MATICS: Jurnal Ilmu Komputer dan Teknologi Informasi
(*Journal of Computer Science and Information Technology)*
ISSN : 1978-161X(p); 2477-2550(e)
Volume 15, No. 2(2023), pp 84-89
Received: October 11th 2023; **Accepted**: October 17th 2023; **Avalaible Online**: October 23th 2023

Implementation Semantic Annotation Recognizing Technique in the Scraper Engine on the E-Publishing Website of the National Research and Innovation Agency (BRIN) Indonesia

Muhammad Izzun Ni'am, Muhammad Haris Firmansyah, and Zikrie Pramudia Alfarhisi

Abstract - The increasing need for swift information dissemination in line with modern technological advancements has emphasized the importance and significant impact of data analysis and processing as relevant academic disciplines. These processes encompass data acquisition from various sources, either through direct collection or extraction methods. Among the most crucial and widely utilized techniques for extracting data from the internet is web scraping, particularly when gathering data for research maintenance during the consolidation of multiple institutions into BRIN (National Research and Innovation Agency). Challenges emerge in effectively integrating existing research into a unified system without proper upkeep, as neglecting maintenance can lead to system degradation and hinder access to stored research. Successful maintenance necessitates centralized repositories for researchers' work data. The implementation of semantic annotation recognizing techniques within the web scraping feature of the E-Publishing website holds the potential to expedite this process. The use of web scraping promises to significantly simplify research data collection, while semantic annotation recognizing techniques are poised to streamline implementation, particularly due to the XML data foundation within the Open Archives Initiative (OAI) system. In the context of institution merging and research sustainability, technologies like web scraping and semantic annotation recognizing play pivotal roles in addressing these challenges.

***Index Terms*— Scrapping, Scrapper Engine, Semantic Annotation Recognizing, Website, Web Scrapping.**

I. Introduction

Information has become the most crucial commodity in this era [1]. The pace at which information is delivered is directly linked to the level of current technological advancements. In Indonesia, where more than 169 million internet users generate millions of data points every second, the advancement of technology highlights the importance of data analysis and processing [2]. Analyzing data sourced from the internet can aid in grasping user preferences, assisting security organizations in detecting potential vulnerabilities, and enabling companies in the goods and services industry to pinpoint their promotional targets [3]. Data analysis requires effective data extraction as one of the initial steps in preparing and processing data for further analysis. Data extraction plays a vital role in a range of functions within diverse fields and activities, including the domain of publishing and publications [4]. In the context of publishing and publication, data extraction is employed to gather articles, content, or other information for dissemination through various media, websites, or scientific journals.

In Indonesia, research works and publications are currently managed by the Open Journal System (OJS) based on specific research subjects. OJS is an open-source software developed by the Public Knowledge Project (PKP) for journal and publication management [5]. Over time, the system has suffered from insufficient maintenance, causing the research stored within it to become inaccessible. This has significantly negatively affected the academic community in Indonesia, as many researchers depend on it for reference and citation purposes. The absence of maintenance for the current system has made it challenging to access numerous valuable research studies [6]. As the government agency overseeing research and technology, the National Research and Innovation Agency (BRIN) is responsible for consolidating the works produced by researchers to

Ni'am, Muhammad Izzun is from Technology Information Programs on the Faculty of Vocational Studies, Universitas Brajiya, Malang, Indonesia. (corresponding author provide phone 085755567252; e-mail mizzunniam@ub.ac.id)

Firmansyah, Muhammad Haris is Technology Information Programs on the Faculty of Vocational Studies, Universitas Brajiya, Malang, Indonesia. (e-mail mail.haris@student.ub.ac.id).

Alfarhisi, Zikrie Pramudia is Technology Information Programs on the Faculty of Vocational Studies, Universitas Brajiya, Malang, Indonesia. (e-mail zikriepa@ub.ac.id)

ensure their effective maintenance. Hence, maintenance efforts are crucial for centralizing the storage of data and information from researchers' works.

Previous research has explained various data extraction methods such as Web Scraping [7], Optical Character Recognition [8], Pattern Recognition [9] SQL Query [10], Natural Language Processing [11], Data Mining [12], and Geographic Information Systems [13]. Scraping method is one of the most critical and widely used techniques for extracting data from the internet [14]. Scraping is the process of obtaining and extracting data to arrange it systematically and present it in a user-friendly manner [15]. Scraping, commonly known as web scraping, involves the automated retrieval of data from websites. Various methods can be utilized for web scraping, such as text pattern matching, HTML parsing, DOM parsing, semantic annotation recognition, and vertical aggregation [16]. The specific model used depends on the type of website and the data to be annotated. Web scraping with semantic annotation recognizing relies on existing models to obtain data and metadata [17].

The implementation of web scraping will significantly simplify the collection of research data, thus facilitating the maintenance process [18]. The semantic annotation recognizing technique will also accelerate the implementation of web scraping because the Open Journal System (OJS) currently supports the Open Archives Initiative (OAI) system, which offers publicly accessible content in Extensible Markup Language (XML) format [7]. This will serve as the foundational model for scraping in this research.

II. Literature Review

A. Data Extraction

Data extraction techniques refer to the methods and processes used to retrieve structured or unstructured data from various sources such as websites, databases, documents, or other repositories. These techniques are essential for collecting, transforming, and preparing data for analysis, reporting, or storage [4]. Some common methods for data extraction are Web Scraping, Optical Character Recognition (OCR) technology, Pattern Matching, SQL Queries, Natural Language Processing (NLP), Data Mining, and Geographic Information Systems (GIS).

These data extraction techniques play a crucial role in various industries, enabling the collection of valuable insights, informed decision-making, and the automation of data-related processes [15]. The choice of technique depends on factors such as the data source, format, and the specific requirements of the data extraction task at hand.

B. Web Scraping

Web scraping is the automated process of extracting data from websites. It involves fetching web pages, parsing the HTML or XML content, and then selecting specific data elements for extraction [7]. Web scraping is widely used for collecting data from websites, social media, and online databases. Some common web scraping techniques are Text Pattern Matching, HTML Parsing, DOM Parsing, XPath, CSS Selectors, Headless Browsers, API Calls, Semantic Annotation Recognizing, Vertical Aggregation, and Dynamic Content Extraction.

C. Semantic Annotation Recognizing Technique

The Semantic Annotation Recognizing Technique is an advanced method used in web scraping to extract structured data from web pages. Unlike traditional web scraping techniques that rely on parsing HTML and searching for specific patterns, semantic annotation recognizing uses predefined models or patterns to recognize and extract relevant information [19].

III. Methods

The research was conducted for approximately 5 months, from August 1, 2022, to December 31, 2022, and was carried out online from the BRIN headquarters located at B.J. Habibie Building, Jl. M.H. Thamrin No. 8, Central Jakarta. Data collection was done through observation and interviews. Data was gathered through online observation, involving direct examination of BRIN's OJS websites to acquire information regarding the system requirements for building the E-Publishing system. This system encompasses repository management, repository types, repository subjects, data sources, and metadata.

Data was collected online through interviews with one of the RMPI BRIN employees to gain insights into the system requirements for constructing the E-Publishing system. The system design phase was carried out using UML diagrams to provide a visual representation of the system flow to be created. The utilization of UML was aimed at providing developers with a clear visualization of the system, minimizing misunderstandings during the system development process.

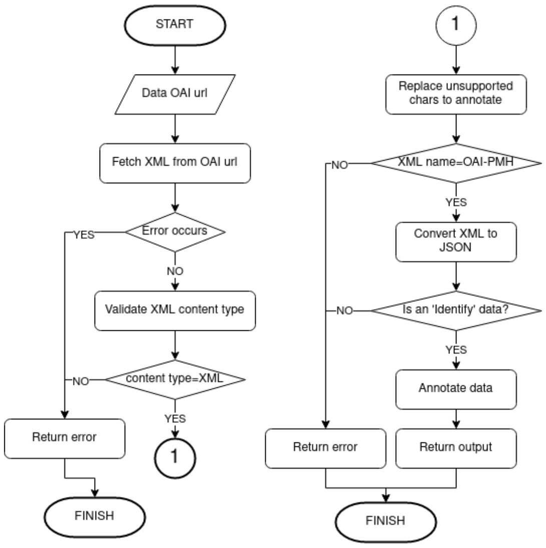


Fig. 1. Flowchart scraper identify

There are two types of data to extract: identification data and metadata. Identification data is retrieved to acquire publisher information from the OAI website, while metadata is collected to obtain publication data or articles published on the OAI website. Figure 1 displays the flowchart for the identification scraper. The flowcharts for the metadata scraper can be seen in Figure 2.

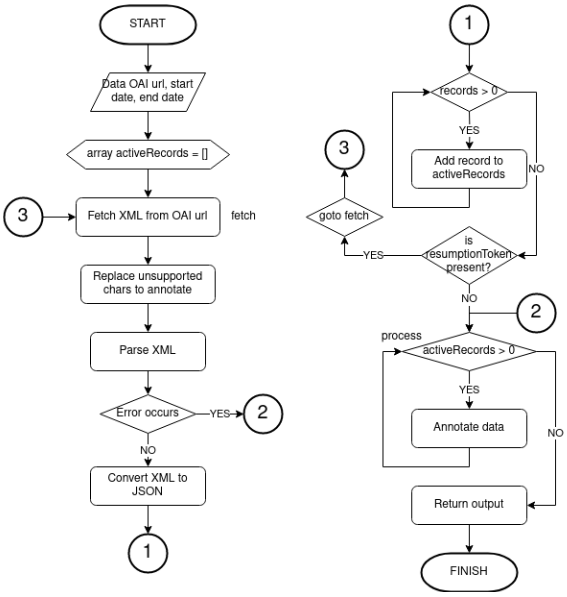


Fig. 2. Flowchart scraper metadata

There is a distinction in the design of the identification scraper and the metadata scraper using flowcharts because the data processed by each scraper varies significantly. The identification scraper focuses on fundamental repository data, while the metadata scraper deals with a comprehensive list of articles within that repository [7].

IV. Results and Discussions

The E-Publishing website is an internal platform within BRIN used for centralized publication management under the Directorate of Repositories, Multimedia, and Scientific Publishing (RMPI) of BRIN. The utilization of the E-Publishing website for publication management encompasses repository management, repository types, repository subjects, data sources, and metadata. Various features are available in the admin section, such as email and password-based login, a dashboard to monitor visitor activities, data extraction from external sources using scraping techniques, user management for administering the website with various access levels, and self-profile management.

Additionally, in the public section, there are features like specific publication searches based on repository type or overall listings, a list of publications sortable by most recently created or most viewed by users, a list of articles within a publication with search capabilities, and detailed publication article pages providing information and download buttons for full text and source articles. There are also "About" pages to view information about the directorate and "Contact" pages for reaching out to the administrative or support team.

In the data source management feature, there is a scraper tool used to extract data about publications and publication articles from various BRIN websites using web scraping techniques. The primary goal of the scraper feature is to obtain comprehensive data regarding publication information, such as publication manager contacts, publication versions, and the initial publication dates. Furthermore, the data to be retrieved includes article publication data, including title, authors, descriptions, publishers, contributors, publication dates, and other technical data required by the system.

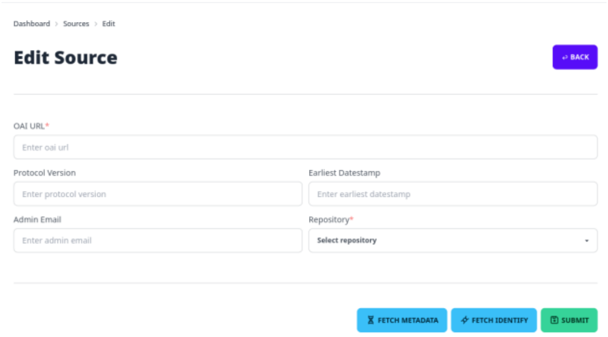
The extraction of publication article data is filtered based on specified start and end dates. Additionally, the articles that are extracted include those with information indicating that the data has not been deleted, and at the end of the data, there is a resumptionToken entity indicating that there are a considerable number of articles within the publication. This ensures that the scraper feature can be used to its fullest potential and effectiveness.

A. System Implementation

The implementation of the scraper on the E-Publishing website is in the form of an Application Programming Interface (API) using the PHP programming language within the Laravel framework. The API scraper implementation on the server side will be divided into two types: the identification scraper and the metadata scraper. Both of these scrapers will utilize semantic annotation recognizing techniques to mark the data elements to be extracted and used as the final output in JSON format. This final data can then be further processed by the website admin.

The implementation of the scraper within the E-Publishing website on the client side takes the form of a user interface for interacting with users. The interface to be implemented is based on the design phase as mentioned earlier. The scraper interface consists of two parts: one for the identification scraper and one for the metadata scraper. These interfaces are built using the ReactJS framework and will utilize the APIs created on the server.

Figure 3 illustrates the outcomes of the identification scraper interface that has been developed. Users can enter the OAI URL and choose a repository to start the scraping process in order to retrieve identification data. The identification data to be extracted from the OAI website encompasses protocol version, earliest date stamp, and admin email. This data will be automatically filled into the form if the scraping process is successful. In case of an unsuccessful scraping process, the system will generate an error message. After users obtain the scraped data, they can save it by clicking the submit button.



Dashboard > Sources > Edit

Edit Source

BACK

OAI URL*

Enter oai url

Protocol Version

Enter protocol version

Earliest Datestamp

Enter earliest datestamp

Admin Email

Enter admin email

Repository*

Select repository

FETCH METADATA

FETCH IDENTIFY

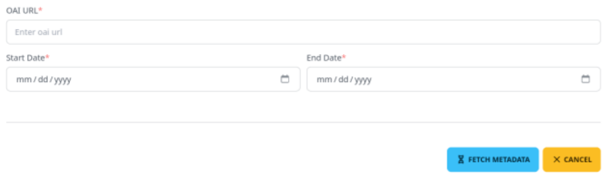
SUBMIT

Fig. 3. Identification scraper interface

Figure 4 illustrates the results of the metadata scraper interface that has been created. This form will appear after users click the "fetch metadata" button below the identification scraper form. Users can initiate the scraping process by inputting the OAI URL, start date, and end date in the scraper form. The OAI URL will automatically be filled based on the OAI URL in the identification scraper form. Users can input a start date to set the initial date for articles published on the OAI website, and for the end date, users can input the desired date. Once all the fields are filled, users can click the "fetch metadata" button to retrieve data from the OAI website. If users do not wish to perform scraping, they can close the form by clicking the "cancel" button.

Table 2. Results of testing the scraper metadata with standard deviation parameters

Request	Min	Q1	Median	Q3	Max
25	7.2	8.9	9.4	11.5	22.9
50	6.9	8.1	9.5	11	20.1
75	6.3	7.4	8.6	9.4	19.4
100	6.3	7.7	8.8	9.9	15.8
125	7.4	7.9	8.8	9.5	13.7



OAI URL*

Enter oai url

Start Date*

mm / dd / yyyy

End Date*

mm / dd / yyyy

FETCH METADATA

CANCEL

Fig. 4. Metadata scraper interface

B. *System Testing*

Testing is divided into functional testing and performance testing. Testing is not conducted on the hosted website but on a local website within that environment. This is done to ensure that testing can be carried out optimally without being influenced by external factors, thus obtaining neutral testing results. Performance testing is carried out using standard deviation and throughput parameters. The data sent from JMeter to the identification and metadata scrapers is JSON data sent in quantities of 25, 50, 75, 100, and 125 with 20 repetitions. The testing conducted will generate a summary from which minimum, Q1, median, Q3, and maximum values will be determined. This data is then visualized using box plots for standard deviation and bar graphs for throughput. Performance testing is carried out based on the scenarios described in the design section. The standard deviation from the performance testing of the identification scraper in the process of scraping data from the OAI website can be seen in Table 1.

There are minimum, Q1, median, Q3, and maximum values for each testing scenario in Table 1. Data visualization is performed by converting the data in Table 1 into box plots to facilitate the analysis process. The box plot for the data in Table 1 can be seen in Figure 5.

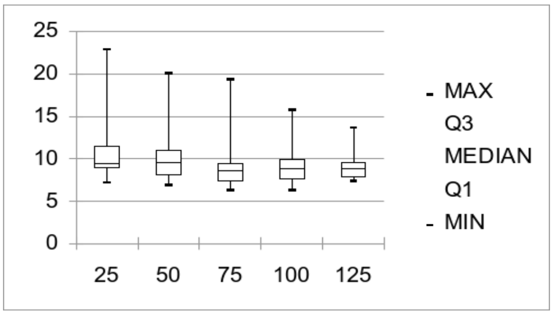


Fig. 5. Result test of scraper identify using parameter standard deviation

Table 1. Results of testing the identify scraper with standard deviation parameters

Request	Min	Q1	Median	Q3	Max
25	7.2	8.5	10.2	11.5	24.2
50	6.6	7.6	8.5	11.3	20.9
75	6.4	8.1	9.3	12.7	18.4
100	6.6	8.3	9.6	10.8	17.3
125	7.3	8.4	9	9.6	12.9

In the box plot, generally, the number of requests made does not have a significant impact on the deviation, meaning that the number of requests does not greatly affect the performance of the identification scraper. Additionally, there is a decrease in outliers as the number of requests increases, approaching the Q3 value. Deviation widens significantly with 75 requests, ranging between 8.1 to 12.7 (excluding outliers). The most optimal deviation is observed with 125 requests, with a median value of 9 deviations. The standard deviation from the performance testing of the metadata scraper in the process of scraping data from the OAI website can be seen in Table 2.

There are minimum, Q1, median, Q3, and maximum values for each testing scenario in Table 2. Data visualization is performed by converting the data in Table 2 into box plots to facilitate the analysis process.

The box plot for the data in Table 2 can be seen in Figure 6.

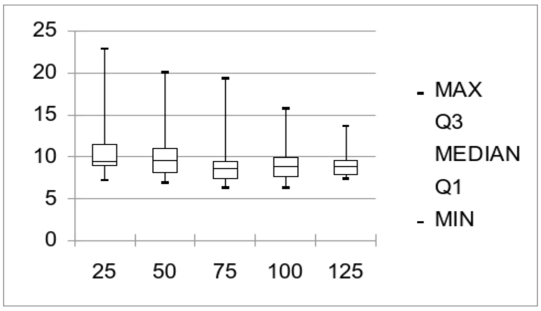


Fig. 6. Result of testing scraper metadata using parameter standard deviation

In the box plot, generally, the number of requests made has a small impact on the deviation, meaning that a higher number of requests can affect the deviation. Additionally, there is a decrease in outliers as the number of requests increases, approaching the Q3 value. Deviation narrows significantly with 75 requests, with a median of 8.6 or a decrease of 9.2% from the previous 50 requests. The deviation values tend to be optimal with a median ranging from 8.6 to 9.5, inclusive. The throughput from the performance testing of the identification scraper in the process of scraping data from the OAI website can be seen in Table 3.

There are minimum, Q1, median, Q3, and maximum values for each testing scenario in Table 3. Data visualization is performed by converting the data in Table 3 into bar charts to facilitate the analysis process. The bar chart for the data in Table 3 can be seen in Figure 7.

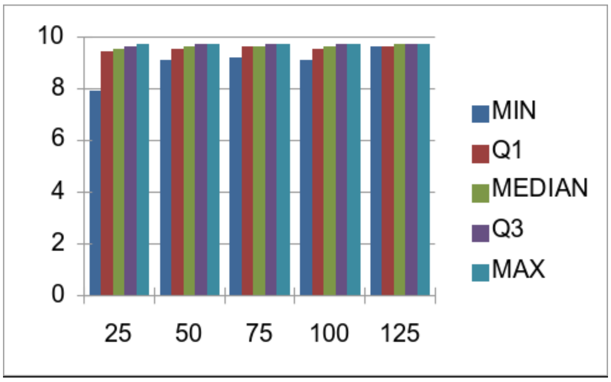


Fig. 7. Result of testing scraper identify using parameter throughput

In the diagram, generally, the number of requests made has a small impact on the throughput, meaning that the scraper can process data at a relatively increased speed as the number of requests increases. The increase in median values for each scenario is directly proportional to the increase in the number of requests, indicating improved performance with a higher number of requests. The throughput from the performance testing of the metadata scraper in the process of scraping data from the OAI website can be seen in Table 4.

Table 3. Results of Testing the Identify Scraper with Throughput Parameters

Request	Min	Q1	Median	Q3	Max
25	7.9	9.4	9.5	9.6	9.7
50	9.1	9.5	9.6	9.7	9.7
75	9.2	9.6	9.6	9.7	9.7
100	9.1	9.5	9.6	9.7	9.7
125	9.6	9.6	9.7	9.7	9.7

There are minimum, Q1, median, Q3, and maximum values for each testing scenario in Table 4. Data visualization is performed by converting the data in Table 4 into bar charts to facilitate the analysis process. The bar chart for the data in Table 4 can be seen in Figure 8. In Figure 14, the obtained throughput values vary. The median values for each scenario result in the same data, which is 9.1 ops/sec. The highest throughput is recorded with 150 requests, which is 9.4 ops/sec, while the lowest throughput is recorded with 125 requests, which is 8.2 ops/sec.

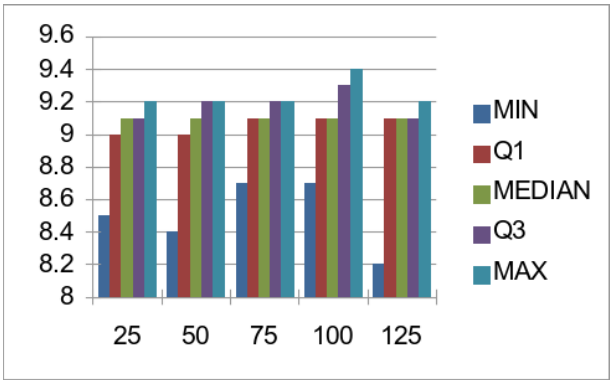


Fig. 8. Result of Testing Scraper Metadata Using Parameter Throughput

Table 4. Results of testing the metadata scraper with throughput parameters

Request	Min	Q1	Median	Q3	Max
25	8.5	9	9.1	9.1	9.2
50	8.4	9	9.1	9.2	9.2
75	8.7	9.1	9.1	9.2	9.2
100	8.7	9.1	9.1	9.3	9.4
125	8.2	9.1	9.1	9.1	9.2

V. CONCLUSIONS

Based on the results of the entire process that has been conducted, it can be concluded that the semantic annotation recognizing technique has been successfully implemented in the scraper on the BRIN E-Publishing website to retrieve data from the OAI website. The implementation is divided into two parts: implementation for the identify scraper and metadata scraper. Both scrapers have successfully passed the testing stages, which include input, processing, and output of data.

The scraper that has implemented the semantic annotation recognizing technique can generate the required output for the E-Publishing website system. Furthermore, the performance of the scraper that has implemented the semantic annotation recognizing technique through testing shows that the average time for the identify and metadata scraper operations in the scraping process is 9.17 seconds, indicating that the scraper's performance is optimal and good because it does not exceed 15 seconds. The average operations that can be performed by the identify and metadata scraper are 9.35 operations per second, demonstrating that the scraper's performance is optimal as it can handle more than one operation per second.

REFERENCES

[1] A. Sigov, L. Ratkin, L. A. Ivanov, and L. Da Xu, "Emerging Enabling Technologies for Industry 4.0 and Beyond," *Information Systems Frontiers*, Jan. 2022, doi: 10.1007/s10796-021-10213-w.

[2] Badan Pusat Statistik Indonesia, "Telecommunication Statistics in Indonesia 2022," Jakarta, Jul. 2023.

[3] R. H. Hariri, E. M. Fredericks, and K. M. Bowers, "Uncertainty in big data analytics: survey, opportunities, and challenges," *J Big Data*, vol. 6, no. 1, Dec. 2019, doi: 10.1186/s40537-019-0206-3.

[4] rd Mustafa Al Rifaee, "A Comparison of Web Data Extraction Techniques," *IEEE - Jordan Intenational Joint Conference on Electrical Enginering and Information Technology (JEEIT)*, pp. 785–789, 2019.

[5] Amrizal, "Pemanfaatan Open Jurnal System (OS) untuk Pengelolaan Jurnal Lumbung di Politeknik Pertanian Negeri Payakumbuh," *Lumbung*, vol. 17, no. 2, pp. 64–74, 2018.

[6] Y. Fu and J. Schneider, "Towards knowledge maintenance in scientific digital libraries with the keystone framework," in *Proceedings of the ACM/IEEE Joint Conference on Digital Libraries*, Institute of Electrical and Electronics Engineers Inc., Aug. 2020, pp. 217–226. doi: 10.1145/3383583.3398514.

[7] M. Khder, "Web Scraping or Web Crawling: State of Art, Techniques, Approaches and Application," *International Journal of Advances in Soft Computing and its Applications*, vol. 13, no. 3, pp. 145–168, Dec. 2021, doi: 10.15849/IJASCA.211128.11.

[8] N. Islam, Z. Islam, and N. Noor, "A Survey on Optical Character Recognition System," 2016.

[9] J. Liu, J. Sun, and S. Wang, "Pattern Recognition: An overview," 2006.

[10] A. Jain and A. Doan, "SQL Queries Over Unstructured Text Databases," *IEEE Access*, pp. 1255–1257, 2006.

[11] C. Nath, M. S. Albaghdadi, and S. R. Jonnalagadda, "A natural language processing tool for large-scale data extraction from echocardiography reports," *PLoS One*, vol. 11, no. 4, Apr. 2016, doi: 10.1371/journal.pone.0153749.

[12] J. L. Martinez-Rodriguez, I. Lopez-Arevalo, and A. B. Rios-Alvarado, "Mining information from sentences through Semantic Web data and Information Extraction tasks," *J Inf Sci*, vol. 48, no. 1, pp. 3–20, Feb. 2022, doi: 10.1177/0165551520934387.

[13] G. Shi and K. Barker, "International Conference on Spatial Data Mining and Geographical Knowledge Services.," *IEEE Access*, pp. 273–278, 2011.

[14] B. Bhardwaj, S. I. Ahmed, J. Jaiharie, R. Sorabh Dadhich, and M. Ganesan, "Web Scraping Using Summarization and Named Entity Recognition (NER)," in *2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS)*, IEEE, Mar. 2021, pp. 261–265. doi: 10.1109/ICACCS51430.2021.9441888.

[15] S. De and S. Sirisuriya, "A Comparative Study on Web Scraping," *Processings of 8th Intenational Research Conference*, vol. 8, pp. 135–140, 2015.

[16] R. Vording, "Harvesting unstructured data in heterogenous business environments; exploring modern web scraping technologies," *Twente Student Conference on IT*, vol. 34, pp. 1–9, 2020.

[17] S. K. Malik and S. Rizvi, "Information extraction using web usage mining, web scrapping and semantic annotation," in *Proceedings - 2011 International Conference on Computational Intelligence and Communication Systems, CICN 2011*, 2011, pp. 465–469. doi: 10.1109/CICN.2011.97.

[18] A. Vlachidis, C. Binding, K. May, and D. Tudhope, "Automatic metadata generation in an archaeological digital library: Semantic annotation of grey literature," *Studies in Computational Intelligence*, vol. 458, pp. 187–202, 2013, doi: 10.1007/978-3-642-34399-5_10.

[19] V. Uren *et al.*, "Semantic annotation for knowledge management: Requirements and a survey of the state of the art," *Web Semantics*, vol. 4, no. 1, pp. 14–28, Jan. 2006, doi: 10.1016/j.websem.2005.10.002.