	featur	estim	work	data
data	**matter**	**deriv**	process	paper	**commun**	**demonstr**	learn	**spectral**	**dynam**
train	state	term	achiev	**discuss**	signal	**frequenc**	temperatur	**inform**	**mechan**
requir	**eeg**	view	**scheme**	import	channel	**consid**	signal	**moistur**	**forc**
point	**stimuli**	problem	**receiv**	given	**user**	work	accuraci	time	**real**
view	**stimul**	**recent**	**distort**	**possibl**	accuraci	**transform**	**wireless**	temperatur	**camera**
work	**region**	estim	term	**describ**	node	**experiment**	**develop**	**polarimetr**	**path**

Table 6.11 Top 30 most salient terms for the top 5 most cited and most productive venues. Terms that only appear in one venue are **bold** and terms that appear in all five venues are in *italics*. "IEEE Trans. Pattern Anal. Mach. Intell." is abbreviated with "IT. PA. M. Int." and "IEEE Trans. Inf. Theory" with "IT. Inf. Theory"

which makes it also overlap with the top 5 most productive authors, who focus on the same area. The Commun. ACM is a broader venue about current trends in CS, which can be somewhat seen through its unique terms "program", "softwar", "inform", "technolog", and "internet". We see that for three of the five venues, the focus on computer vision and pattern recognition becomes visible and their specialization in that field. Other topics are present in the other two venues (i.e., information theory and current trends respectively), so there is not just one clear direction for the topics.

The top 5 most productive venues have three words that overlap, "propos", "imag", and "present". IEEE Access is a multidisciplinary open-access journal, which also shows in the data as it has only a few unspecific unique terms, but the other four venues all have unique terms, that describe the topic of the respective venue well. ICASSP (International Conference on Acoustics, Speech and Signal Processing) has "speech", "filter", and "noise", Sensors has "optic", "sensit", and "wireless", IGARSS (International Geoscience and Remote Sensing Symposium) has "surface", "area", and "radar", and lastly ICRA (International Conference on Robotics and Automation) has "object", "robot", and "motion". We again see the specialization of each venue, and additionally a general focus on engineering topics, or more specifically, topics that heavily use sensors in four of the five venues, the exception being IEEE Access.

Experiment 14: Do the topics of the most prominent venues change over time? (RQ3)

In **Table 6.12** the first period includes all publications before 2000 to reduce issues with data sparsity in the earlier years, similar to Fiala and Tutoky (2017), and starts with the year of the first publication, i.e., 1958 and 1975. We group the venues to try and find trends over time that overlap between the most successful venues, e.g., usage of specific technologies, even if the specific venues might have a different focus. The idea is venue specific terms should be ranked lower, and (common) terms that exist across all venues are ranked higher.

For the most cited venues, the terms "model" and "imag" appear in each period, which is probably due to three of the five venues focusing on computer vision and pattern recognition. The term "brain" is ranked high starting in 2005, which we attribute to NeuroImage being exclusively about brain imagery and first starting publication in 1996. However, in 2015-2019 we also see "dataset", "learn", and "train" appearing for the first time, and "network" appearing again, which indicates a shift to methods leveraging neural networks across venues, similar to the most cited authors in that period (Experiment 8). In that experiment, we already related the findings from other researchers regarding neural networks, so we refrain from repeating them. The most productive venues share multiple terms across all periods. Some are general terms ("present", "propos", "provid"), and some were venue specific in Experiment 13 ("speech", "robot"), similar to "brain" for the most cited venues, which leaves "signal", "imag", "control", and "model". These last four terms indicate a general focus more toward engineering for the most productive venues, which matches the findings of the last experiment (Experiment 13). Again, 2015-2019 indicates a shift toward

Top 5 most cited venues							**Top 5 most productive venues**				
1958-1999	2000-2004	2005-2009	2010-2014	2015-2019	1958-2019	1975-2019	1975-1999	2000-2004	2005-2009	2010-2014	2015-2019
code	code	brain	*imag*	propos	code	paper	*robot*	*propos*	paper	*propos*	*propos*
algorithm	activ	activ	brain	**art**	algorithm	present	*present*	paper	*propos*	paper	*imag*
present	**spl**	*imag*	propos	brain	present	speech	filter	*present*	*robot*	*model*	paper
paper	**sub**	paper	network	**dataset**	paper	result	*control*	*robot*	*imag*	*robot*	network
imag	*imag*	object	activ	compar	imag	signal	estim	data	*present*	result	result
develop	area	present	**cortic**	network	develop	estim	*propos*	*model*	sensor	*present*	**demonstr**
program	*model*	result	data	*imag*	program	propos	*speech*	*imag*	data	*imag*	method
model	investig	subject	present	method	model	network	**adapt**	result	compar	achiev	*robot*
error	compar	code	area	paper	error	time	algorithm	**develop**	*control*	*control*	**optim**
estim	respons	algorithm	subject	**function**	estim	problem	**manipul**	*speech*	sar	*speech*	*control*
user	task	experi	task	studi	user	achiev	*model*	*control*	*speech*	estim	**learn**
object	develop	channel	respons	task	object	algorithm	*signal*	sar	filter	problem	*present*
possibl	studi	*model*	method	present	possibl	featur	**recognit**	compar	featur	nois	featur
softwar	user	cortex	paper	**learn**	softwar	model	**code**	*provid*	**us**	data	data
languag	problem	studi	*model*	**train**	languag	provid	object	obtain	work	work	algorithm
inform	bound	increas	**demonstr**	**end**	inform	vector	task	radar	network	algorithm	*provid*
author	data	task	cortex	differ	author	control	problem	network	*model*	*signal*	problem
result	work	specif	state	object	result	recognit	**motion**	**area**	algorithm	*provid*	*speech*
work	cortex	develop	develop	**visual**	work	imag	**rate**	channel	method	featur	estim
nois	**sup**	obtain	**show**	includ	nois	nois	**describ**	achiev	**number**	**error**	**train**
probabl	**motion**	compar	investig	data	probabl	error	**process**	filter	**surfac**	filter	*model*
rate	experi	differ	**connect**	work	rate	method	*imag*	differ	**resolut**	**simul**	obtain
sourc	**signific**	**us**	studi	featur	sourc	demonstr	*provid*	object	achiev	sensor	**term**
signal	error	problem	obtain	state	signal	channel	**forc**	*signal*	*provid*	channel	work
term	channel	area	**provid**	**domain**	term	filter	sensor	**land**	*signal*	**shown**	**time**
problem	**relat**	**matter**	increas	**control**	problem	word	nois	**remot**	**requir**	method	**perform**
bound	subject	featur	channel	code	bound	train	**design**	featur	problem	**speaker**	comput
number	specif	**effect**	**patient**	area	number	qualiti	obtain	**sens**	radar	comput	**state**
includ	**chang**	**fmri**	**memori**	*model*	includ	requir	**transform**	**band**	nois	task	**commun**
requir	differ	**region**	**segment**	activ	requir	data	**joint**	**satellit**	differ	**oper**	*signal*

Table 6.12 Top 30 most salient terms for the top 5 most cited and most productive venues in different time periods. Terms that only appear in one time period are **bold** and terms that appear in all five time periods are in *italics*.

approaches using neural networks because “learn”, “train”, and “optim” are new terms, and “network” also appears again on a high rank.

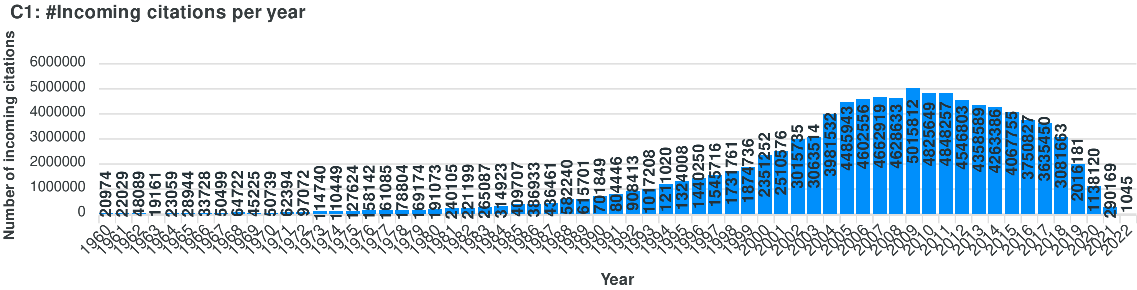
We conclude that this experiment (i.e., grouping venues for overarching topics) only had partly the effect we had hoped for. Generic words are now more present (e.g., “present”, “paper”, and “propos”), which we hoped the saliency measure would prevent (as it takes distinctiveness into account), but also specific technologies are not ranked higher in most cases, or there were simply no overlaps across venues as we hoped for. Especially the two full periods show no clear direction compared to earlier experiments (Experiments 8 and 13). Additionally, we see terms that formerly only related to one specific venue are still being ranked very high, e.g., “brain”, “robot”, and “speech”. It is possible these terms also have some importance in the other venues and thus are ranked high up, but we believe it is more likely, that the measure of saliency causes these issues. Saliency measures the distinctiveness of words considering all topics, and in this case, the topic model correctly sorts all publications with, e.g., “brain” into one topic which gives it a high saliency, so the problem appears to be that the topics model can perfectly sort five venues into 10 different topics. We hope to overcome these issues and eliminate the venue-specific terms in a better way, by grouping more venues together, which we will do when we investigate the topics of the different document types (Experiment 24). Another possibility that CS-Insights supports is to analyze each venue separately, but our goal is to analyze the state of CS, so we have to move this to possible future work (Section 7.2).

6.5 Citations

This section exclusively deals with experiments conducted directly on the citations of publications in CS-Insights without any further aggregations to answer RQ4. We cover both incoming and outgoing citations, including their numbers per year (Experiment 15) and their distributions (Experiments 16 to 17).

Experiment 15: How do incoming and outgoing citations compare over time? (RQ4)

In total CS-Insights contains 97,053,288 incoming citations and 88,302,512 outgoing citations. The incoming citations that publications receive have a peak in 2009 and consistently fall off in earlier and later years (**Figure 6.4**), which we explain with earlier years having fewer publications and thus fewer citations overall, and the publications in later years not being old enough to aggregate enough citations yet. This is supported by Fiala and Tutoky (2017) who show new citations become fewer every year after publication, which explains why older publications have fewer citations. We also explain this with researchers focusing more on current work when citing other publications. The drop-off for newer publications comes from them just not having accumulated so many citations yet, as Fiala and Tutoky (2017) also show that many citations still come in years after publication. We believe this causes a certain point



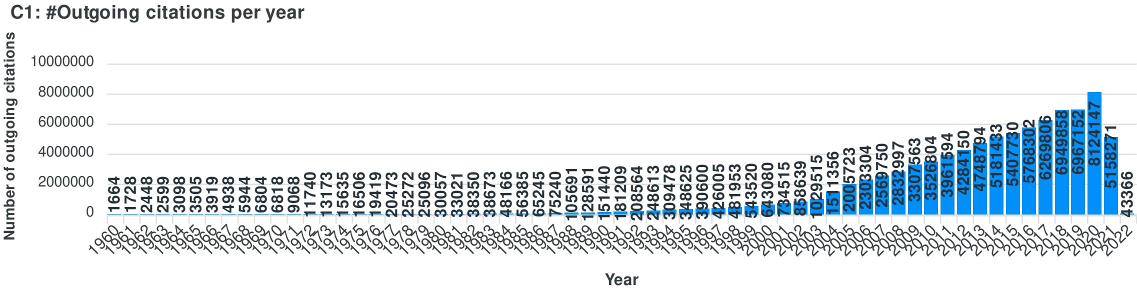


Figure 6.4 Incoming citations per year (top), that publications from that year received in their lifetime, and outgoing citations (references) per year (bottom), that publications from that year listed in their bibliography; starting in 1960. See Figures A.14 and A.15 for the full span.

(i.e., the peak around 2009), where the aforementioned effects related to the drop-offs balance each other out. We already observed a similar trend when investigating the authors (Experiment 5) and explain this with the general trend for authors simply following the general trend of the overall citations. The trend of citations for venues appears different, as some peaks are during earlier periods, which we already covered in Experiment 11. Mohammad (2020b) sees a similar curve for incoming citations, even though 2009 is not the peak, as the curve is more susceptible to irregularities caused by singular highly cited publications, due to the smaller dataset size and fewer publications per year. We can thus infer, that the peak of received (i.e., incoming) citations always is a few years back in time, even though we cite recent papers the most (Fiala and Tutoky 2017) and produce more citations every year (see next paragraph).

For the outgoing citations (references) we observe a consistent increase, that follows similar trends as the number of publications per year, with the first larger increase in the late 1980s and a second large increase since the 2000s (Experiment 1). The consistent increase in outgoing citations shows us we are citing more other publications with each passing year. Whether this is solely due to the increase in publications (Experiment 1) or if we are also citing more publications per paper is investigated in Experiment 17.

In Section 4.1.3 we explained the number of citations is derived from linking citations between publications in our dataset, so each incoming citation should have one matching outgoing citation, and thus, the total number of incoming and outgoing citations should be equal. We believe the total numbers do not match in this experiment, as the numbers were calculated and saved individually for each publication before the export of D3, which possibly lost some publications (Section 7.2). This issue should be fixed in the new version of D3 which was not yet available when conducting the experiments.

Experiment 16: How are the publications distributed based on the number of incoming citations? (RQ4)

#Citations	0	1-9	10-99	100-999	1000+	Total
#Publications	1,441,029	1,909,897	1,372,511	164,601	5,502	4,893,540
%Publications	29.45%	39.03%	28.05%	3.36%	0.11%	100.00%

Table 6.13 Incoming citations sorted into citation bins.

We choose the same binning sizes as Mohammad (2020b) and as a result, sort the 128 publications with 10,000 citations or more into our largest bin (1000+) (**Table 6.13**).

Our citations are heavily skewed to the bins with fewer citations. 29% of all publications in CS-Insights never receive any citations, 39% receive 1-9, and only close to a third of all publications receive 10 or more citations. Mariani et al. (2019a) also see 44% of publications are never cited in NLP and Fiala and Tutoky (2017) see 52% are never cited in CS using Web of Science data. Only Mohammad (2020a) has just 6% of publications with 0 citations, but 48% with 10-99 citations when investigating NLP publications. This is quite unusual, as Fiala and Tutoky (2017) state, that it is a well-known fact in scientometrics, that most papers remain without citations. We believe this large difference in the distribution of citations is due to different datasets (DBLP vs. ACL Anthology), and different ways of obtaining the citation counts (matching citations in the corpus itself vs. Google Scholar).

Experiment 17: How do the distributions of incoming and outgoing citations change over time? (RQ4)

We observe an increase in the median and third quartile of incoming citations until 2005-2009, after which they fall off again (**Table 6.14**), which matches the overall trends of incoming citations per year, as that period also has the most citations overall (Experiment 15). The average number of incoming citations slightly increases between 1960 and 2004, and peaks in 2000-2004, before dropping off, due to newer papers being less cited and the increase in papers since the 2000s (Experiment 1), which drags the average down. This could also explain why the peak of the average is in 2000-2004 and not 2005-2009. Mariani et al. (2019a) see a similar distribution of incoming citations, except they observe a spike in the 1970s and a larger fall-off toward the 1960s, which we explain with only a small number of NLP papers existing during that time. The maximum for both incoming and outgoing citations is lower in the first periods and fluctuates afterward. Outgoing citations seem to increase with every period, with

Time span	Incoming Citations					Outgoing Citations				
	Q1	Med.	Q3	Max.	Avg.	Q1	Med.	Q3	Max.	Avg.
1960-1969	0	1	10	10,747	28.91	0	0	4	509	2.97
1970-1979	0	1	9	18,702	29.44	0	0	6	376	3.91
1980-1989	0	1	12	57,583	31.07	0	0	8	885	5.25
1990-1999	0	2	17	45,635	31.38	0	4	13	2,665	8.22
2000-2004	0	5	22	38,124	34.03	0	8	16	1,348	10.89
2005-2009	1	7	23	37,926	28.26	5	12	22	1,313	15.73
2010-2014	1	6	18	39,111	20.81	8	16	27	4,627	19.77
2015-2019	1	3	10	37,732	11.92	8	18	31	1,292	22.58
1960-2019	0	4	16	57,583	22.08	4	13	24	4,627	17.33

Table 6.14 Distribution of the number of total incoming and outgoing citations per time period showing the first quartile, median, third quartile, maximum, and average.

the respect to the first quartile, median, third quartile, and average which means not only are there more outgoing citations in total (Experiment 15) but also on average, each publication cites more other publications. The same trend of increasing outgoing citations per paper is observed by Mariani et al. (2019a) in NLP.

6.6 Document Types

This section covers experiments that answer RQ5 by investigating the document types and differences between conferences and journals in CS-Insights. We start with an overview of the distribution of document types (Experiment 18), by aggregating CS-Insights's data by document type. The experiments thereafter use no aggregation and instead leverage the *Types of papers* filter to find differences between conferences and journals regarding their trends over time (Experiments 19 to 21), and the most cited and productive authors, most cited venues, and topics (Experiments 22 to 24). We refer to the document types in CS-Insights's UI with "types of paper", and in this section refer to publications of conferences with "papers" and publications in journals with "articles".

Experiment 18: How are the publications distributed across the document types? (RQ5)

The document types are based on the BibTeX types of publications (**Table 6.15**).

We see just over half of all publications CS-Insights are conference papers ("inproceedings"), 45% are journal articles, and together they make up around 98% of our dataset. Thus, CS-Insights is more evenly distributed concerning conference papers and journal articles compared to Fiala and Tutoky (2017) with 56.1% vs. 34.8% (an additional 8.7% are classified as "Article; Paper"), and Coşkun et al. (2019) with 59.75% vs. 38.18% (second period only, the first one is even less evenly distributed),

Document Type	First	#Publications	#Citations	Avg. Citations
inproceedings	1951	2,574,226 (52.60%)	35,420,550 (36.50%)	13.76
article	1936	2,210,231 (45.17%)	58,782,837 (60.57%)	26.60
incollection	1941	64,936 (1.33%)	910,658 (0.94%)	14.02
proceedings	1951	28,408 (0.58%)	334,822 (0.34%)	11.79
book	1949	14,563 (0.30%)	1,602,926 (1.65%)	110.07
phdthesis	1938	1,171 (0.02%)	1,495 (0.00%)	1.28
mastersthesis	1984	5 (0.00%)	0 (0.00%)	0.00

Table 6.15 Distribution of publications and citations across document types.

both in favor of conference papers. Even though there are fewer journal articles, they make up 60% of the total citations and receive on average twice as many citations as conference papers, while books receive on average by far the most citations (110), but they also only make up 0.3% of CS-Insights. Fiala and Tutoky (2017) also show a disbalance in citations, as journals contribute 75.6% of all citations and conference papers only 10.7% (an additional 10.9% are for the type "Article; Paper"). Franceschet (2010) also finds in general journal articles receive more citations, while there are more publications in conferences, which Mohammad (2020a) also finds for NLP, as only 2.5% of the publications in NLP Scholar are journals, but journals have the highest median and average citations of all document types. On the other hand, Vrettas and Sanderson (2015) find that disregarding the quality of the venue, there are few differences between conferences and journals regarding citations. We assume the differences are due to the different data sources (DBLP vs. Microsoft Academic Search), which is the same reason Vrettas and Sanderson (2015) themselves state regarding contradicting results of previous research in this area. One might argue the reason for the lower citations of conference papers in CS-Insights might be the way workshop papers are handled, as Mohammad (2020b) shows they make up the bulk of the papers and are classified as conference papers in CS-Insights. However, Mohammad (2020a) shows that workshop papers still have a higher citation average and median compared to non-top-tier conferences. This means workshop papers are not dragging down the citations for conferences in NLP and rather low-quality conferences, which we assume is also the case for CS-Insights.

Experiment 19: How do the number of publications per year differ for conferences and journals? (RQ5)

Journals appear in CS-Insights since the start of our dataset in 1936, while the first conference was in 1951 (AIEE-IRE Computer Conference) (**Figure 6.5**). Both journals and conferences have published at least one document each year since then, but journals published more each year until 1993, which was the first year conferences published consistently more than journals (17,634 papers vs. 15,982 articles). This trend continues until 2020 when journals published 168,263 articles, but conferences only 137,654 papers. 2020 was also the first year the number of conference papers

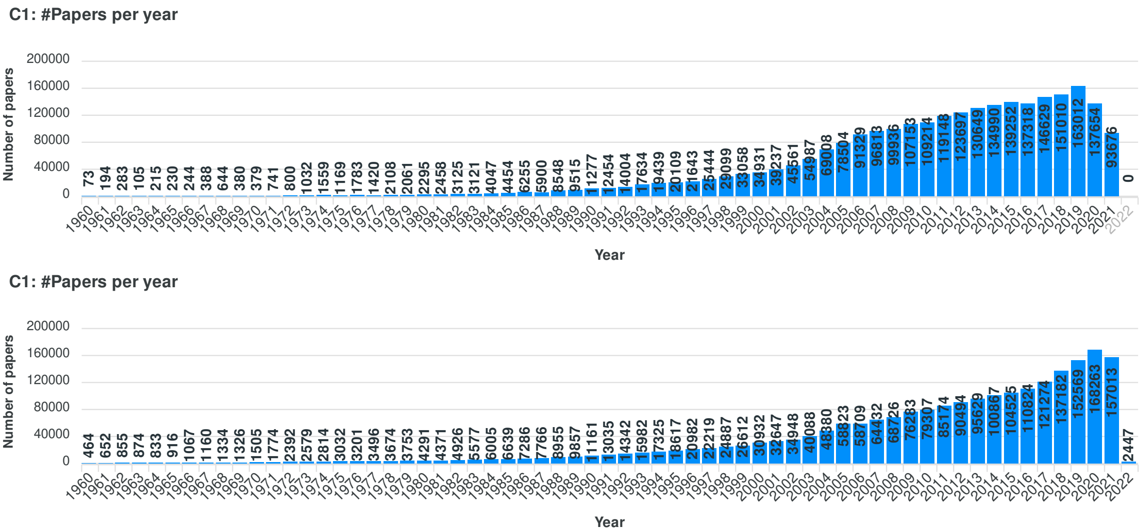


Figure 6.5 Number of publications per year in conferences (top) and journals (bottom). See Figures A.16 and A.17 for the full span.

per year went down compared to the previous year, which is likely linked to the COVID-19 pandemic and in-person events being canceled. Journals, on the other hand, see another increase of 15,000 publications compared to 2019, and their number of publications appears unaffected by the pandemic based on the graph. We can now link this insight to Experiment 1, where we saw a drop in the overall number of publications in 2020, and know this drop is solely caused by conference papers and the increase in journal articles could not compensate.

In general, conferences saw the first spurt in publications in the late 1980s, a second one in the early-mid 2000s, and another one around 2017. The publications per year in journals grow at a more steady rate, but a small spurt during the early-mid 2000s and a larger one around 2017 are also visible. Similar trends and bursts are visible for authors (Experiment 3) and publications (Experiment 1), which show there are general trends visible for both conferences and journals (except for the effect of COVID-19), which makes sense, as more authors mean more publications overall and thus also more publications for conferences and journals. In a later experiment, we will investigate how the number of unique conferences and journals affects the number of publications (Experiment 21). Fiala and Tutoky (2017) also investigate the number of articles and papers per year in Web of Science and find similar spikes in the early 2000s, but their data shows a drop for both conferences and journals soon after. For journals, this is related to the "Lecture Notes in Computer Science" and "Lecture Notes in Artificial Intelligence" being classified differently starting in 2007 and conferences being indexed less from 2008 onward. Conferences are also indexed less before the 1990s in Web of Science, as there are nearly zero indexed publications before 1985, while our data shows a more natural increase in conference publications over the decades. Coşkun et al. (2019) see a similar shift of researchers publishing more in journals, as the disbalance of document types (conferences vs. journals) evened out when comparing 2008-2013 (73% vs. 26%) to 2014-2019 (60% vs. 38%).

Vardi (2009) attributes the preference of researchers in CS for conferences to a best practices memo from the Computing Research Association in 1999, but we show that conference papers overtook journal articles already in 1993 and that the increase in conference papers started in the late 1980s. Similarly, the increase in conference publications in the early 2000s should also not be linked to the memo, as journal papers also saw an increase during the early 2000s. We believe it is possible the memo was merely coincidental or a result of the already existing trend.

Experiment 20: How do the distributions of citations differ for conferences and journals? How do they change over time? (RQ5)

Time span	Conferences #Citations					Journals #Citations				
	Q1	Med.	Q3	Max.	Avg.	Q1	Med.	Q3	Max.	Avg.
1960-1969	0	2	8	4,207	16.80	0	1	11	10,474	31.75
1970-1979	0	1	5	6,582	14.57	0	1	13	18,702	34.89
1980-1989	0	1	10	13,895	20.39	0	0	15	27,195	35.26
1990-1999	0	3	15	26,281	23.71	0	1	21	45,635	38.16
2000-2004	0	5	18	25,843	25.61	0	4	30	30,893	44.10
2005-2009	1	5	15	28,322	18.09	2	13	40	30,815	43.40
2010-2014	1	4	11	16,712	13.28	3	11	31	39,111	31.28
2015-2019	0	2	6	28,076	8.14	1	6	17	37,732	16.51
1960-2019	0	3	11	28,322	14.99	0	7	25	45,635	30.62

Table 6.16 Distribution of the total number of citations for conferences and journals per time period showing the first quartile, median, third quartile, maximum, and average.

Generally, both conferences and journals follow similar trends, with the difference of journals having more citations (**Table 6.16**), which shows the overall trend of journals getting more citations (Experiment 18) is not new and already existed for a few decades. The average peaks in 2000-2004 and the median in 2005-2009 for both conferences and journals, and both fall off before and after, which we also saw for the general trend of citations in Experiment 17. Again, the same reasons for the trends of the overall citations from Experiment 17 apply here, so we do not explain the trends again.

Experiment 21: How does the number of venues influence the number of publications and citations? (RQ5)

Time span	Conferences #Venues				
	#Venues	#Papers	#Citations	Avg. Papers	Avg. Citations
1960-1969	18	2,756	46,289	153.11	2,571.61
1970-1979	151	13,052	190,218	86.44	1,259.72
1980-1989	503	49,718	1,013,942	98.84	2,015.79
1990-1999	1,571	204,161	4,840,908	129.96	3,081.42
2000-2004	1,850	243,724	6,242,358	131.74	3,374.25
2005-2009	2,826	473,735	8,571,443	167.63	3,033.07
2010-2014	3,768	617,698	8,204,522	163.93	2,177.42
2015-2019	4,204	737,221	6,001,438	175.36	1,427.55
1960-2019	8,637	2,342,065	35,111,118	271.17	4,065.20
Time span	**Journals #Venues**				
	#Venues	#Articles	#Citations	Avg. Articles	Avg. Citations
1960-1969	45	9,481	301,017	210.69	6,689.27
1970-1979	121	28,220	984,559	233.22	8,136.85
1980-1989	263	65,673	2,315,765	249.71	8,805.19
1990-1999	607	185,162	7,066,360	305.04	11,641.45
2000-2004	807	186,995	8,245,893	231.72	10,217.96
2005-2009	1,178	326,973	14,190,957	277.57	12,046.65
2010-2014	1,428	451,471	14,122,256	316.16	9,889.54
2015-2019	1,514	626,374	10,342,030	413.72	6,830.93
1960-2019	1,775	1,880,349	57,568,837	1,059.35	32,433.15

Table 6.17 Number of publications and citations in relation to the number of venues for conferences (top) and journals (bottom).

Our goal is to investigate if the rise in publications is caused by venues publishing more or new venues emerging, which Šubelj and Fiala (2017) also investigate for journals in CS and physics, but with a different approach.

We observe, there has been a constant and noticeable increase in both conferences and journals (**Table 6.17**). There were less than 100 venues total in 1960-1969 and over 4000 conferences and 1500 journals in 2015-2019. At the same time, the average number of publications per venue per period has roughly doubled (ignoring the outlier for conferences in 1960-1969), even though the length of the period has halved. We consider redoing this experiment in the future with a constant length of periods (Section 7.2) to be able to compare the numbers from earlier periods to the later ones with different lengths more easily.

Each journal receives on average more citations in the same period compared to conferences, and again, journals also receive more citations in total than conferences. Yet, journals publish less in total (during 1990-2019, as discussed in Experiment 19), but due to there being far fewer journals, they publish more than conferences on average. We think this is due to the nature of journals, which publish multiple issues yearly, while most conferences are only held once a year or less. Additionally, 85% of all journals published at least one issue in 2015-2019, while only half of all conferences were still held in 2015-2019, which we attribute to journals staying the same, while conferences are more fast-paced.

Šubelj and Fiala (2017) attribute the rise in CS journal articles to new journals, and not journals publishing more, which might seem contradictory to our findings, but they argue based on a comparison to journals in physics. They find the number of CS journals increases significantly (from ≈50 in 1975 to almost 450 in 2010), but in physics, the number of journals barely increases (from 100 to 150 in 35 years). The increase in the number of journals is certainly partly responsible for the rise in publications, but we would add to Šubelj and Fiala (2017)'s findings, that journals still publish twice as much on average, or maybe even four times as much when we consider the difference in length of periods. We look into the research fields in Section 6.7, but we cannot compare the number of venues over time for different research fields, as our dataset has a focus on CS and is missing dedicated venues in physics to make this comparison properly, but we might look into this in the future (Section 7.2).

Experiment 22: How are the citations and publications of the most prominent authors split between conferences and journals? (RQ5)

We split the number of publications and citations of the most prominent authors by conferences and journals (**Table 6.18**), which is inspired by the approach of Franceschet (2010). Some numbers do not add up to 100% due to publications with different document types (e.g., books), which are not covered in this table.

When we investigated the topics of the most prominent authors through their venues (Experiment 7), we got a hint that the top 5 most cited authors prefer conferences, while the most productive authors prefer journals. This trend is visible again, as all of the most cited authors publish more in conference proceedings. At the same time, eight out of nine authors also receive more citations in conferences, the exception being Anil K. Jain 0001. The publications of the most productive authors are evenly split between conferences and journals, but more than two-thirds of their citations come from journals. Franceschet (2010) finds the top 10 most productive authors publish more in conferences (63% to 34%) while using the same approach and data source (i.e., DBLP), but over 12 years this appears to have evened out. He also shows that the most prestigious authors publish more in conferences, both based on the h-index (59% to 40%), and for the winners of the ACM A.M. Turing Award (65% to 33%). This matches our finding that the most popular authors based on citations also publish more in conference proceedings.

#	Author (#Citations)	#Pub. (C)	#Pub. (J)	#Cit. (C)	#Cit. (J)
1	*Others*	3,263 (6%)	10,019 (18%)	3,370 (0%)	28,177 (4%)
2	Ross B. Girshick	56 (81%)	13 (19%)	93,437 (64%)	53,430 (36%)
3	Anil K. Jain 0001	369 (56%)	278 (42%)	32,696 (26%)	77,851 (63%)
4	Kaiming He	53 (80%)	13 (20%)	61,151 (53%)	53,179 (47%)
5	Jitendra Malik	181 (78%)	48 (21%)	75,449 (69%)	34,238 (31%)
6	Andrew Zisserman	350 (77%)	98 (22%)	73,043 (70%)	29,794 (28%)
7	Li Fei-Fei 0001	161 (83%)	25 (13%)	69,435 (68%)	33,182 (32%)
8	Luc Van Gool	622 (79%)	151 (19%)	56,774 (59%)	39,570 (41%)
9	Jiawei Han 0001	615 (70%)	188 (22%)	59,869 (63%)	19,585 (21%)
10	Trevor Darrell	243 (84%)	43 (15%)	66,277 (72%)	24,333 (27%)
	Average	294 (73%)	95 (24%)	65,348 (60%)	40,574 (37%)
#	**Author (#Publications)**	**#Pub. (C)**	**#Pub. (J)**	**#Cit. (C)**	**#Cit. (J)**
1	*Others*	3,263 (6%)	10,019 (18%)	3,370 (0%)	28,177 (4%)
2	H. Vincent Poor	698 (42%)	949 (58%)	9,390 (13%)	61,109 (82%)
3	Mohamed-Slim Alouini	643 (44%)	799 (55%)	7,206 (18%)	27,625 (70%)
4	Lajos Hanzo	429 (31%)	949 (69%)	3,382 (9%)	31,376 (87%)
5	*Wei Wang	710 (53%)	624 (47%)	3,575 (16%)	19,230 (84%)
6	Philip S. Yu	850 (66%)	406 (32%)	45,676 (62%)	26,798 (36%)
7	*Lei Zhang	749 (59%)	518 (41%)	6,222 (51%)	6,077 (49%)
8	*Yu Zhang	687 (54%)	574 (46%)	4,722 (41%)	6,658 (59%)
9	Victor C. M. Leung	571 (45%)	685 (54%)	6,111 (20%)	24,548 (80%)
10	*Yang Liu	717 (57%)	528 (42%)	6,222 (71%)	5,596 (64%)
	Average	673 (50%)	670 (50%)	10,278 (30%)	23,224 (68%)

Table 6.18 Top 10 authors based on the number of citations (top) and publications (bottom) with their publications and citations split by conferences (C) and journals (J). The average is computed excluding *Others*. Asterisks (*) denote entries, which refer to disambiguation pages in DBLP and not singular authors.

Experiment 23: Do the most cited conferences and journals show different trends considering the average citations compared to the average mass? (RQ5)

We try to approximate the most prestigious venues by taking the most cited ones and then looking at the average citations per publication (**Table 6.19**). Directly taking the venues with the highest average citations would yield venues with mostly less than 100 publications.

The top 10 most cited journals (most of which are from IEEE) have in total more citations than the top 10 most cited conferences, which we already expected, as the list of the top 10 most cited venues leans heavily toward journals (Experiment 12). Yet, the CVPR conference is the most cited venue overall and has the third-highest average citations, with IEEE Trans. Pattern Anal. Mach. Intell. having the highest average citations (203.85). Generally, conferences take most of the spots going by highest average citations (#2-#6; ECCV, CVPR, ICCV, STOC, KDD), but they

#	Conference Name	First	#Papers	#Citations	Avg. Cit.
1	CVPR	1988	12,757	1,621,492	127.11
2	ICRA	1984	24,997	790,107	31.61
3	ICASSP	1975	45,655	714,945	15.66
4	ICCV	1988	5,198	601,139	115.65
5	CHI	1982	8,864	551,752	62.25
6	INFOCOM	1983	8,785	503,263	57.29
7	ECCV	1990	3,857	490,664	127.21
8	IROS	1988	19,560	422,428	21.60
9	KDD	1999	4,367	386,964	88.61
10	STOC	1969	3,662	353,241	96.46
	Average	1985	13,770	643,600	74.34
#	**Journal Name**	**First**	**#Articles**	**#Citations**	**Avg. Cit.**
1	NeuroImage	1996	16,947	1,377,202	81.27
2	IEEE Trans. Pattern Anal. Mach. Intell.	1975	6,559	1,337,060	203.85
3	IEEE Trans. Inf. Theory	1963	16,325	1,147,862	70.31
4	Commun. ACM	1958	12,742	948,274	74.42
5	IEEE Trans. Ind. Electron.	1990	12,777	802,571	62.81
6	IEEE Trans. Signal Process.	1990	13,328	762,802	57.23
7	IEEE Trans. Autom. Control.	1991	10,762	717,910	66.71
8	IEEE Trans. Image Process.	1991	8,627	702,460	81.43
9	IEEE Trans. Commun.	1972	15,395	660,511	42.90
10	IEEE Trans. Geosci. Remote. Sens.	1987	11,297	619,250	54.82
	Average	1981	12,476	907,590	79.58

Table 6.19 Top 10 most cited conferences (top) and journals (bottom).

also take the last spots (#18-#20; ICRA, IROS, ICASSP), which shows a greater fluctuation in average citations compared to journals. Considering the top ranks are occupied by IEEE Trans. Pattern Anal. Mach. Intell., ECCV, CVPR, and ICCV we see a heavy focus on computer vision and pattern recognition again, similar to the most cited authors (Experiment 7) and most cited venues (Experiment 12). We conclude that on average the top journals are still a bit more prestigious regarding citations compared to conferences, but the gap compared to the average mass, where journals have twice as many citations (Experiment 20), is a lot smaller. A few highly cited conferences are also able to rank higher based on average citations compared to highly-cited journals.

Other works show elite conferences are getting more citations than elite journals, based on different measures for quality. Rahm and Thor (2005) use select high-quality venues from their research field (databases), compare their citations, and find that conferences have a higher citation impact than journals. Vrettas and Sanderson (2015) also compare conference and journal citations and find little difference overall, but when the quality of the venues is considered (using a ranking from the ERA assessment), high-quality conferences have higher average citation rates than high-quality journals, and low-quality journals get more citations than low-quality conferences. In NLP

Mohammad (2020a), on the other hand, finds that journal publications have higher average and median citations than both top-tier and non-top-tier conferences. We also find a higher fluctuation of average citations in the top 10 most cited conferences, which reflects the fluctuation of conference quality Vrettas and Sanderson (2015) find. Higher citation rates for elite conferences are only visible to a limited degree in our research, which we attribute to us only using an approximation of the highest-quality conferences and not a more refined solution to measure the quality of venues.

Lastly, the average number of publications is a bit higher for conferences but also fluctuates more compared to journals. Thus, this experiment also reflects the findings from the most cited and most productive venues (Experiment 12), where the most productive venues are mostly conferences and the most cited venues are mostly journals.

Experiment 24: How do the topics of conferences and journals change over time? How do the topics differ? (RQ5)

Compared to previous experiments (Experiments 8 and 14), we only use single years (**Table 6.20**), as full five-year periods are currently not possible due to technical limitations (e.g., memory size).

The words that are common across all columns (e.g., “paper”) are high up again, and there are more than in previous experiments, which we expected after the results from grouping the most cited and most productive venues (Experiment 14). There are now even more titles and abstracts used to generate these lists and saliency already failed to rank these terms lower when investigating the trends in venues over time (Experiment 14). Many of the terms that appear in all five years and occur in both conferences and journals either tell us nothing about the topics (i.e., “paper”, “propos”, “problem”, “provid”, “method”, and “present”) or only very little (i.e., “network”, “imag”, “algorithm”, “model”, and “control”). The same is true for the terms “develop” and “time”, which are present in all five years for conferences, and “data” and “result” for journals. We can see “imag” which relates to computer vision again and “network” and “model” might relate to neural networks, but in this case, we have to consider there are more engineering-related venues in our dataset, as we saw in Experiments 7 and 12, so neural networks might only make up a part of it and physical networks the other. However, in 2019 “learn” and “train” appear for the first time for conferences together with the repeating terms “featur” and “optim”, which does show a rise in approaches leveraging neural networks and also matches the late appearance for the top venues (Experiment 14) and most cited authors (Experiment 8), where we also related this finding to previous research. The term “learn” also appears in 2019 for journals together with “feature”, “optim”, and “object” which can also indicate usage of neural networks, but none of those terms appear for the first time in 2019. Also, the term “mobil” only appears in 2004 for conferences and “signal” for journals in 2004, which could indicate more research related to wireless communication, e.g. mobile phones and their technologies.

Overall, there are only a few differences between journals and conferences visible

Conferences					Journals				
1999	2004	2009	2014	2019	1999	2004	2009	2014	2019
paper	*propos*	*paper*	*propos*	*propos*	*propos*	*paper*	*paper*	*propos*	*propos*
propos	*paper*	*present*	*network*	*algorithm*	*present*	*problem*	*propos*	*paper*	*paper*
data	*present*	*propos*	*paper*	*paper*	*control*	*propos*	*algorithm*	*model*	*network*
network	*network*	*network*	*present*	result	*paper*	*provid*	*network*	*algorithm*	*result*
imag	*algorithm*	*time*	*problem*	*time*	*algorithm*	*network*	*present*	*network*	learn
problem	*provid*	result	*time*	*problem*	*problem*	*algorithm*	*control*	*present*	time
provid	*time*	*method*	data	**learn**	*provid*	*result*	*result*	*provid*	*provid*
present	*imag*	*model*	*provid*	*network*	*result*	*present*	*problem*	*problem*	*present*
object	result	*imag*	*model*	*imag*	*network*	*data*	*data*	*method*	*imag*
user	*problem*	*provid*	*imag*	*present*	us	time	*model*	*imag*	*control*
algorithm	*method*	process	user	differ	*model*	*imag*	order	number	*model*
set	work	user	featur	**experi**	number	*control*	*method*	develop	*data*
method	**mobil**	*algorithm*	*method*	featur	perform	set	*provid*	time	featur
develop	requir	*problem*	**implement**	*control*	inform	inform	time	*control*	*problem*
control	perform	*control*	*control*	data	includ	*method*	develop	optim	*method*
test	real	perform	differ	**state**	**bound**	perform	function	*result*	studi
time	user	set	process	*provid*	obtain	*model*	inform	featur	*algorithm*
allow	*control*	work	result	*model*	set	**consid**	research	*data*	optim
design	*develop*	featur	power	real	*imag*	**system**	**work**	**design**	test
model	allow	requir	optim	detect	*data*	solut	*imag*	requir	develop
inform	**program**	optim	*algorithm*	test	**given**	**compar**	number	obtain	**accuraci**
robot	**scheme**	*develop*	servic	**comput**	**exampl**	function	solut	order	object
requir	servic	test	detect	optim	graph	test	obtain	demonstr	function
environ	node	power	*develop*	compar	code	includ	graph	perform	us
compar	*model*	servic	**demonstr**	*method*	*method*	number	servic	set	**task**
obtain	introduc	node	studi	*develop*	**describ**	**signal**	us	studi	**estim**
introduc	**reduc**	**achiev**	**signal**	**consid**	object	**exist**	code	servic	demonstr
distribut	**protocol**	design	**increas**	**train**	requir	**structur**	featur	research	**increas**
servic	object	differ	**research**	power	**differ**	**paramet**	learn	function	obtain
import	**estim**	**interact**	**larg**	studi	**technolog**	develop	includ	includ	**condit**

Table 6.20 Top 30 most salient terms for conferences and journals per year. Terms that only appear in one year are **bold** and terms that appear in all five years are in *italics*. Terms that appear in all 10 years are additionally greyed out.

in Table 6.20, which shows the topics are mostly the same. We also do not see many trends over time. Considering Fiala and Tutoky (2017) can leverage the approach of highlighting terms to find trends, we attribute our issues to the high number of common terms and using saliency instead of frequency, similar to the issues already mentioned when investigating the trends of the top venues over time (Experiment 14). In our experiments, we use only 10 topics due to technical limitations (e.g., available memory size), but Anderson et al. (2012) use 73 topics just to cluster publication in NLP, and in Xia et al. (2021)'s research CS covers 15,460 topics, which makes up 16% of their total topics across all research fields. As saliency is calculated from the distinctiveness of terms across all topics we believe the saliency measure would work better with more topics, but this would exceed our current technical capabilities. In the future, we might conduct these term-related experiments again and then include the necessary changes (i.e., filtering common words beforehand and using a different measure) (Section 7.2). Then we might also be able to see some of the trends Coşkun et al. (2019) see (e.g., a shift to topics such as privacy, security, IoT, and big data), or the important current topics Tattershall et al. (2020) and Xia et al. (2021) uncover.

6.7 Fields of Study

This section answers the last research question (RQ6), which focuses on the fields of study (e.g. CS, medicine). We conduct the experiments by aggregating CS-Insights's data by the fields of study and then investigating their distribution (Experiment 25) and the differences between CS and other fields of study regarding preferences for conferences/journals, and topics (Experiments 26 to 27). CS-Insights and D3 are built with data from DBLP, which focuses on CS publications, but D3 also includes data on the fields of study, which makes this analysis possible. As Experiment 25 shows, most publications in CS-Insights are from CS. Even though other fields are also included (each publication can have multiple fields of study), we have to be aware that the publications likely still have strong ties to CS or they would not be included in DBLP.

Experiment 25: How are the publications distributed across the fields of study? (RQ6)

As expected, most publications are from CS (86%), while nearly 700,000 publications (14%) do not have any field of study assigned (**Table 6.21**). This leaves only 2,747 publications, that have a field of study assigned but are not from CS, so when we analyze other fields of study besides CS in the next experiments, we should keep in mind those publications likely all additionally have CS as a field of study. Mathematics, Engineering, and Medicine then take up rank #3-#5, and Psychology has the highest average citations per publication (39.71). Interestingly, nearly all citations are also for publications from CS, and publications without fields of study only have a total of 781 citations. We assume the publications without fields of study cause general issues in the matching processes for both the fields of study and the citations. Possible reasons

Field of Study	First	#Papers	#Citations	Avg. Citations
Computer Science	1936	4,192,059 (86%)	97,037,289 (100%)	23.15
Others	1936	698,734 (14%)	781 (0%)	0.00
Mathematics	1936	696,143 (14%)	21,778,856 (22%)	31.29
Engineering	1936	324,195 (7%)	7,554,187 (8%)	23.30
Medicine	1936	267,808 (5%)	10,179,977 (10%)	38.01
Psychology	1948	80,578 (2%)	3,200,102 (3%)	39.71
Physics	1946	67,195 (1%)	1,400,731 (1%)	20.85
Business	1953	57,282 (1%)	1,497,035 (2%)	26.13
Materials Science	1951	50,049 (1%)	619,066 (1%)	12.37
Biology	1961	30,751 (1%)	985,557 (1%)	32.05
Economics	1953	30,084 (1%)	1,063,618 (1%)	35.35
Sociology	1955	27,719 (1%)	762,886 (1%)	27.52
Environmental Science	1955	26,356 (1%)	410,730 (0%)	15.58
Chemistry	1952	17,180 (0%)	547,508 (1%)	31.87
Geology	1957	16,928 (0%)	304,352 (0%)	17.98
Geography	1953	16,010 (0%)	407,273 (0%)	25.44
Political Science	1941	15,987 (0%)	245,115 (0%)	15.33
Philosophy	1936	5,718 (0%)	61,511 (0%)	10.76
Art	1956	4,334 (0%)	18,521 (0%)	4.27
History	1939	2,760 (0%)	49,628 (0%)	17.98

Table 6.21 Distribution of publications across fields of study. One publication can have multiple fields of study, so the numbers exceed 100% when added.

could be no or only a bad quality full-text, but these issues should be fixed with the new version of the dataset (Section 7.2).

Experiment 26: How is the split between conference papers and journal articles for different fields of study? (RQ6)

We observe nearly every field publishes more in journals than conferences, except CS and Engineering (**Table 6.22**). Engineering has an even larger percentage of publications in conferences (61% to 37%) than CS (55% to 43%). Michels and Fu (2014) find that CS has the highest preference for conferences of all research fields (≈77% to 23%), followed by Engineering (e.g., Electrical Engineering: ≈72% to 28%), based on Web of Science data from 2009. In previous experiments, we already showed researchers in CS slowly shift their publications more toward journals over time (Experiments 19 and 23), which can explain the shift from the numbers Michels and Fu (2014) report (≈77% to 23%) to our numbers (55% to 43%). As we do not investigate Engineering, we cannot say if there was a shift toward conferences or if our data on Engineering is insufficient and skewed too much by CS. We assume it is the latter as Vrettas and Sanderson (2015) also find Engineering has more than 10 times the number of journals compared to conferences, even though it is placed second after CS in terms of the total number of conferences. Investigating this further would require a dataset more focused on Engineering, which we might look into in the future. This

Field of Study	#Papers (C)	#Articles (J)	#Citations (C)	#Citations (J)
Computer Science	2,318,335 (55%)	1,786,215 (43%)	35,420,219 (37%)	58,767,760 (61%)
Others	255,474 (37%)	421,687 (60%)	118 (15%)	73 (9%)
Mathematics	246,873 (35%)	444,253 (64%)	4,942,536 (23%)	16,251,144 (75%)
Engineering	197,828 (61%)	121,161 (37%)	2,645,846 (35%)	4,725,451 (63%)
Medicine	51,424 (19%)	215,234 (80%)	492,404 (5%)	9,637,120 (95%)
Psychology	31,924 (40%)	47,179 (59%)	379,265 (12%)	2,725,165 (85%)
Physics	26,168 (39%)	42,704 (64%)	236,146 (17%)	1,110,054 (79%)
Business	9,725 (17%)	29,307 (51%)	142,053 (9%)	1,236,968 (83%)
Materials Science	22,922 (46%)	26,818 (54%)	121,386 (20%)	493,004 (80%)
Biology	9,163 (30%)	21,202 (69%)	77,018 (8%)	887,691 (90%)
Economics	6,140 (20%)	23,595 (78%)	76,043 (7%)	972,169 (91%)
Sociology	9,725 (35%)	17,179 (62%)	142,053 (19%)	586,270 (77%)
Environmental Science	12,242 (46%)	13,904 (53%)	48,661 (12%)	360,013 (88%)
Chemistry	2,911 (17%)	14,077 (82%)	19,666 (4%)	525,262 (96%)
Geology	5,665 (33%)	11,113 (66%)	30,379 (10%)	272,702 (90%)
Geography	6,783 (42%)	8,272 (52%)	85,174 (21%)	299,778 (74%)
Political Science	6,140 (38%)	8,762 (55%)	52,171 (21%)	141,718 (58%)
Philosophy	1,233 (22%)	4,054 (71%)	3,230 (5%)	47,468 (77%)
Art	2,364 (55%)	1,740 (40%)	7,564 (41%)	9,211 (50%)
History	1,005 (36%)	1,424 (52%)	31,992 (64%)	13,606 (27%)

Table 6.22 Split of publications and citations between conferences (C) and journals (J) per field of research.

preference of Engineering toward conferences also helps to explain why many of the most productive venues were IEEE conferences with a focus on engineering-related tasks (Experiment 12). For the other fields of study, we can still see the preference for journals, which Michels and Fu (2014) also report. They also find all fields of study (including CS and Engineering) either have the same citation rates or higher ones for journals compared to conferences, which is also visible in Table 6.22. Only *Others* and History have a higher percentage of citations in conferences in our case, but we explain this with the small number of citations for *Others* and the small number of publications in History being affected too easily by fluctuations.

Experiment 27: Are the topics of the top research fields different from CS? Does CS influence other research fields? (RQ6)

We only analyze the top fields of study based on the number of publications excluding *Others* and any field of study with only 1% of papers or less, which leaves us with the top 5 fields of study (**Table 6.23**).

There clearly are some differences between CS and other fields of study even though we are using a dataset with a CS focus. In 1999 the unique terms show that Mathematics deals with graph theory (“graph” and “function”) and Engineering with electrical engineering (“power”, “circuit”, and “voltag”), while we see more medical terms in Medicine (“patient”, “medic”, and “neuron”), and Psychology deals with more social aspects (“group”, “team”, “emot”). CS only has a few unique terms in 1999 (“data”, “test”, and “implement”), because many are shared across other fields of study (“imag”, “network”, and “object”). We explain this with our data source DBLP having a focus on CS and thus the publications of the other fields of study

		1999					2019		
CS	**Mathematics**	**Engineering**	**Medicine**	**Psychology**	**CS**	**Mathematics**	**Engineering**	**Medicine**	**Psychology**
paper	paper	control	**patient**	**internet**	result	*propos*	*propos*	imag	**student**
propos	present	paper	imag	learn	network	*paper*	network	*propos*	*learn*
problem	problem	propos	fuzzi	**group**	*propos*	**graph**	provid	*paper*	data
imag	**graph**	imag	**sequenc**	health	*learn*	problem	imag	present	**game**
present	imag	present	**medic**	**technolog**	*paper*	method	**neural**	**mri**	*paper*
provid	code	robot	health	**face**	provid	**consid**	*paper*	**edr**	**task**
algorithm	**given**	**power**	algorithm	**knowledg**	imag	present	present	signal	show
object	control	provid	**movement**	task	present	**comput**	research	*learn*	**aim**
network	system	**circuit**	care	**team**	**featur**	control	method	model	includ
result	obtain	describ	inform	**agent**	data	algorithm	includ	**shore**	**conduct**
data	algorithm	**demonstr**	virtual	softwar	algorithm	network	develop	**respir**	**interact**
design	motion	result	**cluster**	care	time	*learn*	deep	network	experi
inform	**filter**	**voltag**	**clinic**	virtual	method	model	*learn*	result	**virtual**
time	**set**	algorithm	**detect**	commun	develop	**system**	optim	**measur**	**signific**
control	result	signal	**rule**	**emot**	**studi**	provid	signal	method	**educ**
test	signal	**color**	code	**copi**	problem	**set**	**vehicl**	deep	**design**
code	**estim**	**manipul**	**model**	inform	**us**	data	**convolut**	**shell**	**social**
develop	time	consid	learn	**social**	experi	**obtain**	train	**fetal**	**emot**
requir	object	motion	**relat**	**cours**	**reduc**	**equat**	**increas**	power	**robot**
describ	**function**	softwar	servic	**support**	**commun**	**bound**	**accuraci**	data	perform
differ	**exampl**	**engin**	commun	**journal**	**user**	**state**	result	**patient**	**languag**
servic	network	develop	**standard**	research	compar	**code**	control	**apnea**	**find**
allow	**us**	**fault**	develop	**comput**	**detect**	**numer**	power	provid	result
featur	**bound**	**current**	**studi**	**visual**	optim	**solut**	**segment**	optim	**suggest**
obtain	fuzzi	research	task	**teach**	**process**	**introduc**	compar	**sensor**	**discuss**
consid	**known**	**track**	**neuron**	**work**	**differ**	**distribut**	**end**	show	**subject**
robot	**appli**	design	system	**older**	research	time	achiev	estim	**brain**
discuss	**nois**	spl	object	network	train	estim	**energi**	achiev	*propos*
includ	spl	**forc**	**measur**	**user**	perform	**given**	**better**	**demonstr**	**function**
implement	**sub**	**manag**	motion	**eb**	**term**	**number**	**speech**	**predict**	research

Table 6.23 Top 30 most salient terms for the top 5 fields of study in 1999 and 2019. Terms that only appear in one field of study are **bold** and terms that appear in all five fields of study are in *italics*.

automatically being related to CS in some way, or they would not be in DBLP. In 2019 one could argue for CS the unique terms ("featur", "commun", and "detect") together with some not unique terms ("train", "learn") indicate a trend toward neural networks again, which we already saw in previous experiments (Experiments 8, 14 and 24). The term "learn" is present in each field in 2019, but the other fields also show unique terms from their research areas again. Mathematics again covers graph theory ("graph") in 2019, but also other fields ("bound", "equat", "numer", "distribut"). Engineering deals with "vehicl" and "energi", but also seems to leverage neural networks more ("convolut", "neural"), which is highlighted even more by the not unique terms ("deep", "learn"), possibly in the context of autonomous driving. In Medicine, we again see medical terms ("mri", "respir", "apnea"), but also a trend to neural networks with the not unique terms ("network", "deep", "learn"), while Psychology sticks with social aspects ("game", "social", "emot"). We conclude there are certainly distinct topics in other fields of study besides CS, even though the underlying data of CS-Insights focuses on CS. The trend toward adopting approaches leveraging neural networks is also visible outside of CS. This is also covered in other literature, e.g., for medicine Aggarwal et al. (2022) show the importance of artificial intelligence, machine learning, and deep learning and their gains for healthcare in face of the COVID-19 pandemic.

6.8 Summary

In this chapter, we conducted 27 experiments to answer our research questions (Section 1.2). Here, we shortly show that all research questions were answered and reference the corresponding experiments. Our most interesting findings are listed in the conclusion in the next chapter (Section 7.1).

RQ1 How many publications, authors, and venues are in our dataset? How do the numbers change over time? How many authors and venues are currently active?

The dataset overview showed the number of publications, authors, venues, and more (Table 6.1). We investigated the changes in the numbers per year for publications (Experiment 1), authors (Experiment 3), and venues (Experiment 9). The activity for the last five years was also shown for authors (Experiment 4) and venues (Experiment 10).

RQ2 How are the citations and publications distributed across authors and venues? How do the distributions change over time?

We covered the trends in the distribution of the number of citations and papers over multiple periods for authors (Experiment 5) and venues (Experiment 11).

RQ3 What are the most prominent authors and venues? Are there preferences for topics? Do the topics change over time?

We showed the most cited and most productive authors (Experiment 6), their preferences for venues/topics (Experiment 7), and how the topics changed over time (Experiment 8). Similarly, we covered the most cited and publishing venues (Experiment 12), their preferences for topics (Experiment 13), and how the topics changed over time (Experiment 14).

RQ4 How do incoming and outgoing citations evolve over time? How do their distributions differ?

We investigated the changes in the number of incoming and outgoing citations per year (Experiment 15), the distribution of incoming citations based on citation bins (Experiment 16), and the trends in the distribution of incoming and outgoing citations over multiple periods (Experiment 17).

RQ5 How do conferences and journals compare in their number of publications and citations over time? How do the top venues and topics differ? Do top authors prefer conferences or journals?

After showing the distribution of document types (Experiment 18), we covered the development of conferences and journals over time regarding their number of publications per year (Experiment 19), their distribution of citations (Experiment 20), and topics (Experiment 24). The most cited conferences and journals (Experiment 23), and the preferences of top authors for conferences and journals (Experiment 22) were also examined.

RQ6 How do the most prominent fields of study differ from CS in topics and preference for conferences or journals?

Lastly, we investigated the distribution of the fields of study (Experiment 25) and how other fields (e.g., medicine) differed from CS regarding their preference for conferences or journals (Experiment 26) and their topics (Experiment 27).

7 Final Considerations

This chapter presents the final considerations of this thesis. We start with the conclusion of our experiments from the previous chapter and a summary of our contributions (Section 7.1) and close this thesis with the current limitations and future work of our research (Section 7.2).

7.1 Conclusion

Our goal was to analyze the state of CS research, covering authors, venues, document types, fields of study, and their publications, citations, and topics to uncover implicit patterns in CS literature. To achieve this, we introduced Computer Science Insights (CS-Insights), an interactive, responsive open-source browser-based visualization system to facilitate the exploration of CS publications, and the DBLP Discovery Dataset (D3) that contains metadata associated with 6m CS papers in its newest version[1]. The CS-Insights system crawls and processes publications from DBLP and enriches them with additional metadata (e.g., citations and abstracts) from their full-texts to create D3. CS-Insights is also built in a modular architecture to facilitate the maintenance and incorporation of more efficient components in the future. Both CS-Insights[2] and D3[3] are fully open-access and freely available online in their respective GitHub repositories. To reproduce the results from our work the original version of D3 is also available online[4].

We then used CS-Insights to conduct a case study on CS and demonstrate its capabilities. Some of the most interesting and relevant findings are listed below.

- CS attracts increasingly more new authors (30% joined in the last five years and those 30% make up half of all authors who published in the last 5 years), who also publish more papers per year. At the same time, the number of venues and their publications per year also increases.
- Incoming (received) citations peak in 2009, and fall off before and after, while on average each paper cites more other papers with each passing year. Yet, 29% of all publications have no citations, and only a third get 10 or more citations.
- The most cited authors do their research in computer vision and pattern recognition and prefer to publish in journals, while the most productive authors cover signal processing and communication, and prefer to publish in conferences. Similarly, the most cited venues are journals and focus on computer vision and pattern recognition, while the most productive venues are mostly conferences and more focused on engineering topics.

[1] https://zenodo.org/record/7069915
[2] https://github.com/gipplab/cs-insights
[3] https://github.com/jpwahle/lrec22-d3-dataset
[4] https://zenodo.org/record/6477785

- In total, most publications in CS are from conferences (53%). Due to the COVID-19 pandemic, the number of conference papers dropped in 2020, also affecting the overall number of publications in 2020, but the number of journal articles appeared unaffected. This made journals again overtake conferences in the number of publications per year, a first since 1992. A shift back to journal articles was visible before 2020, as the gap between journal articles and conference papers per year was already getting smaller in recent years.
- Journal articles get on average twice as many citations as conference papers, a trend that has been visible for decades. The citation gap between the most cited journals and conferences is much smaller, but still favors journals. Some highly cited conferences reach on average more citations per publication than highly cited journals, but the average citations of highly cited conferences also fluctuate more than for highly cited journals.
- CS and engineering are the only fields favoring conferences over journals considering the number of publications, but all investigated fields of study get more citations in journals than conferences.
- Overall, an increase in the popularity of approaches leveraging neural networks was visible in 2015-2019 across conferences, journals, and top fields of study, authors, and venues.

We conclude that CS appears to be a strongly growing and very active field, and with a scientometric analysis supported by CS-Insights, we can show its characteristics, trends, and implicit patterns through its core components and attributes (i.e., publications, authors, venues, citations, topics, document types, and differences to other fields of study).

The same methodology we used to analyze CS can also be applied to other areas, by using the CS-Insights system to conduct the same experiments on different datasets (see future work; Section 7.2) from other research fields (e.g., medicine) or sub-fields (e.g., NLP). To the best of our knowledge, no other researchers have conducted a study into CS as extensive as we did in this thesis. While many authors already looked into certain aspects of CS, it was always only partially, with always different datasets, and different approaches. Some authors then get contradicting results and it becomes hard to determine the exact reason, e.g., when investigating if conferences or journals get more citations (Rahm and Thor 2005; Franceschet 2010; Vrettas and Sanderson 2015). Having a common and flexible system that can efficiently perform the same analysis on different datasets allows for better comparability of scientometric research in the future. Comparing datasets more easily might also enable researchers to make more informed decisions about which dataset they want to use for their research.

7.2 Limitations & Future Work

In this section, we cover limitations and future work regarding the data, backend structure, features of the frontend, and analysis, some of which were already mentioned in Chapters 4 and 6. We show most of the current limitations are already planned to be fixed in future work. More details on future versions, features, and improvements of the CS-Insights system are also available in a roadmap on GitHub[5].

Data

Even though DBLP is the largest repository of CS publications, with an extensive list of features at its disposal, it does not contain all publications about CS (e.g., journal and conference volumes without openly available metadata cannot be automatically indexed by DBLP[6]). We are now in the process of switching the data source to Semantic Scholar, as we can access their data, we are allowed to use it, and it gives us more flexibility in the future. The crawler was already changed to get the data from Semantic Scholar and not DBLP[7], but the structure in the backend is not yet adjusted accordingly. Using data from Semantic Scholar allows us to extract the same publications we already covered with DBLP, but also cover other research fields (e.g., physics) or datasets (e.g., PubMed, ACL Anthology) integrated into Semantic Scholar in a unified way. This way we will be able to compare research fields and datasets more easily in the future, as explained in Section 7.1. An additional benefit of using Semantic Scholar data is that it fixes the current issues from the broken export of D3, e.g., publications that got missing, duplicate publications, author names missing special characters (e.g., umlauts; see Experiment 2) or not being disambiguated (e.g., Saif M. Mohammad, Saif Mohammad), and venues with an increasing counter in DBLP (e.g., "HCI (42)"; see Experiment 10). We can then also use Semantic Scholar's citation counts, which removes the necessity to use GROBID for the extraction of bibliographies. This should fix the citation counts for publications without research fields (Experiment 25) and the incoming and outgoing citation counts not matching (Experiment 15). The new dataset also adds newer publications, as our current dataset from DBLP was crawled on 2 December 2021 and thus is not complete for 2021. Some limitations remain with Semantic Scholar data, e.g., the missing affiliations (institutions and countries) and publishers, or not being able to make some abstracts available in D3 due to copyright reasons. However, Semantic Scholar contains other measures for authors (e.g., influential citations and h-index) and we also thought about adding other quality measures for venues in the future (e.g., impact factor or grade) from other sources.

We also plan some data-related features besides switching to Semantic Scholar, e.g., adding an update functionality that automatically updates our data by adding new

[5] `https://github.com/users/jpwahle/projects/1`

[6] `https://dblp.org/faq/5210229.html`

[7] We already used the new crawler to get the DBLP data from Semantic Scholar in our newest version of D3 available on zenodo: `https://zenodo.org/record/7069915`

publications and updating old ones. Another feature would be an import functionality for other data sources where we cannot provide the data ourselves due to restricted access (e.g., Web of Science, Scopus). Should the users have access to those services (e.g., through their institution) and be able to download the data, they could then import it into their local version of CS-Insights to analyze it.

Backend Structure

Some queries for the authors take up to a minute on a good machine, but a machine with less memory and a worse hard drive might need several minutes. This severely limits people without access to a better (and more expensive) machine and requires more server resources to host our demo. As we are already changing the backend schema for the Semantic Scholar data, we are also looking into increasing the performance of the backend. One idea is to change the schema and copy all data to each collection to pre-aggregate it by authors, venues, etc. (Section 4.2.1). Another idea is to use a different database, e.g., PostgreSQL[8] or Neo4j[9] (a graph database).

Frontend Features

We also intend to expand the frontend functionalities in the future by improving the current dashboards and visualizations and building new ones based on the things we learned from the analysis. The goal is to automate more analyses but still allow a great variety of analyses and visualizations, so researchers can continue to explore the field of CS with CS-Insights and D3. For example, we plan to add line charts for the average authors per paper per year and/or papers per author per year, as we currently cannot compute either from the currently available information (CS-Insights only shows the unique number of authors per year; see Experiment 5). Line charts to visualize the average number of publications or citations over time are also possible. New features apart from the visualizations are also planned (e.g., a "NOT" option for filters or better account management), or new features leveraging new models in the prediction endpoint (e.g., a frequency measure for terms; see Experiments 14 and 24). CS-Insights does not have a search function for publications or authors and we do not plan to add one, as both DBLP and Semantic Scholar already provide this functionality.

We also consider some larger features, which would take a considerable amount of time to implement and which we present two of in this paragraph. The first idea is to make comparisons easier by selecting two sets of filters and showing the information side by side, e.g., with two lines or bars in the same visualization. Another idea includes leveraging our data for analyses with networks and graphs. We already have data on incoming and outgoing citations in D3 through our crawler (Section 4.1.3), which Semantic Scholar will continue to provide after the switch, so citation networks would be possible. One potential approach to implementing this would be to integrate

[8]https://www.postgresql.org/
[9]https://neo4j.com/

VOSViewer just like Zeta Alpha does for the terms of publications. Creating networks for co-authorship or terms from titles or abstracts would then also be a possibility.

Analysis

Generally, the analysis in this thesis (Chapter 6) is limited by the features of CS-Insights's UI, as we do not conduct any analysis directly on the data and only through the UI CS-Insights provides. During the analysis we also found drawbacks of our approaches and some other interesting aspects to look into in the future, which we cover in the following.

We conducted multiple experiments using data from 1960-2019, and used two different lengths for the time periods (i.e., 1960-1999 used 10 years and 2000-2019 five years). Originally, this worked well to uncover trends in the number of citations and papers over time (Experiments 5, 11, 17 and 20) and the distribution appeared unaffected (i.e., first quartile, median, third quartile, average). The maximum was only introduced later on and showed the first issues, as it sometimes sank when crossing from 1990-1999 to 2000-2004. Experiment 21 then showed a noticeable effect, which also skewed the numbers for the average publications per venue but, unfortunately, that was the last experiment we conducted using this approach. In retrospect, it might have been better to stick to one length for the periods, but start in 1980 to keep it at eight periods and not blow up the tables too much compared to 12 five-year periods starting in 1960 would result in.

Another issue we ran into was the saliency measure used in our topic modeling visualization. For some experiments, the measure worked well and we got good results (Experiments 8, 13 and 27), but others caused issues (Experiments 14 and 24). Saliency measures the distinctiveness of terms across all topic clusters and boosts the terms that are exclusive to specific topic clusters. Experiment 23 grouped five venues, but the topic model could perfectly sort those five venues into 10 topics, which made venue-specific terms (e.g. "brain") nearly exclusive to one topic and thus saliency boosted the terms. Generic terms (e.g., "paper") also still appeared in high ranks, which saliency should prevent in theory. In Experiment 24, similar issues became apparent, as most terms were generic terms, and the saliency measure failed to rank them lower. This might be due to trying to fit all publications of CS from various sub-fields into 10 topics, while other researchers use more topics, which was not possible for us due to technical limitations. Considering the drawbacks of these two experiments and the resources required to overcome the technical limitations to make saliency more viable, a normal frequency measure with a filter for generic terms might prove more useful in the future.

We also found some interesting aspects while conducting our experiments we did not further explore in this thesis. When investigating the most cited and most productive venues (Experiment 12) it appeared as if older venues are more respected and get more citations. It might be interesting to see how the citations differ between more established venues and newer ones, and how long it might take newer once to get more established and also receive more citations. In the same experiment, we found large differences once we only considered open-access publications. Investigating the difference between venues that make their publications open-access, and venues that lock them behind a paywall might show differences in their impact, considering authors can more easily verify the contents of open-access papers for references, but paid-access papers might be published in more renowned venues. Lastly, a comparison between singular venues (Experiment 14) and their authors, topics, citations, etc. might generally yield interesting results.

A Appendix

A.1 Additional Tables/Figures

Attribute	Example
publication	
id	conf/acl/Mohammad20b
modified date	2021-09-12
title	NLP Scholar - An Interactive ...
pages	232-255
year	2020
type	Conference and Workshop Papers
access	open
links	[https://doi.org/...]
doi	10.18653/v1/2020.acl-demos.27
publisher	ACL
author	
id	58/380
fullname	Saif M. Mohammad
webpage	http://saifmohammad.com/
venue	
names	[International Conference on Lang...]
acronyms	[LREC]
type	Conference or Workshop
id	conf/lrec
affiliation	
id	4eb3...f094
name	National Research Council Canada
country	Canada
city	Ottawa
postcode	K1A 0R6
addressline	1200 Montreal Road, Bldg. M-58
publication	
outgoing citations	
ids	[7615..., 76af...]
count	2
incoming citations	
ids	[7ca5..., 7d0e...]
count	11
keywords	[Scientometrics, Citations, ...]
ocr title	NLP Scholar: An Interactive ...
ocr abstract	As part of the NLP Scholar ...

Table A.1 D3 attributes as proposed in Wahle et al. (2022) (top half: data from DBLP, bottom half: data extracted from full-texts).

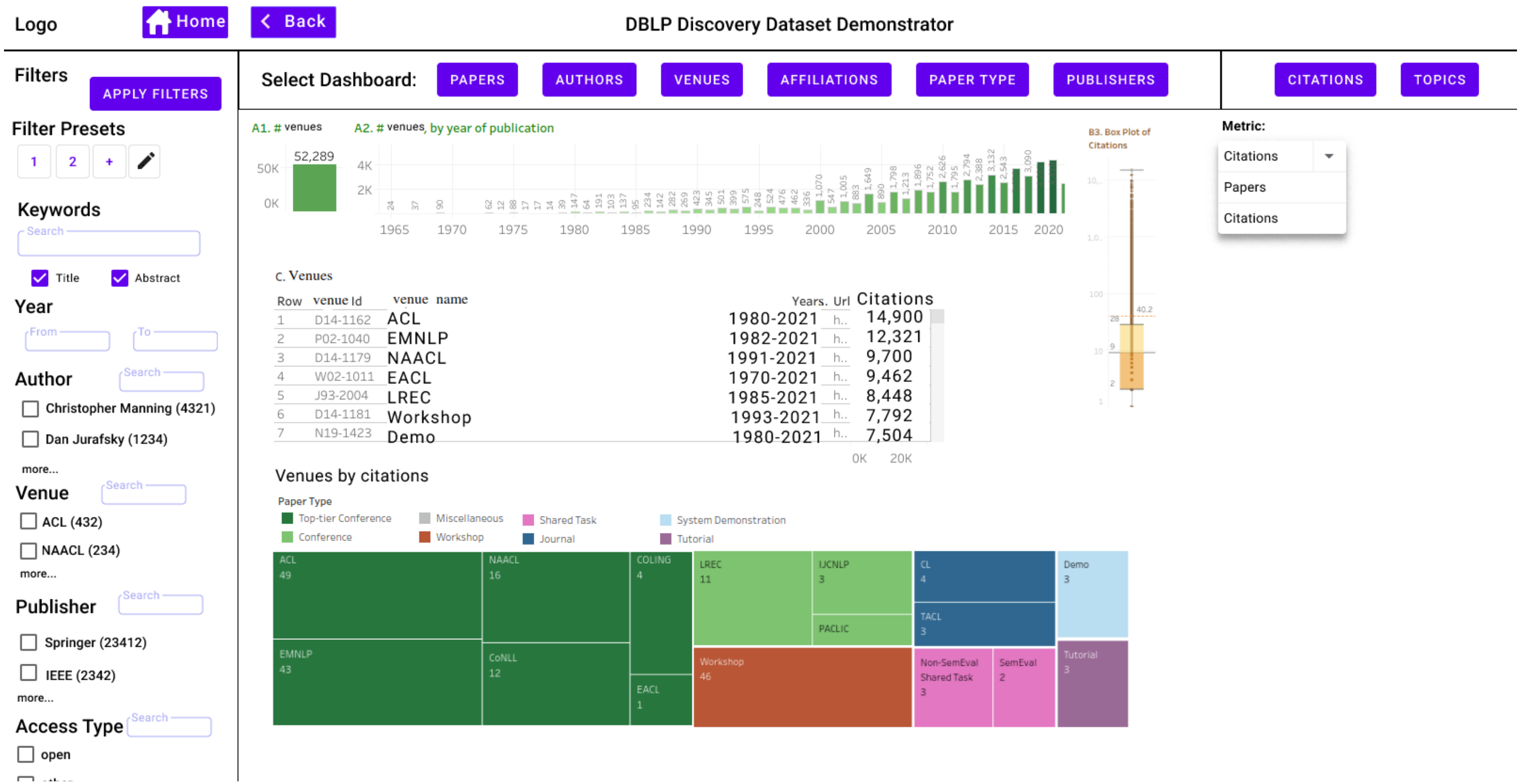


Figure A.1 Final prototype of the CS-Insights frontend (current selection: venues dashboard). The graphs shown are taken from NLP Scholar (Mohammad 2020c) with the author's permission and altered for the prototype.

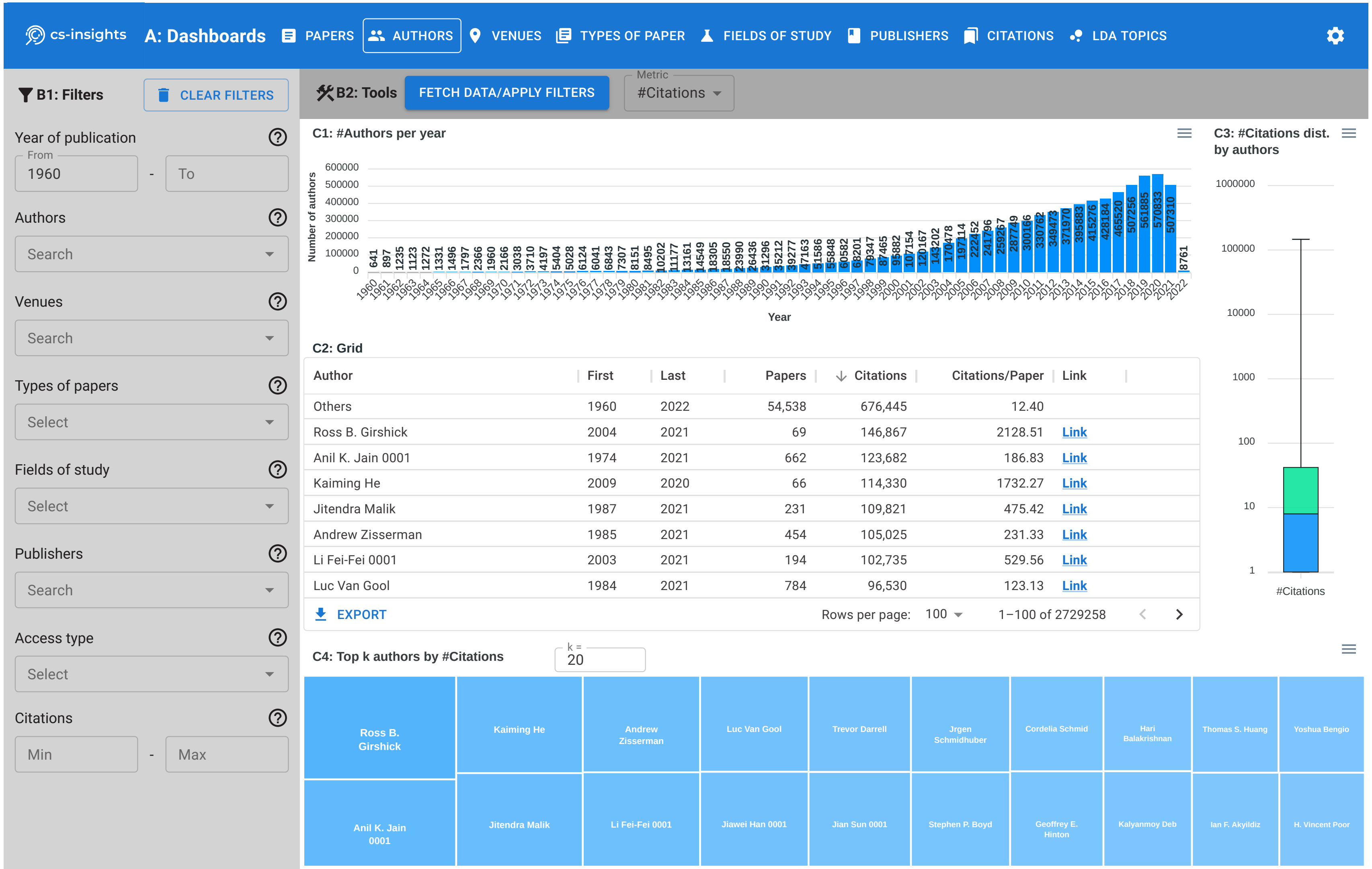


Figure A.2 Authors dashboard in CS-Insights.

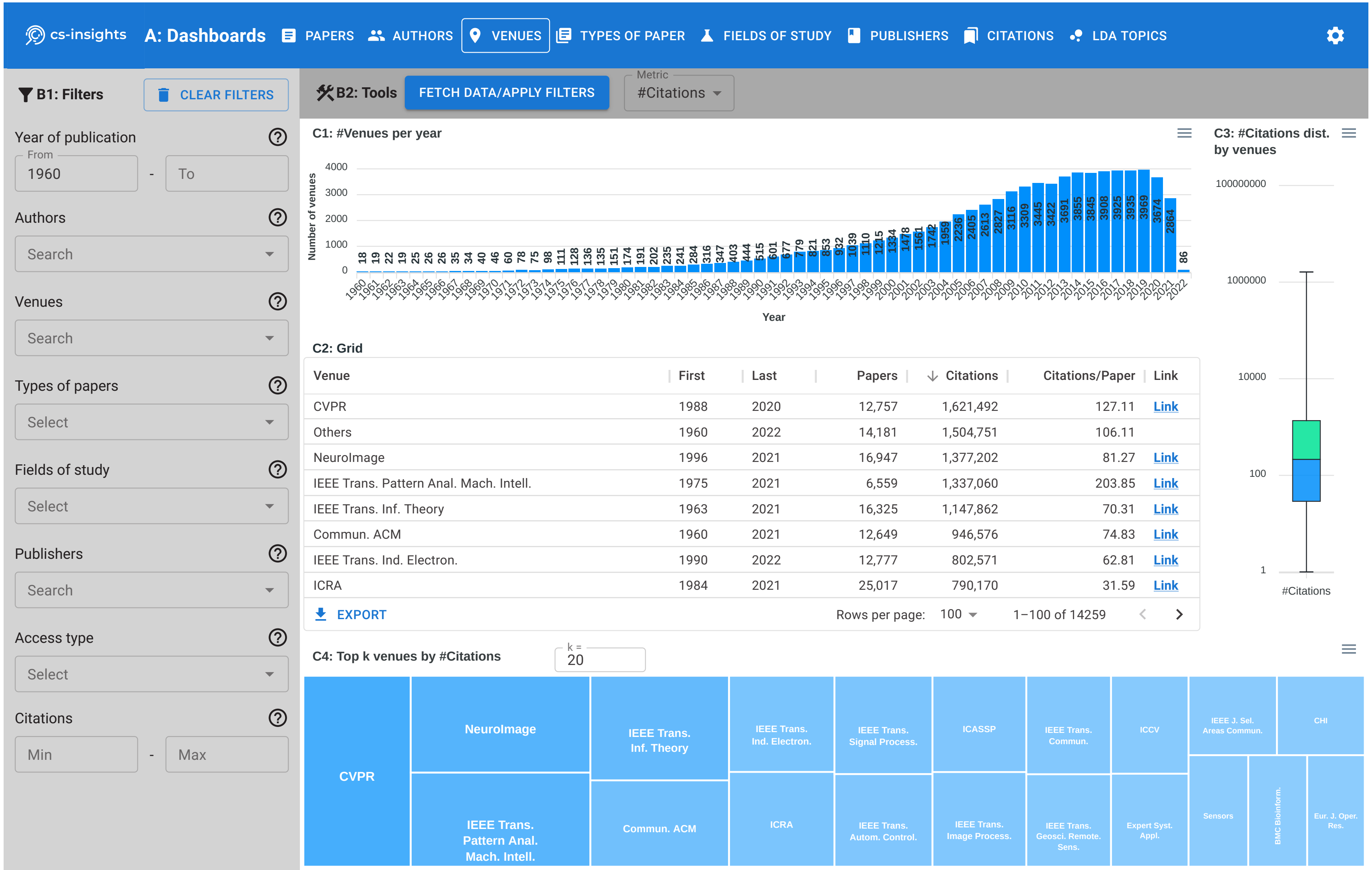


Figure A.3 Venues dashboard in CS-Insights.

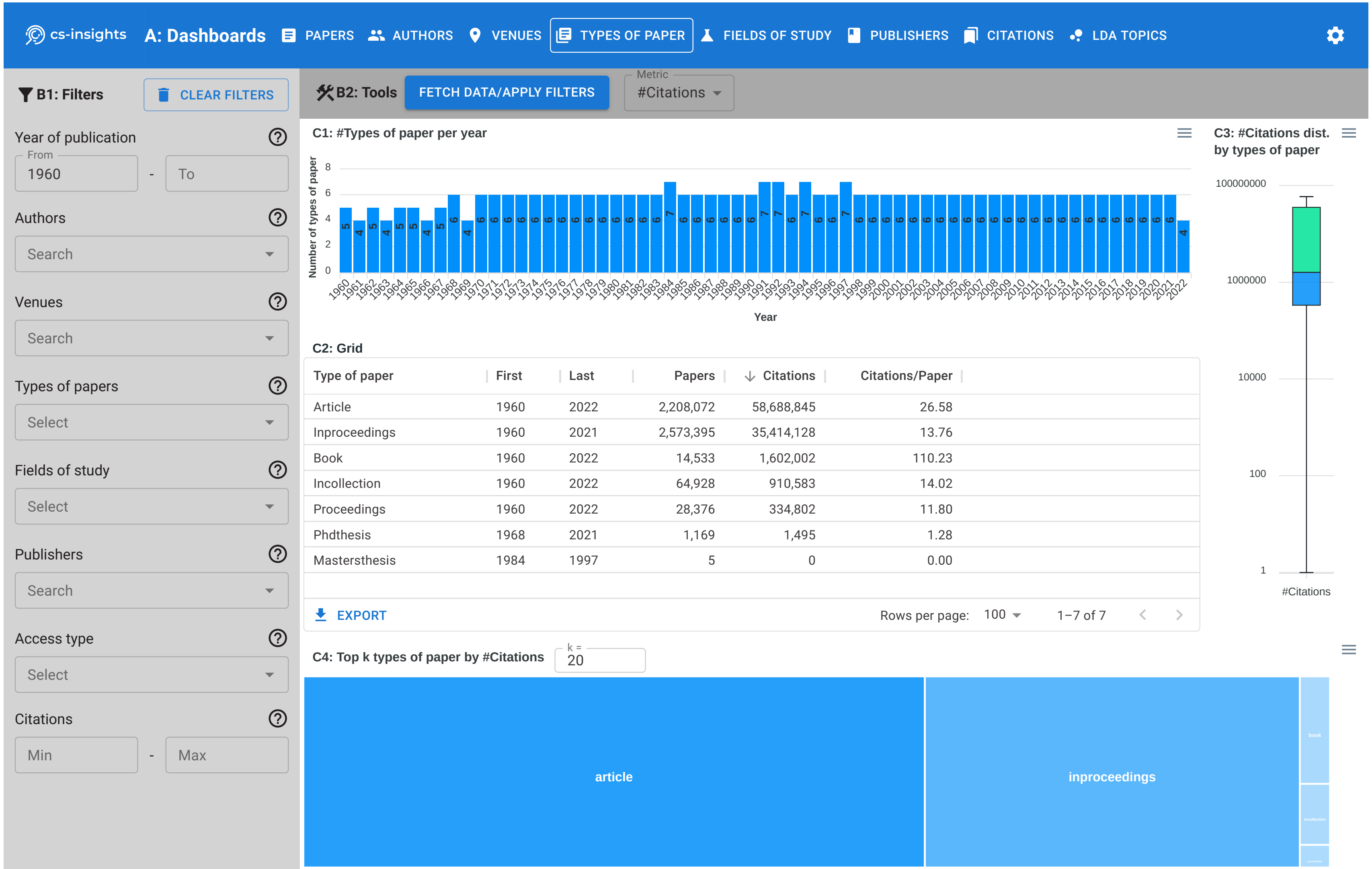


Figure A.4 Types of Paper (document types) dashboard in CS-Insights.

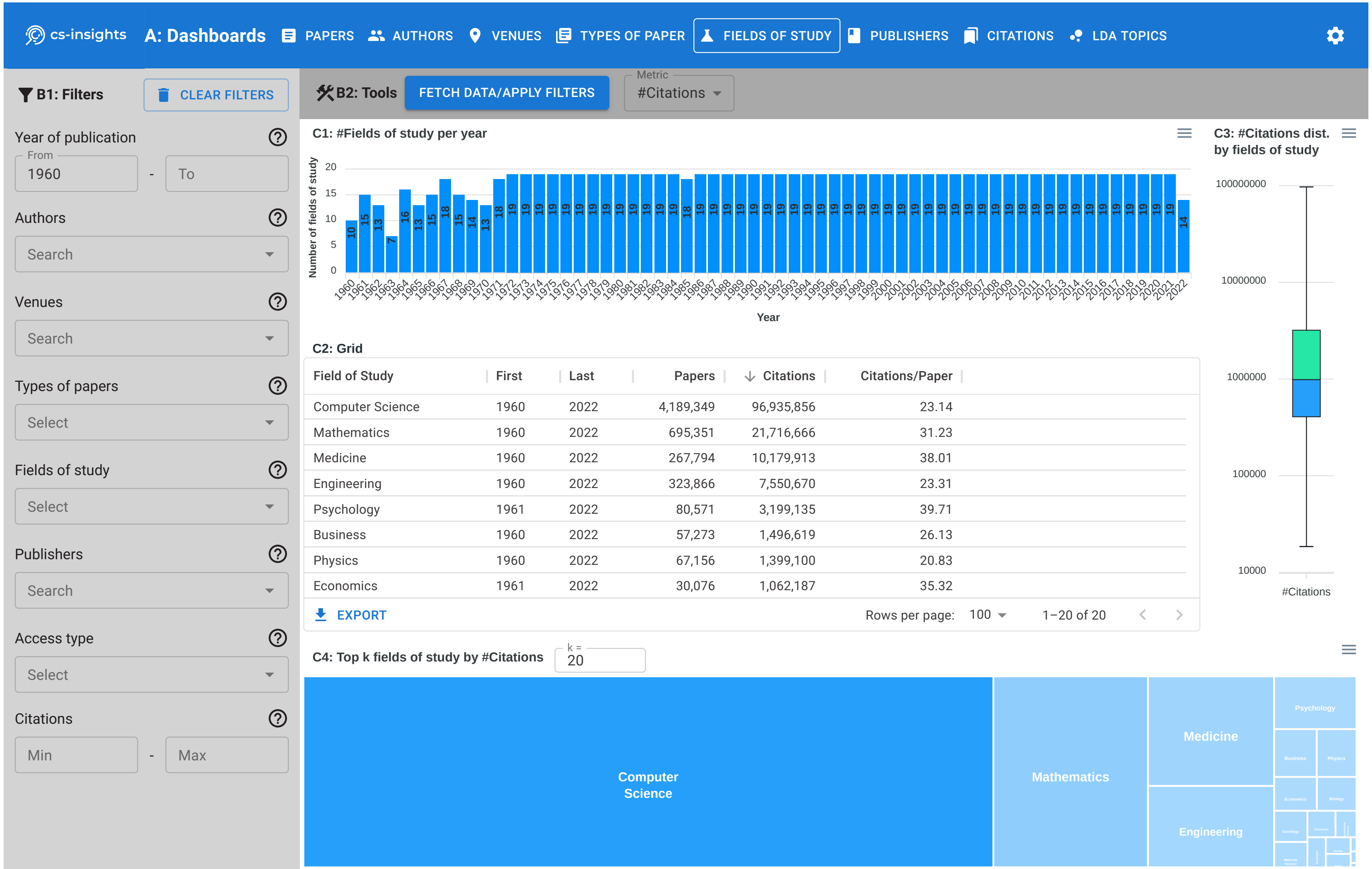


Figure A.5 Fields of Study dashboard in CS-Insights.

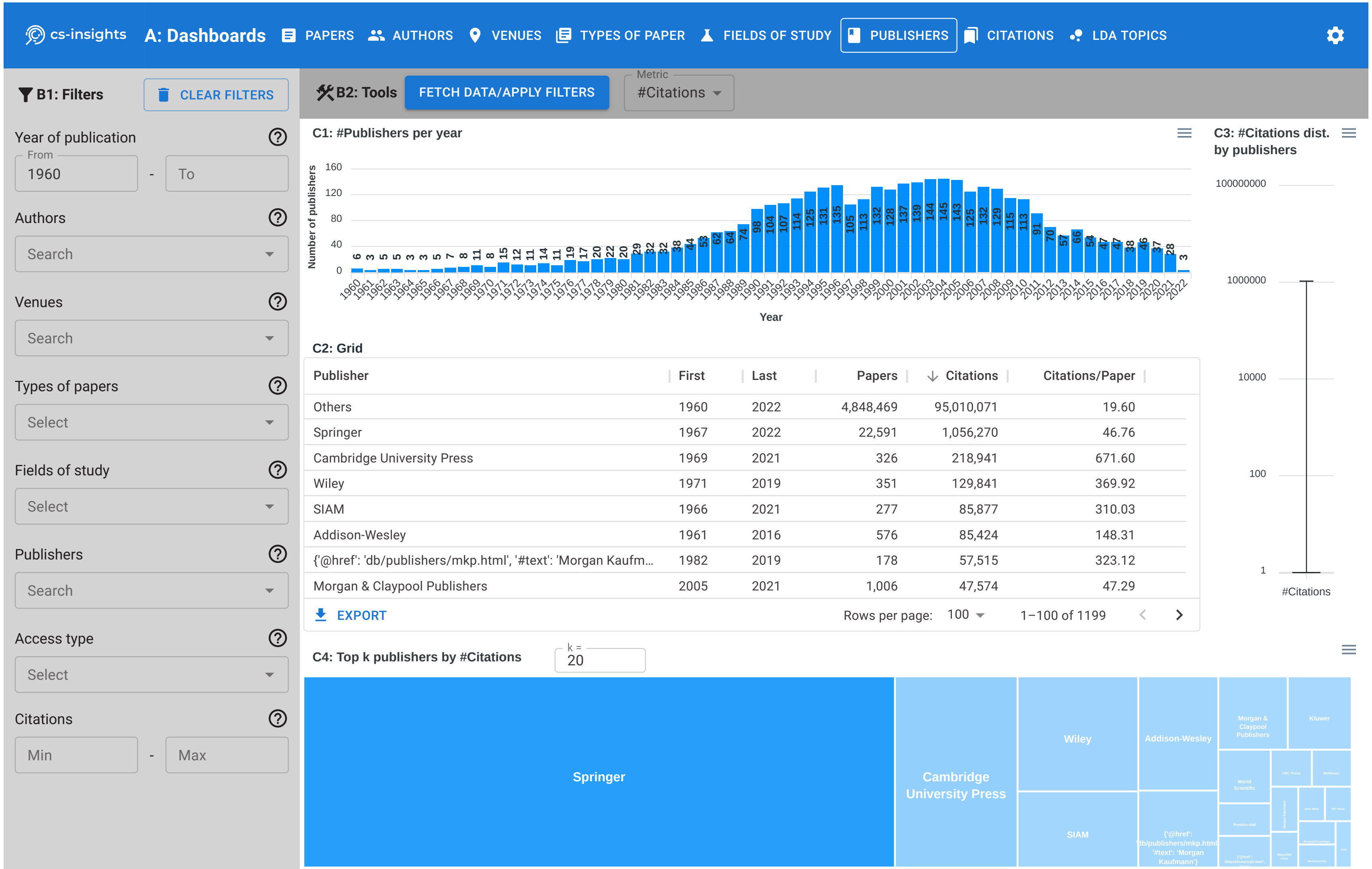


Figure A.6 Publishers dashboard in CS-Insights. The grid shows most publications have no publisher and fall under *Others*.

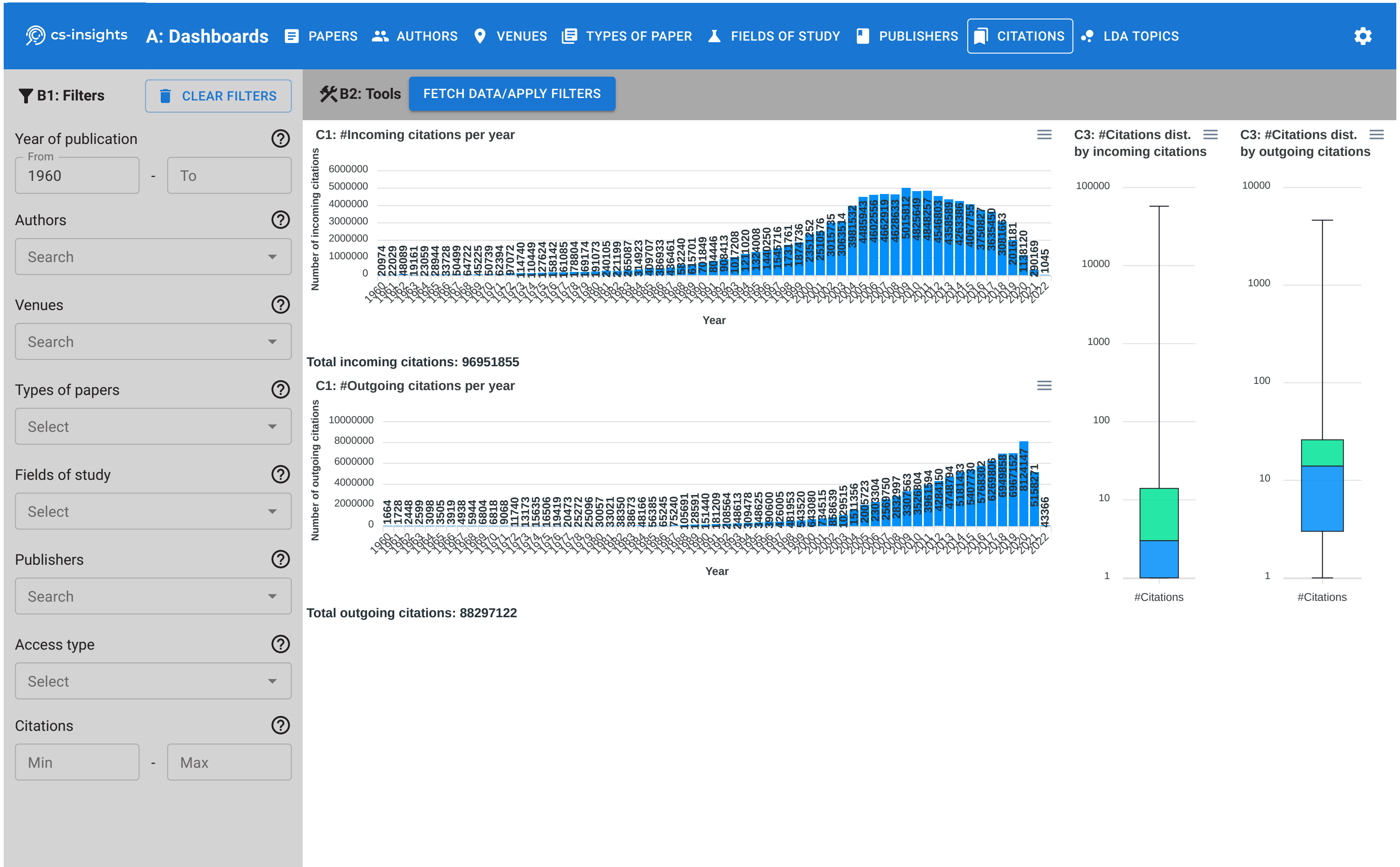


Figure A.7 Citations dashboard in CS-Insights.

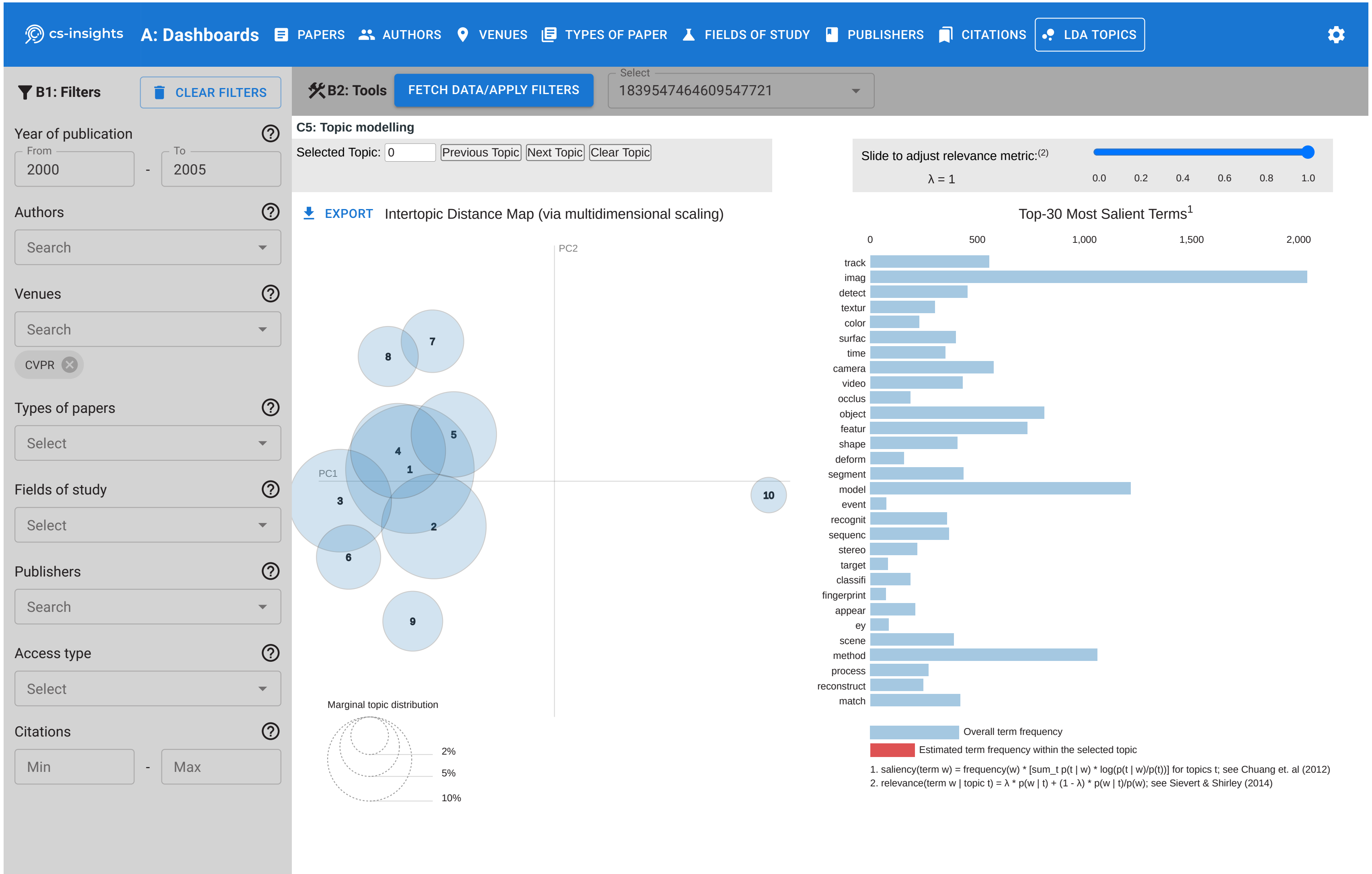


Figure A.8 Topics dashboard in CS-Insights showing a visualization of the topics for CVPR (2000-2005).

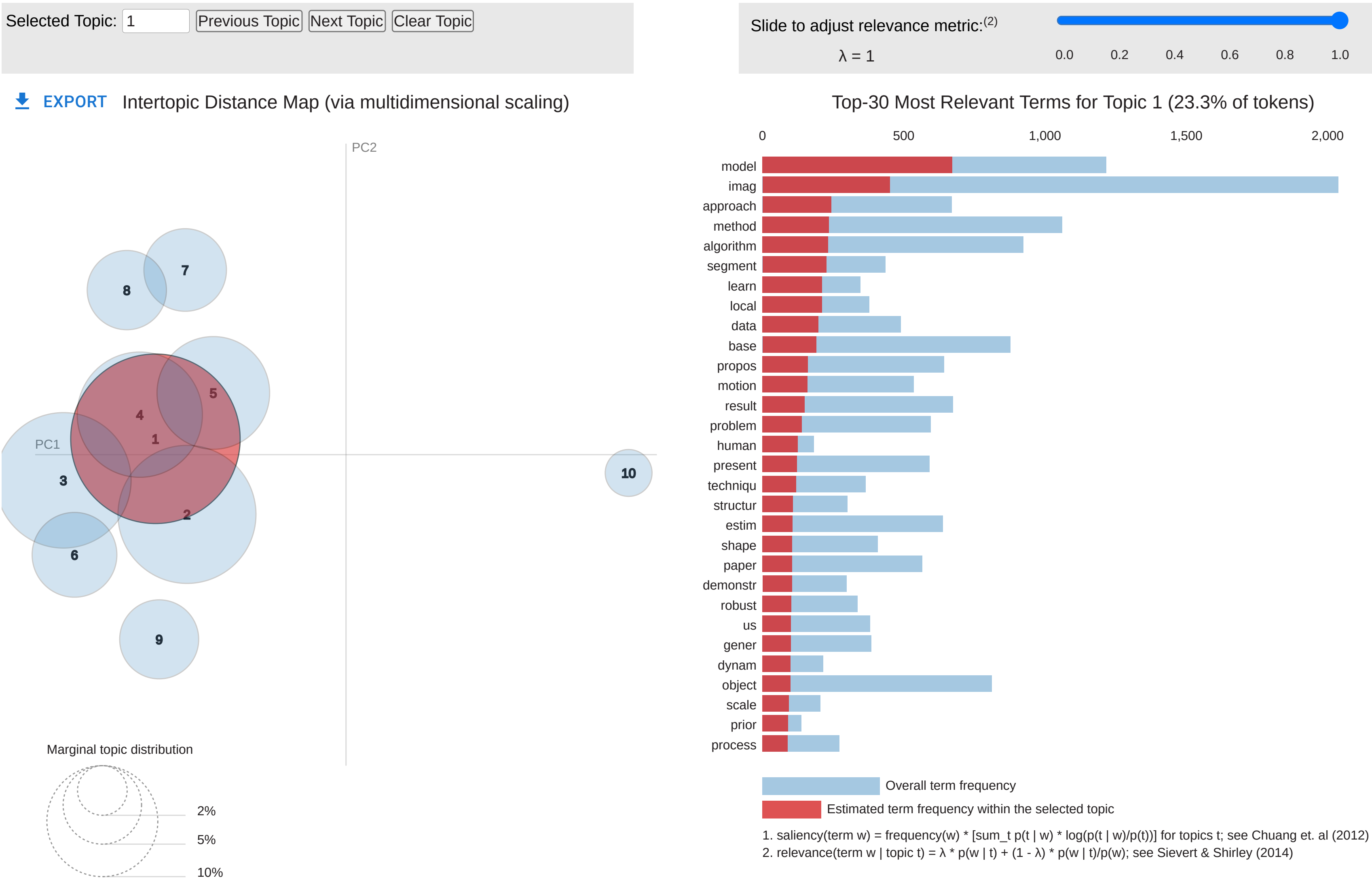


Figure A.9 Visualization of the topics for CVPR (2000-2005) with cluster 1 selected.

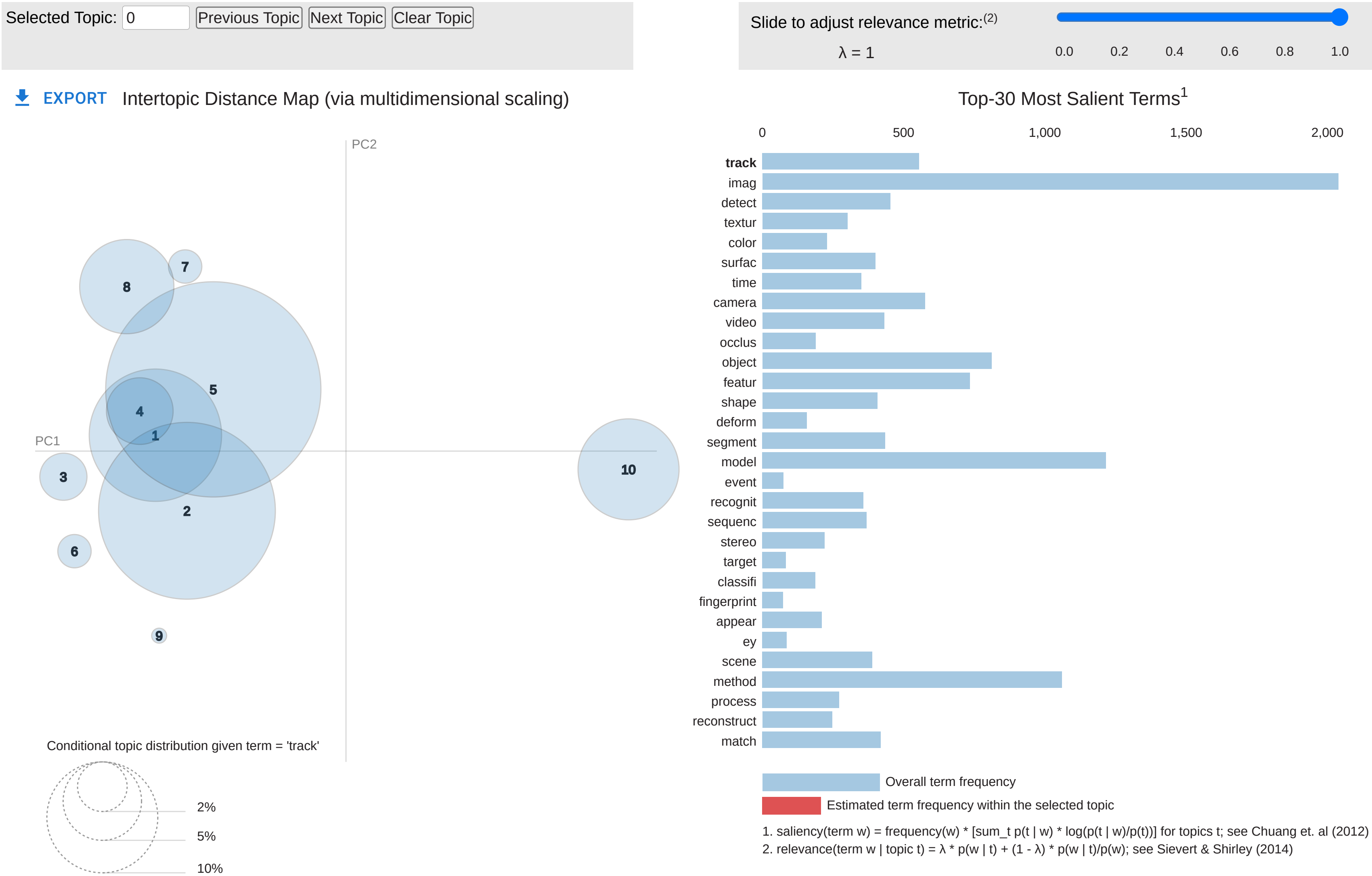


Figure A.10 Visualization of the topics for CVPR (2000-2005) with the term “track” selected.

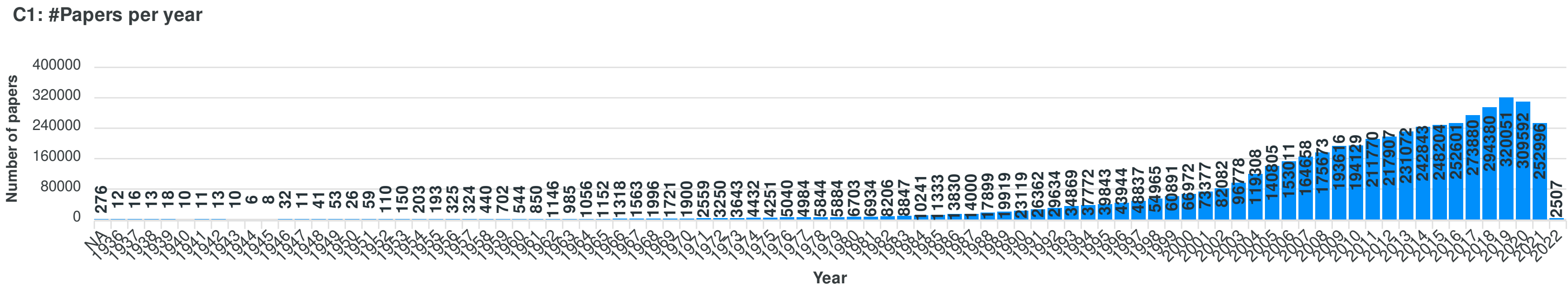


Figure A.11 Number of publications per year.

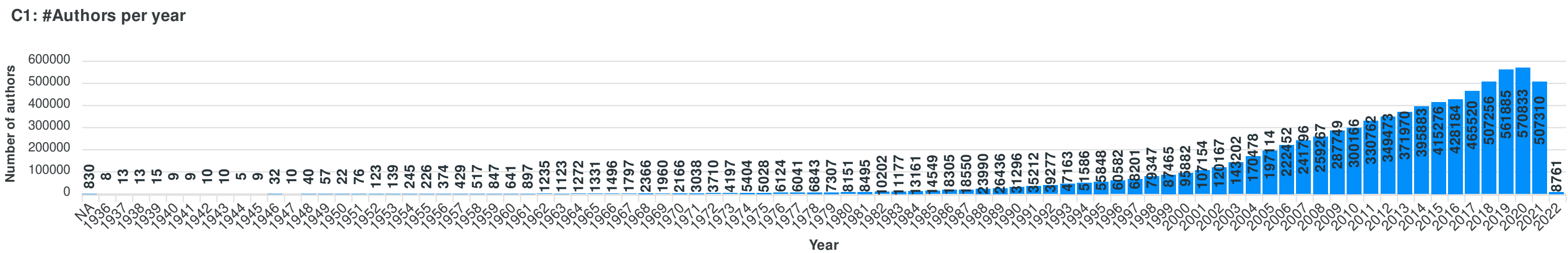


Figure A.12 Number of unique authors per year.

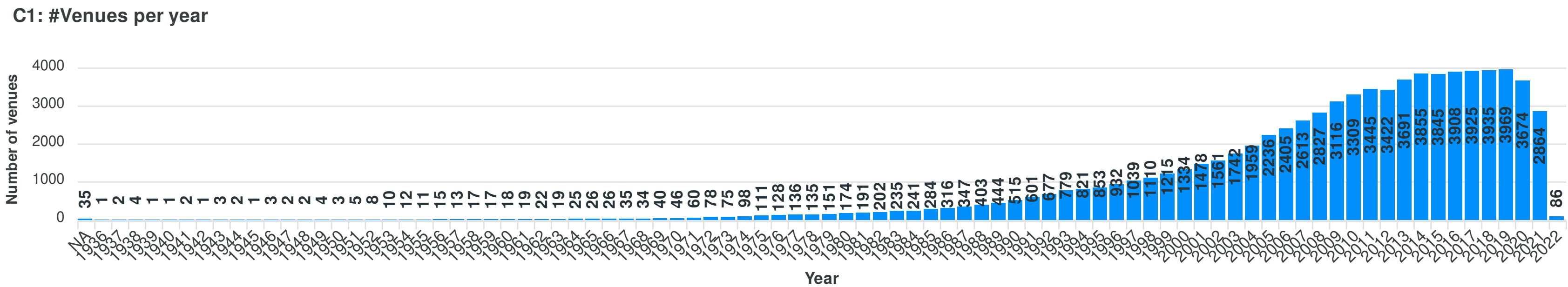


Figure A.13 Number of unique venues per year.

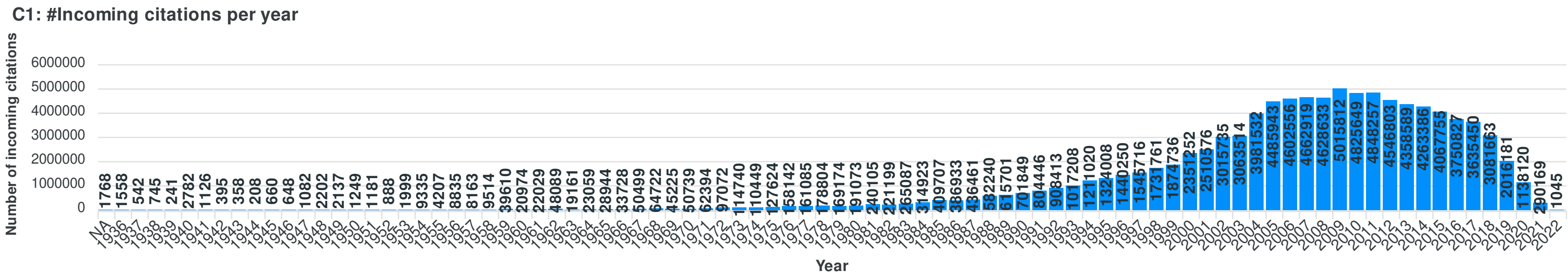


Figure A.14 Number of incoming (received) citations per year.

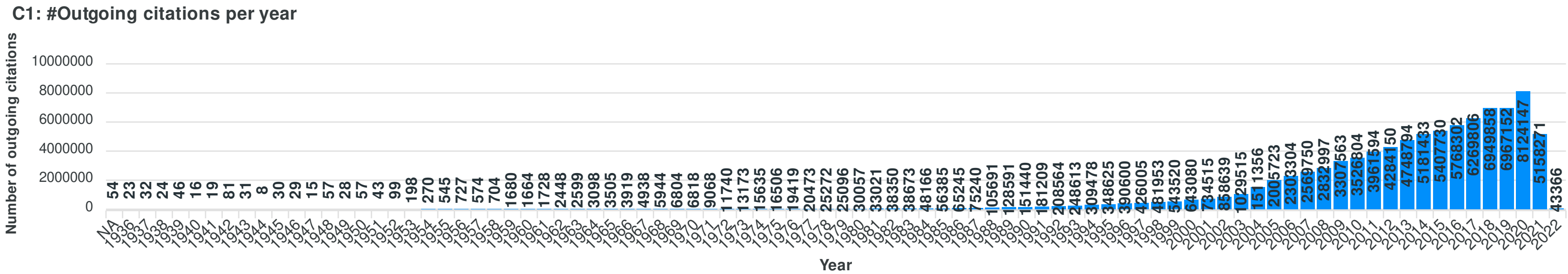


Figure A.15 Number of outgoing citations (references) per year.

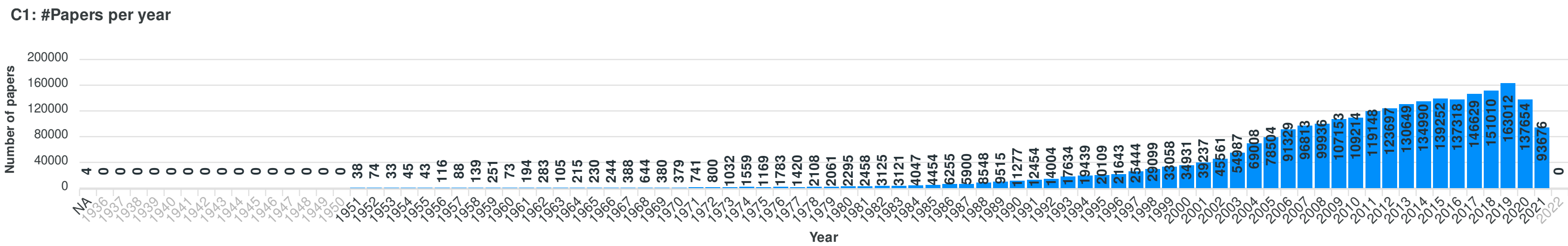


Figure A.16 Number of conference papers per year.

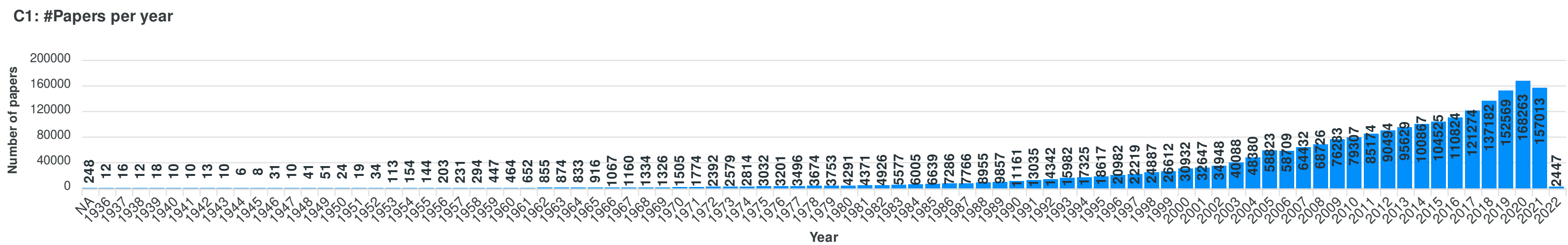


Figure A.17 Number of journal articles per year.

List of Figures

List of Tables

Abbreviations

AI Artificial Intelligence 18

API Application User Interface 4

CRUD Create, Read, Update, Delete 6

CS Computer Science VI

DOI Digital Object Identifier 21

ERA Excellence in Research for Australia 12

ERM express-restify-mongoose 27

LDA Latent Dirichlet Allocation 7

NLP Natural Language Processing 1

UI User Interface 4

Glossary

42Papers
https://42papers.com/ 18

ACL Anthology
https://aclanthology.org/ 19

ACM Digital Library
https://libraries.acm.org/digital-library 17

arXiv
https://arxiv.org/ 1

CiteSeerX
https://citeseerx.ist.psu.edu/ 17

CiteSpace
http://cluster.cis.drexel.edu/~cchen/citespace/ 19

DBLP
https://dblp.org/ VI

DRIFT
https://gchhablani-drift-app-t0asgh.streamlitapp.com/ 20

EACL
https://2023.eacl.org/calls/demos/ 28

Google Scholar
https://scholar.google.com/ 1

GROBID
https://github.com/kermitt2/grobid 23

IEEE Xplore
https://ieeexplore.ieee.org/ 17

LREC
https://lrec2022.lrec-conf.org/en/ 29

Microsoft Academic
https://www.microsoft.com/en-us/research/project/academic/ 16

Microsoft Academic Search
: `https://web.archive.org/web/20170105184616/https://academic.microsoft.com/FAQ` 12

MongoDB
: `https://www.mongodb.com/` 6

NLPExlorer
: `http://nlpexplorer.org` 20

NLP Index
: `https://index.quantumstat.com/` 18

NLP4NLP
: `http://www.nlp4nlp.org/` 14

NLP Scholar
: `http://saifmohammad.com/WebPages/nlpscholar-demo-basic.html` 1

Papers With Code
: `https://paperswithcode.com/` 18

PubMed
: `https://pubmed.ncbi.nlm.nih.gov/` 18

SciVal
: `https://www.scival.com/` 11

Scopus
: `https://www.scopus.com/` 1

Semantic Scholar
: `https://www.semanticscholar.org/` 1

VOSViewer
: `https://www.vosviewer.com/` 1

Web of Science
: `https://www.webofscience.com/` 1

Zeta Alpha
: `https://search.zeta-alpha.com/` 18

Bibliography

Aggarwal, Karan; Maad M. Mijwil; Sonia; Abdel-Hameed Al-Mistarehi; Safwan Alomari; Murat Gök; Anas M. Zein Alaabdin, and Safaa H. Abdulrhman (Jan. 30, 2022). "Has the Future Started? The Current Growth of Artificial Intelligence, Machine Learning, and Deep Learning". In: *Iraqi Journal For Computer Science and Mathematics* 3.1 (1), pp. 115–123. ISSN: 2788-7421. DOI: `10.52866/ijcsm.2022.01.01.013`. URL: `https://journal.esj.edu.iq/index.php/IJCM/article/view/100` (visited on 09/27/2022).

Ammar, Waleed; Dirk Groeneveld; Chandra Bhagavatula; Iz Beltagy; Miles Crawford; Doug Downey; Jason Dunkelberger; Ahmed Elgohary; Sergey Feldman; Vu Ha; Rodney Kinney; Sebastian Kohlmeier; Kyle Lo; Tyler Murray; Hsu-Han Ooi; Matthew Peters; Joanna Power; Sam Skjonsberg; Lucy Wang; Chris Wilhelm; Zheng Yuan; Madeleine van Zuylen, and Oren Etzioni (June 2018). "Construction of the Literature Graph in Semantic Scholar". In: *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 3 (Industry Papers)*. NAACL-HLT 2018. New Orleans - Louisiana: Association for Computational Linguistics, pp. 84–91. DOI: `10.18653/v1/N18-3011`. URL: `https://aclanthology.org/N18-3011` (visited on 09/06/2022).

Anderson, Ashton; Dan Jurafsky, and Daniel A. McFarland (July 2012). "Towards a Computational History of the ACL: 1980-2008". In: *Proceedings of the ACL-2012 Special Workshop on Rediscovering 50 Years of Discoveries*. Jeju Island, Korea: Association for Computational Linguistics, pp. 13–21. URL: `https://aclanthology.org/W12-3202` (visited on 10/10/2022).

Blei, David M.; Andrew Y. Ng, and Michael I. Jordan (2003). "Latent Dirichlet Allocation". In: *Journal of Machine Learning Research* 3 (Jan), pp. 993–1022. ISSN: ISSN 1533-7928. URL: `https://jmlr.org/papers/v3/blei03a.html` (visited on 08/26/2022).

Bornmann, Lutz; Robin Haunschild, and Rüdiger Mutz (Oct. 7, 2021). "Growth Rates of Modern Science: A Latent Piecewise Growth Curve Approach to Model Publication Numbers from Established and New Literature Databases". In: *Humanities and Social Sciences Communications* 8.1 (1), pp. 1–15. ISSN: 2662-9992. DOI: `10.1057/s41599-021-00903-w`. URL: `https://www.nature.com/articles/s41599-021-00903-w` (visited on 09/04/2022).

Bradshaw, Shannon; Eoin Brazil, and Kristina Chodorow (Dec. 9, 2019). *MongoDB: The Definitive Guide: Powerful and Scalable Data Storage*. "O'Reilly Media, Inc." 514 pp. ISBN: 978-1-4919-5443-0. Google Books: `pIrCDwAAQBAJ`.

Chen, Chaomei (Apr. 6, 2004). "Searching for Intellectual Turning Points: Progressive Knowledge Domain Visualization". In: *Proceedings of the National Academy of Sciences* 101 (suppl_1), pp. 5303–5310. DOI: `10.1073/pnas.0307513100`. URL: `https://www.pnas.org/doi/10.1073/pnas.0307513100` (visited on 09/09/2022).

— (Feb. 1, 2006). “CiteSpace II: Detecting and Visualizing Emerging Trends and Transient Patterns in Scientific Literature”. In: *Journal of the American Society for Information Science and Technology* 57.3, pp. 359–377. ISSN: 15322882, 15322890. DOI: 10.1002/asi.20317. URL: https://onlinelibrary.wiley.com/doi/10.1002/asi.20317 (visited on 09/09/2022).

Chuang, Jason; Christopher D. Manning, and Jeffrey Heer (2012). “Termite: Visualization Techniques for Assessing Textual Topic Models”. In: *International Working Conference on Advanced Visual Interfaces, AVI 2012, Capri Island, Naples, Italy, May 22-25, 2012, Proceedings*. Ed. by Tortora, Genny; Levialdi, Stefano, and Tucci, Maurizio. ACM, pp. 74–77. DOI: 10.1145/2254556.2254572.

Coşkun, Erman; Güzin Özdağoğlu; Muhammet Damar, and Büşra Alma Çallı (Apr. 11, 2019). “SCIENTOMETRICS-BASED STUDY OF COMPUTER SCIENCE AND INFORMATION SYSTEMS RESEARCH COMMUNITY MACRO LEVEL PROFILES”. In: *12th IADIS International Conference Information Systems 2019*. 12th IADIS International Conference Information Systems 2019. IADIS Press, pp. 180–188. ISBN: 978-989-8533-87-6. DOI: 10.33965/is2019_201905L023. URL: http://www.iadisportal.org/digital-library/scientometrics-based-study-of-computer-science-and-information-systems-research-community-macro-level-profiles (visited on 08/26/2022).

Dror, Rotem; Gili Baumer; Segev Shlomov, and Roi Reichart (July 2018). “The Hitchhiker’s Guide to Testing Statistical Significance in Natural Language Processing”. In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. ACL 2018. Melbourne, Australia: Association for Computational Linguistics, pp. 1383–1392. DOI: 10.18653/v1/P18-1128. URL: https://aclanthology.org/P18-1128 (visited on 09/13/2022).

Faiz, Adnan (2020). “Bibliometric Analysis of Computer Science Literature of Pakistan”. In: 8.2, p. 21.

Fiala, Dalibor and Gabriel Tutoky (Sept. 29, 2017). “Computer Science Papers in Web of Science: A Bibliometric Analysis”. In: *Publications* 5. DOI: 10.3390/publications5040023.

Fielding, Roy Thomas (2000). “In Information and Computer Science”.

Franceschet, Massimo (Dec. 1, 2010). “The Role of Conference Publications in CS”. In: *Communications of the ACM* 53.12, pp. 129–132. ISSN: 0001-0782. DOI: 10.1145/1859204.1859234. URL: https://doi.org/10.1145/1859204.1859234 (visited on 08/28/2022).

Gusenbauer, Michael (Jan. 1, 2019). “Google Scholar to Overshadow Them All? Comparing the Sizes of 12 Academic Search Engines and Bibliographic Databases”. In: *Scientometrics* 118.1, pp. 177–214. ISSN: 1588-2861. DOI: 10.1007/s11192-018-2958-5. URL: https://doi.org/10.1007/s11192-018-2958-5 (visited on 09/12/2022).

Győrödi, Cornelia; Robert Győrödi; George Pecherle, and Andrada Olah (June 2015). “A Comparative Study: MongoDB vs. MySQL”. In: *2015 13th International Conference on Engineering of Modern Electric Systems (EMES)*. 2015 13th International

Conference on Engineering of Modern Electric Systems (EMES), pp. 1–6. DOI: 10.1109/EMES.2015.7158433.
Halpern, Joseph Y. (May 1, 2000). "CoRR: A Computing Research Repository". In: *ACM Journal of Computer Documentation* 24.2, pp. 41–48. ISSN: 1527-6805. DOI: 10.1145/337271.337274. URL: https://doi.org/10.1145/337271.337274 (visited on 09/07/2022).
Kim, Jinseok (2019). "Author-Based Analysis of Conference versus Journal Publication in Computer Science". In: *Journal of the Association for Information Science and Technology* 70.1, pp. 71–82. ISSN: 2330-1643. DOI: 10.1002/asi.24079. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/asi.24079 (visited on 08/28/2022).
Kopecký, Jacek; Paul Fremantle, and Rich Boakes (Jan. 28, 2014). "A History and Future of Web APIs". In: *it - Information Technology* 56. DOI: 10.1515/itit-2013-1035.
Kousha, Kayvan and Mike Thelwall (Aug. 1, 2020). "COVID-19 Publications: Database Coverage, Citations, Readers, Tweets, News, Facebook Walls, Reddit Posts". In: *Quantitative Science Studies* 1.3, pp. 1068–1091. ISSN: 2641-3337. DOI: 10.1162/qss_a_00066. URL: https://doi.org/10.1162/qss_a_00066 (visited on 09/27/2022).
Kumari, Priti and Rajeev Kumar (2020). "Scientometric Analysis of Computer Science Publications in Journals and Conferences with Publication Patterns". In: *Journal of Scientometric Research* 9.1, p. 9.
Lavergne, Thomas; Olivier Cappé, and François Yvon (July 2010). "Practical Very Large Scale CRFs". In: *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*. ACL 2010. Uppsala, Sweden: Association for Computational Linguistics, pp. 504–513. URL: https://aclanthology.org/P10-1052 (visited on 09/13/2022).
Ley, Michael (1997). "Die Trierer Informatik-Bibliographie DBLP". In: *Informatik '97 Informatik als Innovationsmotor*. Ed. by Jarke, Matthias; Pasedach, Klaus, and Pohl, Klaus. Informatik aktuell. Berlin, Heidelberg: Springer, pp. 257–266. ISBN: 978-3-642-60831-5. DOI: 10.1007/978-3-642-60831-5_34.
— (2002). "The DBLP Computer Science Bibliography: Evolution, Research Issues, Perspectives". In: *String Processing and Information Retrieval*. Ed. by Laender, Alberto H. F. and Oliveira, Arlindo L. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, pp. 1–10. ISBN: 978-3-540-45735-0. DOI: 10.1007/3-540-45735-6_1.
— (Aug. 1, 2009). "DBLP: Some Lessons Learned". In: *Proceedings of the VLDB Endowment* 2.2, pp. 1493–1500. ISSN: 2150-8097. DOI: 10.14778/1687553.1687577. URL: https://doi.org/10.14778/1687553.1687577 (visited on 08/28/2022).
Lo, Kyle; Lucy Lu Wang; Mark Neumann; Rodney Kinney, and Daniel Weld (July 2020). "S2ORC: The Semantic Scholar Open Research Corpus". In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. ACL 2020. Online: Association for Computational Linguistics, pp. 4969–4983. DOI:

10.18653/v1/2020.acl-main.447. URL: https://aclanthology.org/2020.acl-main.447 (visited on 09/06/2022).

Lopez, Patrice (Aug. 26, 2022). *GROBID*. URL: https://github.com/kermitt2/grobid (visited on 08/26/2022).

Mariani, Joseph; Gil Francopoulo, and Patrick Paroubek (Feb. 7, 2019a). "The NLP4NLP Corpus (I): 50 Years of Publication, Collaboration and Citation in Speech and Language Processing". In: *Frontiers in Research Metrics and Analytics* 3, p. 36. ISSN: 2504-0537. DOI: 10.3389/frma.2018.00036. URL: https://www.frontiersin.org/article/10.3389/frma.2018.00036/full (visited on 09/20/2022).

Mariani, Joseph; Gil Francopoulo; Patrick Paroubek, and Frédéric Vernier (Feb. 7, 2019b). "The NLP4NLP Corpus (II): 50 Years of Research in Speech and Language Processing". In: *Frontiers in Research Metrics and Analytics* 3, p. 37. ISSN: 2504-0537. DOI: 10.3389/frma.2018.00037. URL: https://www.frontiersin.org/article/10.3389/frma.2018.00037/full (visited on 09/20/2022).

Michels, Carolin and Jun-Ying Fu (Aug. 1, 2014). "Systematic Analysis of Coverage and Usage of Conference Proceedings in Web of Science". In: *Scientometrics* 100.2, pp. 307–327. ISSN: 1588-2861. DOI: 10.1007/s11192-014-1309-4. URL: https://doi.org/10.1007/s11192-014-1309-4 (visited on 09/14/2022).

Mingers, John and Loet Leydesdorff (Oct. 1, 2015). "A Review of Theory and Practice in Scientometrics". In: *European Journal of Operational Research* 246.1, pp. 1–19. ISSN: 0377-2217. DOI: 10.1016/j.ejor.2015.04.002. URL: https://www.sciencedirect.com/science/article/pii/S037722171500274X (visited on 10/18/2022).

Mohammad, Saif M. (Nov. 8, 2019). *The State of NLP Literature: A Diachronic Analysis of the ACL Anthology*. DOI: 10.48550/arXiv.1911.03562. URL: http://arxiv.org/abs/1911.03562 (visited on 08/26/2022).

— (2020a). "Examining Citations of Natural Language Processing Literature". In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Online: Association for Computational Linguistics, pp. 5199–5209. DOI: 10.18653/v1/2020.acl-main.464. URL: https://www.aclweb.org/anthology/2020.acl-main.464 (visited on 08/26/2022).

— (May 2020b). "NLP Scholar: A Dataset for Examining the State of NLP Research". In: *Proceedings of the 12th Language Resources and Evaluation Conference*. LREC 2020. Marseille, France: European Language Resources Association, pp. 868–877. ISBN: 979-10-95546-34-4. URL: https://aclanthology.org/2020.lrec-1.109 (visited on 08/26/2022).

— (2020c). "NLP Scholar: An Interactive Visual Explorer for Natural Language Processing Literature". In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*. Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations. Online: Association for Computational Linguistics, pp. 232–255.

DOI: 10.18653/v1/2020.acl-demos.27. URL: https://www.aclweb.org/anthology/2020.acl-demos.27 (visited on 08/26/2022).

Mohr, John W. and Petko Bogdanov (Dec. 1, 2013). "Introduction—Topic Models: What They Are and Why They Matter". In: *Poetics*. Topic Models and the Cultural Sciences 41.6, pp. 545–569. ISSN: 0304-422X. DOI: 10.1016/j.poetic.2013.10.001. URL: https://www.sciencedirect.com/science/article/pii/S0304422X13000685 (visited on 10/17/2022).

Parmar, Monarch; Naman Jain; Pranjali Jain; P. Jayakrishna Sahit; Soham Pachpande; Shruti Singh, and Mayank Singh (2020). "NLPExplorer: Exploring the Universe of NLP Papers". In: *Advances in Information Retrieval*. Ed. by Jose, Joemon M.; Yilmaz, Emine; Magalhães, João; Castells, Pablo; Ferro, Nicola; Silva, Mário J., and Martins, Flávio. Lecture Notes in Computer Science. Cham: Springer International Publishing, pp. 476–480. ISBN: 978-3-030-45442-5. DOI: 10.1007/978-3-030-45442-5_61.

Radev, Dragomir R.; Pradeep Muthukrishnan, and Vahed Qazvinian (Aug. 2009). "The ACL Anthology Network". In: *Proceedings of the 2009 Workshop on Text and Citation Analysis for Scholarly Digital Libraries (NLPIR4DL)*. Suntec City, Singapore: Association for Computational Linguistics, pp. 54–61. URL: https://aclanthology.org/W09-3607 (visited on 09/13/2022).

Rahm, Erhard (Oct. 2, 2008). "Comparing the Scientific Impact of Conference and Journal Publications in Computer Science". In: *Information Services and Use* 28. DOI: 10.3233/ISU-2008-0562.

Rahm, Erhard and Andreas Thor (Dec. 1, 2005). "Citation Analysis of Database Publications". In: *ACM SIGMOD Record* 34.4, pp. 48–53. ISSN: 0163-5808. DOI: 10.1145/1107499.1107505. URL: https://doi.org/10.1145/1107499.1107505 (visited on 08/28/2022).

Ruas, Terry; Jan Philip Wahle; Lennart Küll; Saif M. Mohammad, and Bela Gipp (Oct. 13, 2022). *CS-Insights: A System for Analyzing Computer Science Research*. DOI: 10.48550/arXiv.2210.06878. URL: http://arxiv.org/abs/2210.06878 (visited on 10/18/2022). Submission planned for EACL'23 (System Demonstrations).

Sharma, Abheesht; Gunjan Chhablani; Harshit Pandey, and Rajaswa Patil (Nov. 2021). "DRIFT: A Toolkit for Diachronic Analysis of Scientific Literature". In: *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics, pp. 361–371. DOI: 10.18653/v1/2021.emnlp-demo.40. URL: https://aclanthology.org/2021.emnlp-demo.40 (visited on 08/26/2022).

Shneiderman, Ben; Catherine Plaisant; Maxine Cohen; Steven Jacobs, and Niklas Elmqvist (2018). *Designing the User Interface: Strategies for Effective Human-Computer Interaction*. Sixth edition, global edition. Boston Columbus Indianapolis New York San Francisco Hoboken Amsterdam Cape Town Dubai London Madrid Milan Munich Paris Montréal Toronto Delhi Mexico City Sao Paulo Sydney Hong Kong Seoul Singapore Taipei Tokyo: Pearson. 622 pp. ISBN: 978-1-292-15391-9.

Sievert, Carson and Kenneth Shirley (June 2014). "LDAvis: A Method for Visualizing and Interpreting Topics". In: *Proceedings of the Workshop on Interactive Language Learning, Visualization, and Interfaces*. Baltimore, Maryland, USA: Association for Computational Linguistics, pp. 63–70. DOI: 10.3115/v1/W14-3110. URL: https://aclanthology.org/W14-3110 (visited on 08/26/2022).

Spinde, Timo; Manuel Plank; Jan-David Krieger; Terry Ruas; Bela Gipp, and Akiko Aizawa (Nov. 2021). "Neural Media Bias Detection Using Distant Supervision With BABE - Bias Annotations By Experts". In: *Findings of the Association for Computational Linguistics: EMNLP 2021*. EMNLP-Findings 2021. Punta Cana, Dominican Republic: Association for Computational Linguistics, pp. 1166–1177. DOI: 10.18653/v1/2021.findings-emnlp.101. URL: https://aclanthology.org/2021.findings-emnlp.101 (visited on 09/30/2022).

Supriyadi, Edi (Aug. 29, 2022). "A Bibliometric Analysis: Computer Science Research From Indonesia". In: *TIERS Information Technology Journal* 3.1 (1), pp. 28–34. ISSN: 2723-4541. DOI: 10.38043/tiers.v3i1.3706. URL: https://journal.undiknas.ac.id/index.php/tiers/article/view/3706 (visited on 09/11/2022).

Tattershall, E.; G. Nenadic, and R. D. Stevens (Jan. 2020). "Detecting Bursty Terms in Computer Science Research". In: *Scientometrics* 122.1, pp. 681–699. ISSN: 0138-9130, 1588-2861. DOI: 10.1007/s11192-019-03307-5. URL: http://link.springer.com/10.1007/s11192-019-03307-5 (visited on 08/27/2022).

Uddin, Ashraf; Vivek Kumar Singh; David Pinto, and Ivan Olmos (Oct. 1, 2015). "Scientometric Mapping of Computer Science Research in Mexico". In: *Scientometrics* 105.1, pp. 97–114. ISSN: 0138-9130. DOI: 10.1007/s11192-015-1654-y. URL: https://doi.org/10.1007/s11192-015-1654-y (visited on 08/27/2022).

Van Eck, Nees Jan and Ludo Waltman (Aug. 1, 2010). "Software Survey: VOSviewer, a Computer Program for Bibliometric Mapping". In: *Scientometrics* 84.2, pp. 523–538. ISSN: 1588-2861. DOI: 10.1007/s11192-009-0146-3. URL: https://doi.org/10.1007/s11192-009-0146-3 (visited on 09/10/2022).

Vardi, Moshe (May 1, 2009). "Conferences vs. Journals in Computing Research". In: *Commun. ACM* 52, p. 5. DOI: 10.1145/1506409.1506410.

Vrettas, George and Mark Sanderson (May 1, 2015). "Conferences versus Journals in Computer Science: Conferences vs. Journals in Computer Science". In: *Journal of the Association for Information Science and Technology* 66. DOI: 10.1002/asi.23349.

Wahle, Jan Philip; Terry Ruas; Norman Meuschke, and Bela Gipp (Sept. 2021). "Are Neural Language Models Good Plagiarists? A Benchmark for Neural Paraphrase Detection". In: *2021 ACM/IEEE Joint Conference on Digital Libraries (JCDL)*. 2021 ACM/IEEE Joint Conference on Digital Libraries (JCDL), pp. 226–229. DOI: 10.1109/JCDL52503.2021.00065.

Wahle, Jan Philip; Terry Ruas; Saif Mohammad, and Bela Gipp (June 2022). "D3: A Massive Dataset of Scholarly Metadata for Analyzing the State of Computer Science Research". In: *Proceedings of the Language Resources and Evaluation Conference*. Marseille, France: European Language Resources Association, pp. 2642–2651. URL: https://aclanthology.org/2022.lrec-1.283.

Xia, Wanjun; Yanping Jiang; Weifeng Zhu; Shuang Zhang, and Tianrui Li (May 8, 2021). “Research Fronts of Computer Science: A Scientometric Analysis”. In: *Journal of Scientometric Research* 10.1, pp. 18–26. ISSN: 23216654, 23200057. DOI: `10.5530/jscires.10.1.3`. URL: `http://www.jscires.org/article/397` (visited on 09/11/2022).

Zurita, Gustavo; Jose M. Merigo; Valeria Lobos, and Carles Forteza (Feb. 3, 2020). “Bibliometrics in Computer Science: An Institution Ranking”. In: *Journal of Intelligent & Fuzzy Systems* 38, pp. 1–13. DOI: `10.3233/JIFS-179636`.

arXiv (Jan. 3, 2022). *arXiv Submission Rate Statistics | arXiv E-Print Repository*. URL: `https://arxiv.org/help/stats/2021_by_area` (visited on 09/04/2022).

Šubelj, Lovro and Dalibor Fiala (Apr. 1, 2017). “Publication Boost in Web of Science Journals and Its Effect on Citation Distributions”. In: *Journal of the Association for Information Science and Technology* 68, pp. 1018–1023. DOI: `10.1002/asi.23718`.