

Analysis of the Accuracy and Completeness of SINTA Author Data Extraction

Muhammad Arfah Asis^{1✉}, Nia Kurniati², Muhammad Alfarid Jufda³

^{1,2,3} Informatics Engineering, Faculty of Computer Science, Universitas Muslim Indonesia, Indonesia

✉ **Corresponding Author** : Muhammad Arfah Asis (e-mail: muhammad.arfah.asis@umi.ac.id)

Article Information	ABSTRACT
Article History Received : December 12, 2025 Revised : December 23, 2025 Published : January 16, 2026	The advancement of information technology has increased the use of web scraping for scientific data collection, including from the SINTA (Science and Technology Index) platform, which provides researcher profiles, affiliations, publications, and citation data. However, scraping SINTA poses challenges, particularly when multiple authors share identical scores that trigger changes in display order. This instability can lead to duplicated or missing entries when using a single-pass scraping approach. This study evaluates the accuracy and completeness of SINTA author data collection by implementing repeated scraping as a strategy to handle dynamic data ordering. Experiments were conducted on the Universitas Muslim Indonesia (UMI) affiliation, targeting 915 active authors. The methodology involved page-structure analysis, spider development using Python and Scrapy, sequential scraping through pagination, and validation of data completeness and uniqueness. A three-second delay between requests was applied to maintain responsible scraping practices. The results show that a single scraping attempt failed to retrieve all authors, capturing an average of only 877.2 authors (95.86%). Due to unstable ordering, repeated iterations were required. Through 4–8 scraping cycles per trial, all 915 authors were successfully collected without duplication. These findings indicate that for platforms with dynamic data structures such as SINTA, repeated scraping provides a more reliable method for ensuring data completeness and accuracy, supporting the development of stable and responsible publication-data automation systems.
Keywords: <i>Data completeness;</i> <i>Dynamic data ordering;</i> <i>Python;</i> <i>SINTA;</i> <i>Web scraping;</i>	

INTRODUCTION

Advancements in information technology and the increasing availability of digital data have transformed how researchers gather, analyze, and utilize scientific information. One technique that has become increasingly important in data-driven research is web scraping, an automated process of extracting data from websites using programming algorithms (Yuda, 2025). This technique is particularly relevant for researchers in Indonesia who rely on data from SINTA (Science and Technology Index), a national platform that provides profiles of researchers, affiliations, publications, and scientific citations.

Beyond its role in research, accurate and complete SINTA author data is a critical foundation for academic information systems in higher education. Many institutional platforms, including academic information systems (Lengkong et al., 2023), accreditation management (Asis et al., 2024), tracer studies (Kurniatuhadi et al., 2024), and lecturer performance evaluation systems, depend on reliable lecturer and publication profiles for institutional reporting and decision-making. Inaccurate or incomplete author data may lead to reporting errors, misinterpretation in accreditation assessments, and incorrect evaluation of lecturer performance. As SINTA functions as the national index of scientific output in Indonesia, automated author-level data retrieval is essential to ensure data consistency and reduce reliance on manual updates.

Therefore, obtaining accurate author information, particularly lecturer SINTA IDs, is a fundamental prerequisite before publication data can be integrated into interconnected academic systems.

In practice, obtaining data from SINTA presents several technical and structural challenges. Before publication data can be analyzed, researchers must first retrieve author information based on institutional affiliation to identify each lecturer's SINTA ID. This step is crucial as it forms the foundation for subsequent processes such as data enrichment and bibliometric analysis. However, automated extraction is not always straightforward because SINTA employs dynamic page structures and request-limiting mechanisms that can affect the efficiency of scraping processes.

Previous studies have shown that web scraping is widely used to collect scientific data from various platforms. Several investigations highlight that web scraping is effective for assessing the quality of publication data, although its success is strongly influenced by website structure (Purnomo & Ayub, 2021). Other studies make use of SINTA data to analyze the alignment of university research themes (Aidillah et al., 2025), while different research develops business-intelligence models based on scraped data (Hidayat et al., 2025). Additional work leverages SINTA's online data to support author productivity weighting (Rochim et al., 2021).

Furthermore, additional studies explore the use of scraping in health information systems as part of national data integration efforts (Madjido et al., 2019). Research focusing on automated retrieval of health research data (Sahria, 2020) and the use of scraping to accelerate article discovery on journal websites has also been conducted (Ulfah & Najiah, 2023). Other investigations emphasize the significant role of scraping in collecting scientific data from online platforms (Adila, 2022; Saniyah & Ghozali, 2023).

In the academic context, earlier studies demonstrate the benefits of scraping for multidisciplinary data collection and the development of scientific article recommendation systems (Guyt et al., 2024; Lotfi et al., 2021). Subsequent research shows that integrating scraping with data-mining techniques enables broader analyses of publication trends (Giuliana et al., 2024; Rahmatulloh & Gunawan, 2020). From a technical perspective, several studies discuss performance and efficiency challenges in scraping, including rate-limiting restrictions and timeout issues in large-scale data retrieval (Fahrudin et al., 2025; Goel et al., 2024). Other research notes that asynchronous methods such as AsyncIO can accelerate the process but may reduce stability under high simultaneous request loads (Zbarcea & Tudose, 2024). Additional studies emphasize the importance of balancing speed and reliability, especially when dealing with complex website structures (Shah et al., 2025). Beyond technical factors, ethical and sustainability considerations are also highlighted in recent work. Prior studies stress the importance of responsible scraping practices to avoid disrupting source-website stability (Kazmali & Sayar, 2025), while other research recommends conducting gradual testing before performing scraping at a large scale (Logos et al., 2023).

Overall, existing studies demonstrate that web scraping has been extensively applied to collect and utilize scientific data for bibliometric analysis, business intelligence, productivity assessment, and the development of academic information systems. Most prior research focuses on how scraped data is used for analysis, integration, and decision support. However, limited attention has been given to evaluating the accuracy and completeness of author data obtained through scraping, particularly from the SINTA platform. Given that SINTA data increasingly serves as a foundation for academic reporting, accreditation processes, and lecturer performance evaluation, the lack of systematic validation of scraped author data represents a critical research gap. This study addresses this gap by examining the reliability and completeness of author-level data collected from SINTA using web-scraping techniques.

The primary objective of this study is to analyze the accuracy and completeness of SINTA author data extraction based on institutional affiliation. This study evaluates how dynamic author ordering affects the reliability of single-pass scraping and examines the effectiveness of repeated scraping in minimizing missing and duplicated data. The findings are expected to provide

empirical evidence for improving automated author-data retrieval processes and supporting reliable publication mapping, lecturer productivity analysis, and institutional reporting systems.

RESEARCH METHODS

This study is an experiment designed to collect author data based on institutional affiliation from the SINTA (Science and Technology Index) platform. The main objective is to retrieve all author records without duplication and without missing entries. The method employed is sequential scraping, a scraping approach executed in a strictly ordered and non-parallel manner (Darmawan et al., 2022). This method was chosen because it is safer, reduces the load on the server, and aligns with responsible scraping principles. The scraping process was conducted using Python with the Scrapy library.

The research object is the authors list page from one of SINTA's official affiliations, Universitas Muslim Indonesia (UMI), accessible at:

<https://sinta.kemdiktisaintek.go.id/affiliations/authors/2528>.

The primary data collected include author names and SINTA IDs, which serve as unique identifiers for each researcher. The target dataset consists of 915 researchers, corresponding to the number of active authors listed on the page at the time of data retrieval (14 November 2025). The scraping experiment was executed in 10 separate sessions. All sessions were conducted in a controlled environment using Windows 11, Python 3.13.3, and a stable 100 Mbps network connection to ensure consistent experimental conditions. The research procedure consists of four main stages: page structure analysis, scraping module design, sequential scraping execution, and validation and evaluation of the results. The stages of the research are illustrated in Figure 1.

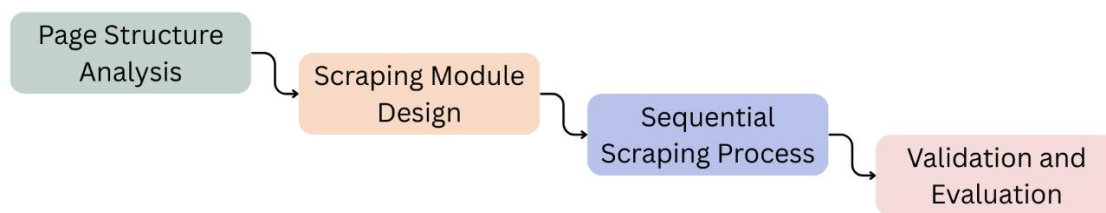


Figure 1. Research Workflow

Page Structure Analysis

The first stage involved analyzing the HTML structure of the authors list page. Key elements such as author names, departments, SINTA IDs, SINTA scores, and Scopus and Google Scholar h-index values were identified based on consistent HTML tags and class attributes. In addition, the pagination mechanism and the URL patterns between pages were examined to ensure that all data could be accessed by the scraper. This analysis formed the foundation for designing the scraping module using Python and Scrapy.

Scraping Module Design

The scraping module was developed based on the page structure analysis. The Scrapy library was used to create a spider that extracts HTML elements using CSS selectors and XPath (Darmawan et al., 2022). The module was designed to automatically navigate through all pages via the platform's pagination mechanism and to store extracted data in a temporary structure. This structure was used for filtering and further inspection during the experimental process.

Sequential Scraping Process

The scraping process was executed sequentially on each page, where each page contains 10 authors. Every session adhered to responsible scraping principles by introducing a minimum delay of three seconds between requests to reduce server load (Molina et al., 2025). The scraper also used an appropriate user-agent to avoid being detected as a bot. The general workflow of the sequential scraping process is shown in Figure 1.

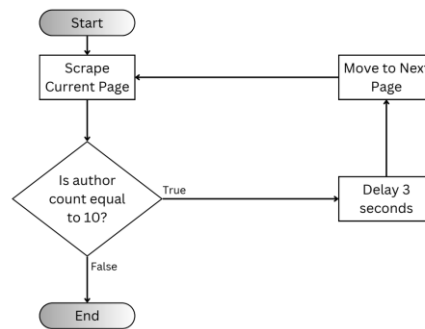


Figure 1. Sequential Scraping Workflow

Validation and Evaluation

After all scraping sessions were completed, the results were validated by comparing sampled scraped data with the SINTA webpage manually. This validation ensured that no data were missing and that all entries were unique and free of duplication. The effectiveness of the method was assessed by comparing the outcomes of a conventional single-run scraping process with those of an iterative, repeated scraping approach. The evaluation considered several parameters, including the total amount of data successfully collected, the presence of duplicate entries, and the number of pages successfully extracted. A checkpoint recovery mechanism was also implemented to ensure that the process could continue seamlessly in the event of a network interruption.

RESULTS AND DISCUSSION

This study successfully collected all author data from a single SINTA affiliation. The first stage involved analyzing the HTML structure of the authors list page. Various elements within each author entry could be extracted using different selection techniques. The list of extracted elements along with their extraction methods is presented in Table 1.

Table 1. Author Data Elements and Extraction Techniques (Scrapy)

No	Field Name	Selector / Technique (Scrapy)	Description
1	name	author_item.css("div.profile-name a::text").get("").strip()	Extracts the author's name from the <a> element inside div.profile-name. Returns an empty string if unavailable.
2	department	author_item.css("div.profile-dept a::text").get("").strip()	Retrieves the department or specific affiliation (e.g., "Teknik Informatika") from the <a> element in div.profile-dept.
3	sinta_id	Extract all text → author_item.css("div.profile-id::text").getall(), then filter strings containing 'ID :'	Extracts the SINTA ID from the raw text in div.profile-id by locating the line that contains "ID :".
4	scopus_h_index	Extract all text → author_item.css("div.profile-hindex span::text").getall(), then locate 'Scopus H-Index'	Extracts the Scopus H-Index by parsing the number from text such as "Scopus H-Index : 4". Defaults to 0 if parsing fails.
5	gs_h_index	Extract all text → author_item.css("div.profile-hindex span::text").getall(), then locate 'GS H-Index'	Retrieves the Google Scholar H-Index from the relevant element in profile-hindex. Defaults to 0 if not found.

No	Field Name	Selector / Technique (Scrapy)	Description
6	gs_i10_index	Hardcoded value = 0 (not available on the author list page)	The field is set to 0 because the SINTA author list does not display the i10-index.
7	sinta_score_3yr	Extract numbers from div.stat-num::text + div.stat-text::text when the text contains 'SINTA Score 3Yr'	Extracts the 3-year SINTA Score using text-number pairs. Commas and periods are removed before converting the value to an integer.
8	sinta_score_overall	Extract numbers from div.stat-num::text + div.stat-text::text when the text contains 'SINTA Score' (without 3Yr)	Retrieves the overall SINTA Score using the corresponding text-number pairs.

From the table, it can be seen that eight data elements were collected for each author. By default, the SINTA author list is displayed based on the 3-year SINTA Score, sorted from highest to lowest. However, observations showed that when multiple authors have the same 3-year score, their order becomes inconsistent or random. Similar behavior also appears when the sorting is based on Scopus H-index, Google Scholar H-index, or overall SINTA Score. An example of the author page ordered by Scopus h-index is shown in Figure 2.

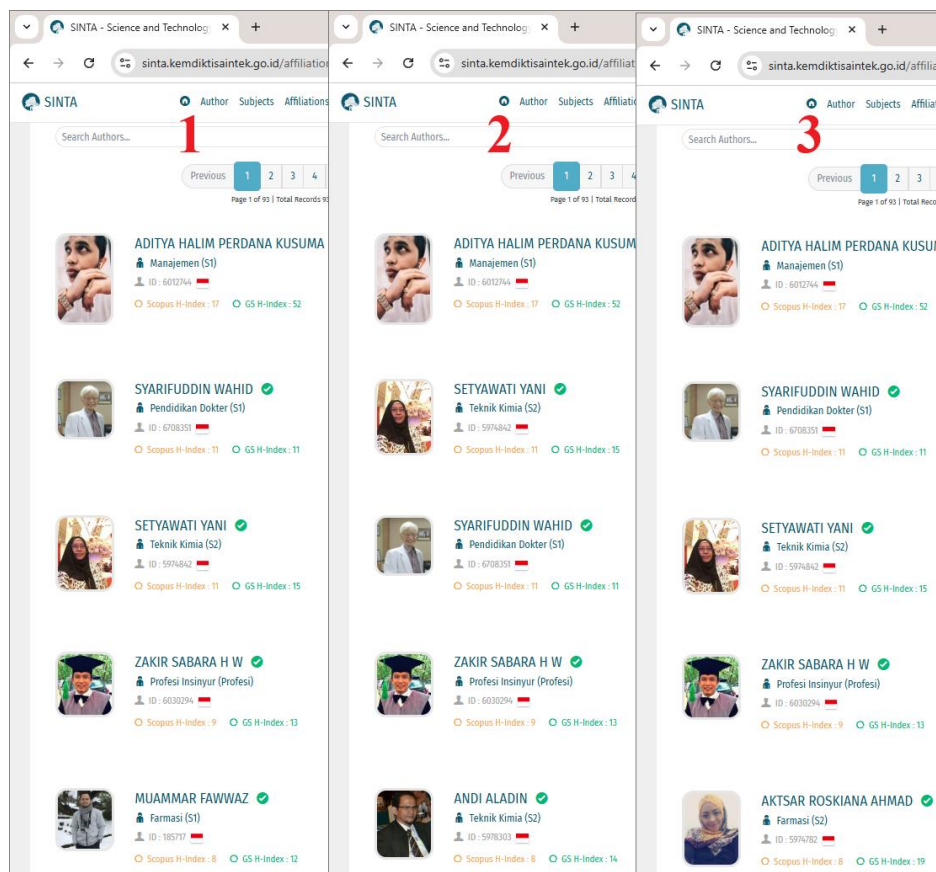


Figure 2. Three different access results of the SINTA author page ordered by Scopus h-index

Source: <https://sinta.kemdiktisaintek.go.id/affiliations/authors/2528?page=1>

Accessed: January 2, 2026

In Figure 2, three different access results of the author page are shown, each displaying different author data and ordering. Each access presents five author entries. The first position remains unchanged because only one author has a Scopus h-index of 17. The second and third positions alternate because both authors have the same Scopus h-index of 11. Similarly, the fourth

position remains stable, as only one author has a Scopus h-index of 9. In contrast, the fifth position may vary among three different authors, as three authors share the same Scopus h-index of 8.

These inconsistencies cause authors with identical scores to appear on different pages across scraping sessions, which introduces two risks: certain entries may appear more than once as duplicates, and certain entries may not appear at all in a single scraping run. Because of this, repeated scraping sessions are necessary to ensure that the complete dataset is captured accurately without duplication. After the scraping module was designed, the workflow was visualized in a flowchart, as shown in Figure 3.

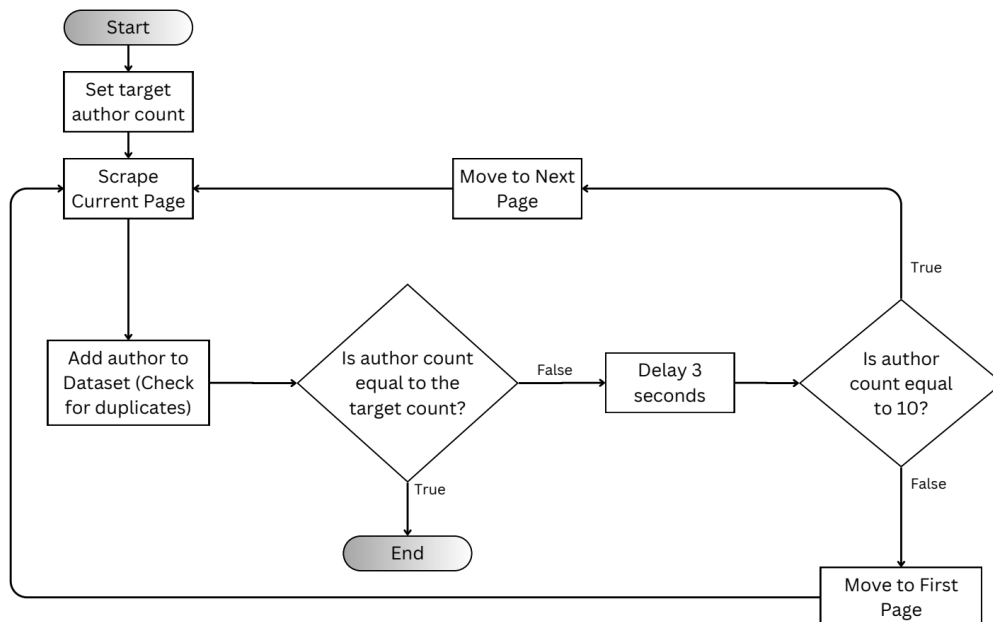


Figure 3. Scraping Process Flowchart

The flowchart illustrates that each iteration includes a check for duplication and data completeness. Scraping proceeds to the next page if the page contains 10 records. If the number of records is fewer than 10, the process returns to the first page and the scraping is repeated. Once all data (915 authors) have been collected and no duplication is detected, the process is terminated. After implementing the program using Python and Scrapy, the scraping was executed across ten trials. The results of each trial are presented in Table 2.

Table 2. Number of Author Records Successfully Collected per Scraping Iteration

Trial	Iteration							
	1	2	3	4	5	6	7	8
1	871	904	914	915				
2	880	909	914	914	915			
3	869	903	912	915				
4	878	910	914	915				
5	880	905	912	915				
6	880	906	910	914	915			
7	878	899	908	912	913	914	914	915
8	874	897	910	914	915			
9	884	910	914	914	915			
10	878	905	911	913	915			

During the first iteration of each trial, the number of collected records did not yet reach 915. The average result of the first scraping iteration was only 877.2 records, or 95.86% of the total targeted data. This finding confirms that a single data collection is insufficient to obtain all author data from SINTA. Based on our observations, there are 43 score variations that appear

inconsistently across two or more pages. This condition causes the single scraping approach to fail in collecting 100% of the data, as some records may appear more than once, while others do not appear at all. In addition, 79 records with a score of 0 were found to appear inconsistently across nine pages, namely pages 84 to 92, which further reduces the likelihood of obtaining complete data through a single data collection. Therefore, repeated data collection is proven to be more effective in obtaining a complete and consistent dataset from the SINTA website.

Based on Table 2, it took between 4 and 8 iterations to reach the maximum total of 915 records. A visualization of the scraping results is shown in Figure 4.

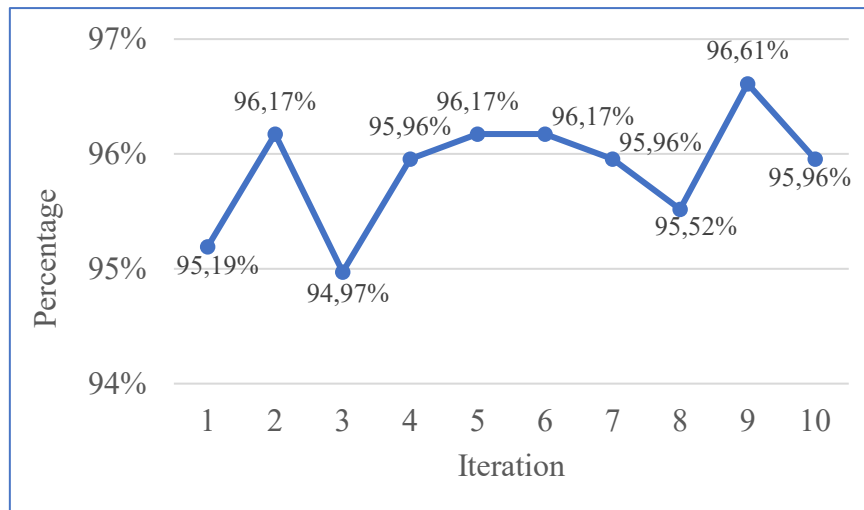


Figure 4. Percentage of Successfully Collected Data per Iteration

The figure shows that the highest values obtained from a single scraping attempt were only 96.61% and 94.97%. This indicates that a single scraping pass cannot be relied upon to retrieve complete data. After the scraping process was repeated multiple times, all data were successfully collected and validated. The verification results confirmed that the 915 author entries obtained did not contain any duplicates.

This study yields an important finding: although several authors share identical scores, in practice they still appear on only one page once the full scraping process is successfully completed. However, when more authors with identical scores are distributed across multiple pages, the likelihood of duplication or missing data in a single scraping attempt increases. Because the ordering of displayed data is random, the number of iterations required to collect the complete dataset cannot be determined with certainty.

CONCLUSION

The findings of this study indicate that collecting author data from SINTA cannot rely on a single scraping attempt due to variations in data ordering that change with each access. Although SINTA pages display authors based on specific scoring criteria, the ordering becomes random when multiple authors share identical scores, causing some entries to shift across pages, appear more than once, or fail to appear entirely. This condition results in a single-run scraping success rate of only about 95–96% of the total 915 entries. Sequential scraping with repeated iterations proved effective in achieving complete data collection, with ten trials showing that four to eight iterations were needed to gather all entries without duplication. These results demonstrate that iterative scraping is necessary for platforms with dynamic ranking systems to ensure data completeness and accuracy, and that sequential scraping combined with per-iteration validation offers a stable and responsible approach. This outcome provides a foundation for designing more reliable publication data collection systems, particularly for applications such as faculty productivity mapping, bibliometric analysis, and periodic institutional data updates.

ACKNOWLEDGEMENTS

The authors would like to express their sincere gratitude to the Rector of Universitas Muslim Indonesia (UMI) Makassar for providing research funding through the Lembaga Penelitian dan Pengembangan Sumberdaya (LP2S) UMI under the Penelitian Unggulan Fakultas (PUF) scheme. This support, granted in 2025, made it possible to carry out this research and produce the resulting scientific outputs..

REFERENCES

- Adila, N. (2022). Implementation of Web Scraping for Journal Data Collection on the SINTA Website. *Sinkron*, 7(4), 2478–2485. <https://doi.org/10.33395/sinkron.v7i4.11576>
- Aidillah, K. H., Dodi Vionanda, & Dony Permana. (2025). Analysis on Scopus Articles Padang State University Based on SINTA Website. *UNP Journal of Statistics and Data Science*, 3(1), 79–85. <https://doi.org/10.24036/ujsds/vol3-iss1/346>
- Asis, M. A., Purnawansyah, P., & Salim, Y. (2024). Rancang Bangun Sistem Manajemen Data Akreditasi berbasis Web. *Journal Cerita*, 10(1), 32–38. <https://doi.org/10.33050/cerita.v10i1.2989>
- Darmawan, I., Maulana, M., Gunawan, R., & Widiyasono, N. (2022). Evaluating Web Scraping Performance Using XPath, CSS Selector, Regular Expression, and HTML DOM With Multiprocessing Technical Applications. *JOIV: International Journal on Informatics Visualization*, 6(4), 904. <https://doi.org/10.30630/joiv.6.4.1525>
- Fahrudin, T. M., Funabiki, N., Brata, K. C., Naing, I., Aung, S. T., Muhaimin, A., & Prasetya, D. A. (2025). An Improved Reference Paper Collection System Using Web Scraping with Three Enhancements. *Future Internet*, 17(5), 195. <https://doi.org/10.3390/fi17050195>
- Giuliana, B., Mariangela, L., & Marianna, L. (2024). Bibliometric Insights into Web Scraping and Advanced AI-Based Models for Valuable Business Data. *Proceedings of the 26th International Conference on Enterprise Information Systems*, 321–328. <https://doi.org/10.5220/0012686900003690>
- Goel, A., Zhu, J., Netravali, R., & Madhyastha, H. V. (2024). Sprinter: Speeding Up High-Fidelity Crawling of the Modern Web. *NSDI'24: Proceedings of the 21st USENIX Symposium on Networked Systems Design and Implementation*.
- Guyt, J. Y., Datta, H., & Boegershausen, J. (2024). Unlocking the Potential of Web Data for Retailing Research. *Journal of Retailing*, 100(1), 130–147. <https://doi.org/10.1016/j.jretai.2024.02.002>
- Hidayat, M. K., Sugiarto, D., Fitriana, R., & Liang, Y.-C. (2025). Business intelligence system model to measure the performance of lecturers' scientific publications. *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, 23(4), 954. <https://doi.org/10.12928/telkomnika.v23i4.26221>
- Kazmali, A. S., & Sayar, A. (2025). Web Scraping: Legal and Ethical Considerations in General and Local Context - A Review. *Procedia Computer Science*, 259, 1563–1572. <https://doi.org/10.1016/j.procs.2025.04.111>
- Kurniatuhadi, R., Hadary, F., Sulistyarini, S., Anzani, Y. M., Fahrana, Y., & Muthahhari, M. (2024). The Chronology of Development Tracer Study System at Tanjungpura University. *Jurnal Teknologi Informasi Dan Pendidikan*, 17(1), 218–228. <https://doi.org/10.24036/jtip.v17i1.746>
- Lengkong, J. S. J., Jacobus, S. N. H., Dondokambey, R., Ratumbuisang, K. F., Paath, D., & Liow, E. S. (2023). Web-Based Academic Information Systems in Vocational School. *International Journal of Information Technology and Education*, 2(4), 12–25.

<https://doi.org/10.62711/ijite.v2i4.153>

- Logos, K., Brewer, R., Langos, C., & Westlake, B. (2023). Establishing a framework for the ethical and legal use of web scrapers by cybercrime and cybersecurity researchers: learnings from a systematic review of Australian research. *International Journal of Law and Information Technology*, 31(3), 186–212. <https://doi.org/10.1093/ijlit/eaad023>
- Lotfi, C., Srinivasan, S., Ertz, M., & Latrous, I. (2021). Web Scraping Techniques and Applications: A Literature Review. In *SCRS CONFERENCE PROCEEDINGS ON INTELLIGENT SYSTEMS* (pp. 381–394). Soft Computing Research Society. <https://doi.org/10.52458/978-93-91842-08-6-38>
- Madjido, M., Espressivo, A., Maula, A. W., Fuad, A., & Hasanbasri, M. (2019). Health Information System Research Situation in Indonesia: A Bibliometric Analysis. *Procedia Computer Science*, 161, 781–787. <https://doi.org/10.1016/j.procs.2019.11.183>
- Molina, I., Morales, J., & Keith, B. (2025). Web Scraping Chilean News Media: A Dataset for Analyzing Social Unrest Coverage (2019–2023). *Data*, 10(11), 174. <https://doi.org/10.3390/data10110174>
- Purnomo, L. M., & Ayub, M. (2021). Analisis Data Hasil Web Scraping untuk Menentukan Kualitas Jurnal Ilmiah. *Jurnal Strategi*, 3(1), 122–132.
- Rahmatulloh, A., & Gunawan, R. (2020). Web Scraping with HTML DOM Method for Data Collection of Scientific Articles from Google Scholar. *Indonesian Journal of Information Systems*, 2(2), 95–104. <https://doi.org/10.24002/ijis.v2i2.3029>
- Rochim, A. F., Nugraha, T., Widodo, A. P., Eridani, D., & Martono, K. T. (2021). Design data collection tool and weighting classification of authors in their scholar outputs based on percent-contribution-indicated (PCI) method. *Journal of Physics: Conference Series*, 1943(1), 012111. <https://doi.org/10.1088/1742-6596/1943/1/012111>
- Sahria, Y. (2020). Implementasi Teknik Web Scraping pada Jurnal SINTA Untuk Analisis Topik Penelitian Kesehatan Indonesia. *URECOL (University Research Colloquium)*, 297–306.
- Saniyah, A., & Ghozali, M. L. (2023). Implementation of the DSN-MUI Fatwa on Product Testimonials from Istihsan Perspective. *EKSYAR: Jurnal Ekonomi Syari'ah & Bisnis Islam*, 10(2), 223–232. <https://doi.org/10.54956/eksyar.v10i2.474>
- Shah, A., Shah, H., Bafna, V., Khandor, C., & Nair, S. (2025). Validation and Extraction of Reliable Information Through Automated Scraping and Natural Language Inference. *Engineering Applications of Artificial Intelligence*, 147, 110284. <https://doi.org/10.1016/j.engappai.2025.110284>
- Ulfah, A., & Najiah, I. (2023). Implementasi Web Scraping pada Situs Jurnal Sinta Menggunakan Framework Selenium Webdriver Python. *JIKA (Jurnal Informatika)*, 7(1), 29. <https://doi.org/10.31000/jika.v7i1.7037>
- Yuda, M. A. D. (2025). Implementasi Web Scraping Untuk Ekstraksi Data Penjual dan Produk Panel Surya Di E-Commerce. *KONSTELASI: Konvergensi Teknologi Dan Sistem Informasi*, 5(1). <https://doi.org/10.24002/konstelasi.v5i1.11751>
- Zbarcea, A., & Tudose, C. (2024). Migrating from Developing Asynchronous Multi-Threading Programs to Reactive Programs in Java. *Applied Sciences*, 14(24), 12062. <https://doi.org/10.3390/app142412062>