International Journal of Scientific Research and Engineering Development-– Volume 9 Issue 2, Mar-Apr 2026
Available at www.ijsred.com

RESEARCH ARTICLE OPEN ACCESS

SCRAPE-IT: Data Scraper for Social-Media Platform Analytics

Alok Singh, Amogh Dixit, Faizan Badr, Dr. Diwakar Yagyasen
Department of Computer Science , Babu Banarasi Das Institute of Technology and Management, Lucknow ,U.P.,India
Email: aloksingh12a06@gmail.com
Department of Computer Science , Babu Banarasi Das Institute of Technology and Management, Lucknow ,U.P.,India
Email: dixitbetu37@gmail.com
Department of Computer Science , Babu Banarasi Das Institute of Technology and Management, Lucknow ,U.P.,India
Email: faizanbadr0@gmail.com
Department of Computer Science , Babu Banarasi Das Institute of Technology and Management, Lucknow ,U.P.,India
Email: dylucknow@bbdnitm.ac.in

----------**********----------

Abstract:

Platforms such as Reddit have become essential tools for gauging public opinion and monitoring trending topics, but the sheer volume and speed of this information create substantial challenges for data acquisition. This project will meet the requirement for organized and up-to-date social data by creating a modular, Python-coded Reddit Data Scraper to convert unorganized HTML into an organized data stream. Although manual data acquisition is necessarily error-susceptible and impractical on a large scale, this automated tool makes it easier to systematically retrieve data for analysis purposes such as sentiment analysis and topic modeling specifically in dynamic content processing. Ethical issues are profoundly intertwined through rate limiting and pseudonymization to reconcile scientific advancement and privacy. The final design results in a verified tool that optimizes research efficiency and provides a basis for complex NLP applications. Future versions will incorporate interactive dashboards for real-time analysis and extend the tool to multi-platform trend analysis.

Keywords — **Web-Based, Data Scraper, Scraping, Reddit, Extraction, Beautiful Soup, JSON, CSV, Social, Media, Meta, X, Scraper, Data gathering, Data Analytics.**

----------**********----------

1. INTRODUCTION

1.1 *Project Overview and Context*

In the modern digital landscape, online communities have evolved into primary sources of information regarding public opinion and emerging trends. Platforms like Reddit, often described as "goldmines" for spotting new trends, generate a constant flow of unstructured information. To derive actionable insights from this chaos, researchers and data scientists require rob mechanisms to systematically collect and structure this data.

The system design is divided into three main layers: Data Acquisition, Processing, and Storage . The data acquisition layer employs the PRAW library for API-friendly access, with additional BeautifulSoup support for customized HTML parsing in cases where API access is limited. The data is then cleaned in the processing layer with the pandas library to optimize high-quality metadata, including post titles, text, and interaction data. Finally, a storage layer exports results in JSON and CSV formats to maximize compatibility with analysis software such as R or Tableau.

The massive and rapidly changing information present on platforms such as Reddit is a valuable, real-time source of community sentiment, especially in the programming community. The problem, however, is that this information is necessarily unstructured and disorganized. The main problem for researchers and data scientists is that of requiring a strong, automatic, and targeted system to efficiently harvest discussion titles and posts from important subreddits, turning raw HTML and API results into structured metadata.This project focuses on the design, implementation, and evaluation of a custom web scraping tool. The tool is engineered to gather specific data points—including posts, comments, and engagement metrics—from targeted subreddits to facilitate trend analysis. By creating a reliable data pipeline, the project aims to identify shifting narratives and gauge sentiment within niche communities, transforming real-time conversations into structured datasets ready for analysis.

As noted in the current literature, web data tends to be disorganized, inconsistent, and incomplete, thus requiring a “data hermeneutical perspective” where technical constraints are seen as signs of social and organizational processes. Without a proper schema, data such as post bodies, upvote counts, and timestamps cannot be tapped for large-scale analysis. Moreover, human analysis is unable to cope with the “information overload” present in social media timelines, where key conversations are often buried under unnecessary and irrelevant posts. This particular project aims to address the research gap in developing a scalable ETL (Extract, Transform,

Load) process that filters data at the point of extraction to ensure high signal-to-noise ratios.

1.2 *Problem Statement*

The central problem addressed by this project is the "unstructured" nature of vast, rapidly evolving information on social platforms. While Reddit represents a rich source of community sentiment, particularly within the programming world, the raw data is often inaccessible for direct computational analysis. There is a critical need for an automated, targeted system that can efficiently extract discussion titles and relevant posts, transforming raw HTML or JSON into structured data.Furthermore, standard manual data collection is prone to errors, lacks scalability, and cannot keep pace with the velocity of social media. Existing solutions often struggle with the balance between ethical compliance (API limits) and the need for deep, historical data, necessitating a hybrid approach to data engineering.

1.3 *Objectives*

The core objectives of the proposed work are defined as follows:Targeted Extraction: Develop a scraper using requests, BeautifulSoup, and PRAW to extract post titles and links from specific subreddits (e.g., /r/userquery1).Filtering & Pipeline: Implement keyword-based filtering and develop a pipeline to store data in both JSON (for structure) and CSV (for interoperability).Quantitative Analysis: Provide a clear summary of relevant topics found across targets.

Ethical Compliance: Strict adherence to ethical practices, specifically implementing time.sleep() delays to respect server load and API rate limits.

2. Background & Related Work

Existing literature focuses on the importance of web scraping as an interpretive task in the data science process. Scholars such as Jakob Jünger argue for a "data hermeneutical perspective," considering web scraping as a co-production of knowledge between platforms and researchers. Methodological analyses, such as Lotfi's systematic review of 180 papers, focus on the importance of the Pushshift API in accessing historical Reddit data. In addition, comparative studies by Kahlon and Singh compare the performance of Python-based tools such as Scrapy and Selenium with desktop tools. Finally, scholars such as Dogucu and Cetinkaya argue for the importance of incorporating these methods into academic curricula to close the gap between theoretical modeling and real-world data acquisition.

2.1 *The Hermeneutical Perspective on Data*

A significant portion of the literature emphasizes that scraping is not merely a technical act of downloading but an interpretive process. Jünger (2023) introduces the "data hermeneutical perspective," arguing that social media data is co-produced by users, platforms, and researchers.Representation vs. Reality: Researchers do not observe entities directly but rather "shadows of actors" through URLs and data representations.Format Influence: The structure of data—whether HTML for users or JSON for APIs—guides the process of knowledge production differently. Transforming hierarchical data into flat tabular formats is a process of "reconstructive data hermeneutics" rather than simple measurement.Platform Mediation: Limitations such as rate limits and access restrictions are not just technical hurdles; they indicate the platform's logic and priorities (e.g., marketing over research).

2.2 *Web Scraping Technologies: Static vs. Dynamic*

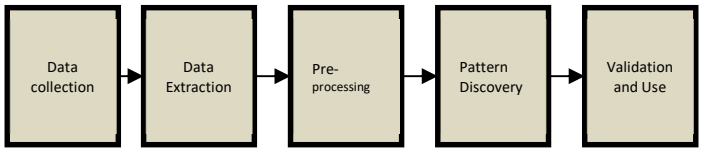

Fig 1: Scraping Workflow

The literature draws a sharp distinction between scraping static content and dynamic, client-side rendered content.

Static Scraping: For static pages, libraries like BeautifulSoup are highlighted as efficient parsing tools that create a Document Object Model (DOM) tree from HTML fetched via requests. This method is preferred for its speed and lower resource consumption when the target data is embedded directly in the initial HTML response.

Dynamic Scraping (Selenium): Modern web applications (SPAs) often load data asynchronously via JavaScript (AJAX).) traditional HTTP requests fail here. Selenium, a browser automation framework, mimics human actions (scrolling, clicking) to trigger JavaScript execution, making it essential for capturing dynamic content. However, this comes at the cost of higher CPU and memory usage.

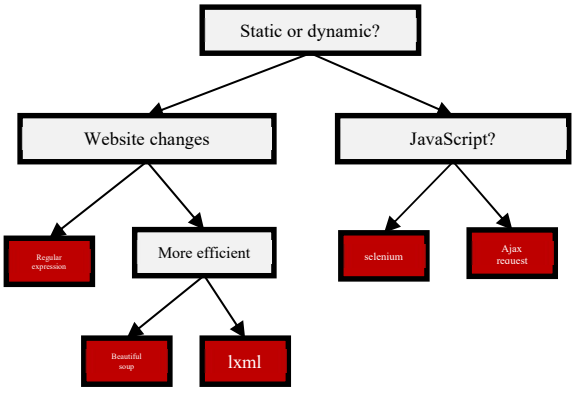

Fig 2: Approaches for different webpages

2.3 *Reddit as a Research Data Source*

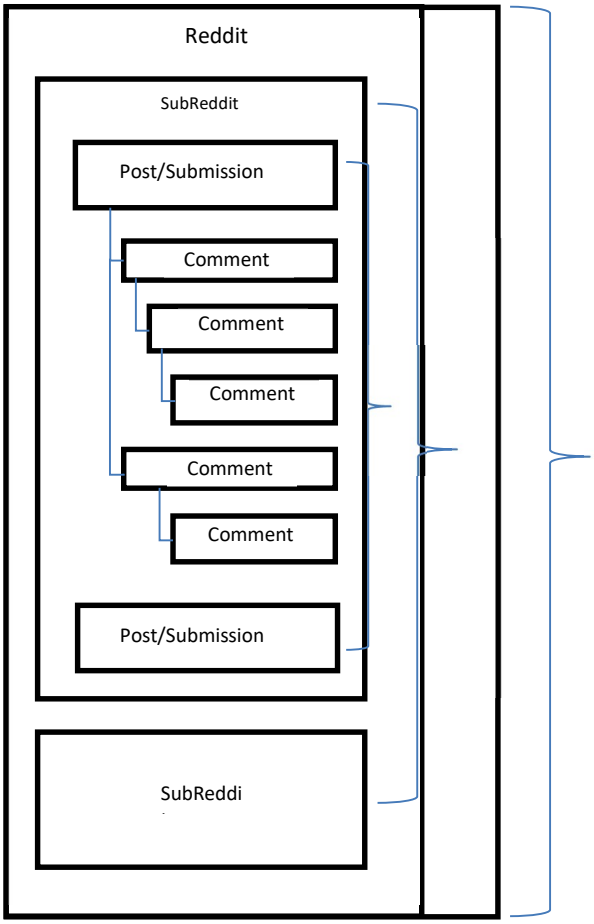

Fig 3: Reddit Thread Structure

Reddit is increasingly viewed as a premier source for academic inquiry. A systematic review of 727 study(2010–2020) indicates a steady rise in Reddit-based research across Computer Science, Sociology, and Public Health.Methodologies: The dominant methodologies involve quantitative computational methods, specifically NLP and Machine Learning.

Data Access: There is a dichotomy in access methods. While the official Reddit API provides structured, policy-compliant access, many researchers (over 90% in some studies) prefer the Pushshift API due to its access to historical archives and more flexible rate limits, despite concerns over deleted data and validity.Topic Coverage: Reddit covers a vast majority (79%) of topics found in Global Google Trends, validating its utility for real-time trend detection.

2.4 *Educational and Ethical Context*

Dogucu and Cetinkaya,argue for the inclusion of web scraping in Data Science curricula. They contend that classroom datasets are often pre-cleaned, giving students a false sense of reality. Scraping introduces students to the "messy" world of unorganized, inconsistent data, fostering critical thinking about data quality and ethics.Ethically, the literature warns that technical capability does not imply permission. Researchers must navigate "hermeneutical imperatives" and respect terms of service. Since Reddit data may contain sensitive behavioral information (e.g., mental health discussions), scraping designs must include anonymization and aggregation measures at the extraction stage.

2.5 *Comparative Analysis of Tools*

A comparative study of scraping tools (BeautifulSoup, Scrapy, Selenium, OctoParse) reveals trade-offs:Scrapy: Best for runtime and memory usage.Selenium: Essential for JavaScript-heavy pages but resource-intensive.Desktop Tools (OctoParse): More user-friendly for non-programmers but lack the community support and flexibility of Python libraries.

3. Research Gaps

Despite the existence of numerous scraping tools, the review identifies a number of critical gaps that this project attempts to fill:

Dynamic Complexity of Content: Modern websites using infinite scrolling or AJAX defeat traditional methods. Though Selenium is available, the main challenge is handling frequent layout changes and interactive elements.

API Limitations vs. Scraping Necessity: While officially provided APIs (Tier 1) ensure reliability, depth, and historical range are limited. For very niche data, custom scraping in a more ad hoc manner is required; however, this is less stable. What it really needs is a hybrid model of sorts to bring everything together nicely.

Data Quality and Standardization: The extracted social data is intrinsically noisy, featuring a low signal-to-noise ratio. There is a lack of automated pipelines that integrate immediately extraction with cleaning and normalization (e.g., stop word removal, duplicate detection) before storage.

Integrated frameworks: Although there are tools for each of these functions, there is a serious lack of frameworks that would put together data extraction with the semantic structuring necessary for business or academic workflow.

4. Proposed Work and Methodology

4.1 *Proposed Approach and Architecture*

The project proposes a Python-based application designed to programmatically interact with Reddit. The solution is structured based on a Decision Tree logic for identifying which extraction method to use depending on the nature of the target: Static versus Dynamic.

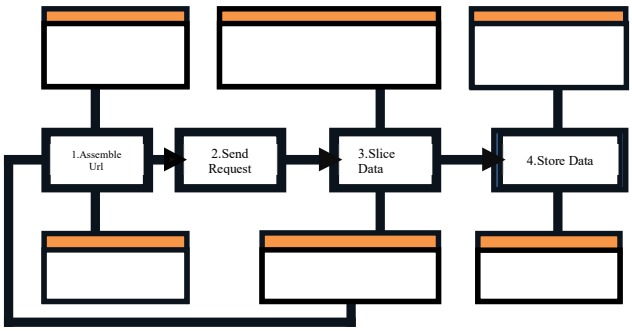

Fig 4: Architectural workflow

The system consists of three basic components:

Data Acquisition Layer: This will handle connection and retrieval using PRAW- Python Reddit API Wrapper, along with standard requests. It ensures OAuth compliance and handles API rate limiting.

- This layer acts as the gateway between the Reddit ecosystem and your local environment.
- **Authentication & Security:** Utilizes OAuth 2.0 via PRAW to ensure secure, authorized access to the Reddit API.

- **Resilience Logic:** Implements **exponential backoff** and "sleep" timers to gracefully handle Reddit's 429 (Too Many Requests) errors and API rate limits.
- **Stream vs. Batch:** Supports both real-time comment streaming (using praw.models.util) and historical batch fetching for specific subreddits or keywords.

Data Processing Layer: This uses pandas for the manipulation of raw JSON into DataFrames. It cleans, extracts features such as timestamp conversion, and structures the data**.**

- This is where raw, nested JSON data is turned into actionable intelligence.
- **Normalization:** Flattens complex, nested JSON objects (like comment trees) into a tabular format using pandas.json_normalize.
- **Feature Engineering:** * Converts Unix timestamps into human-readable ISO-8601 datetime objects.
    - Calculates "Engagement Ratios" (Upvote ratio vs. Comment count).
    - Flags "Stickied" or "Distinguished" posts for priority analysis.
- **Data Sanitization:** Removes duplicates (using Submission IDs), handles missing values (NaN), and strips irrelevant HTML character entities from body text.

Storage & Output Layer*:* Responsible for the persistent storage through the export of the structured DataFrames into CSV or SQL formats for interoperability.This layer ensures that processed data is ready for downstream consumption.

- **Relational Storage (SQL):** Uses SQLAlchemy to interface with SQLite or PostgreSQL, allowing for complex relational queries and indexing on author_id or created_utc.
- **Flat-File Export (CSV/Parquet):** Supports high-speed exports to CSV for spreadsheet use, or **Apache Parquet** for optimized storage size and faster read times in machine learning workflows.
- **Schema Enforcement:** Ensures that data types remain consistent (e.g., ensuring IDs are strings and scores are integers) to prevent "broken" data during future imports.

Monitoring & Logging:

- To make the system "production-ready," you should include a layer for oversight:
- **Logging:** Tracks successful fetches and logs specific API errors to a .log file for troubleshooting.

- **Validation:** A final check to ensure that the number of records retrieved matches the number of records stored.

4.2 *Implementation Workflow*

Authentication -The Gateway: The authenticating scraper registers a Reddit application for their client_id and client_secret, then uses this with PRAW to authenticate using the provided user agent string of the scraper running to instate a session.

Target Definition: The user can specify, among other parameters, the subreddits to target-here r/dataisbeautiful-the timeframe, which will be the last 24 hours, the sorting-hot or new-and the fetch limits.

Extraction Loop: The scraper starts executing API calls, for instance, reddit.subreddit.top(). Automatic pagination is supported: it iterates through sets of data until a limit is reached.

Extraction of Data Fields: It extracts, for each posting, the following critical metadata - title, selftext, score, num_comments, created_utc, and permalink.

The Data Processing Pipeline: "The Clean-up"

Handling Missing & Null Values

- Reddit data is often "sparse." For example, a "Link Post" has no body text (selftext), while a "Text Post" does.
- Imputation: Filling NaN values in the selftext column with empty strings ("") to prevent errors in text analysis.
- Filtering: Dropping rows where critical metadata (like author or id) is missing due to deleted content.

. Temporal Normalization

- Reddit's API provides time in Unix Epoch format (e.g., 1704701234).
- Standardization: Converting Unix integers to datetime objects.
- Localization: Adjusting timestamps to a specific timezone (e.g., UTC) to ensure consistency when merging data from different regions.

Text Sanitization (NLP Preparation)

- Raw text from Reddit is often "dirty" with formatting artifacts.
- HTML Unescaping: Using html.unescape() to turn & into & and < into <.
- Markdown Removal: Stripping out Reddit-specific Markdown like [links](url), **bold**, and > quotes if the goal is pure text analysis.
- Whitespace Trimming: Removing trailing newlines (\n) and extra spaces that occur during copy-pasting.

Categorical Encoding & Type Casting

- To save space and improve query speed in the Storage Layer:
- Boolean Mapping: Converting "Is Original Content" or "Over 18" fields into binary True/False (1/0).
- Downcasting: Changing 64-bit integers to 32-bit where possible to reduce the memory footprint of large DataFrames.Reddit data can be repetitive if you run your script multiple times.ID Validation: Using df.drop_duplicates(subset='id') to ensure that even if a post was scraped twice, it only appears once in your final CSV or Database.

Converting Timestamps: Unix timestamps represented by created_utc get converted to human-readable objects**.**

Preprocessing of Text: This removes special characters and handles emojis to make text ready for NLP.

Error Handling: The missing data-deleted users/posts are handled with default values so that the pipeline doesn't fail. The result is a data set which is validated, hence reusable. It is stored as CSV in order to ensure maximum interoperability with common analytical tools such as R, Excel, and Tableau. This structured data forms the basis of the project's second phase: performing sentiment analysis or topic modeling on the collected discourse**.**This reflects techniques shown to reduce extraction errors in multilingual and low-quality data labels.

5.Comparative analysis of existing approaches

S. No.	Study	Platform / Focus	Methodology	Year
1	YouTube Data Tools	YouTube	API Extraction	2024
2	Netlytic	Multi-platform	API Analysis	2023
3	Phantom Buster	LinkedIn / TikTok	Browser Automation	2025
4	Octoparse	Instagram / X	Visual Scraping	2026
5	Apify	TikTok / Meta	Cloud Automation	2026

1. *The "Post-API" Transition*
From the table, there is a noticeable chronological progression in terms of methodology. The earlier tools (2021-2023) such as Netlytic and Twint were heavily dependent on official APIs or network requests. However, the 2024-2026 tools (such as PhantomBuster and Apify) lean towards Browser Automation and Cloud Automation. This indicates that with the restriction of official APIs by platforms (such as X/Twitter and Meta), the community has moved towards emulating human behavior to ensure data retrieval.

2. *Accessibility vs. Technical Control*
From the methodologies, there is a "user-capability gap" identified:
No-Code Tools: Octoparse (Visual Scraping) and PhantomBuster (Browser Automation) emphasize accessibility for researchers who lack advanced programming knowledge.
Developer-Centric Tools: Apify and Snscrape provide advanced technical control through scripting, which is required for overcoming contemporary anti-bot protection systems that no-code tools may lack.
3. *The Move Towards Managed Infrastructure*
The latest technologies (2025-2026) emphasize that "Data Scraping" is no longer a matter of HTML retrieval; it's a matter of Infrastructure Management. The emphasis on Proxy

Rotation (Bright Data) and Cloud Automation (Apify) shows that the biggest problem in 2026 is not "how to scrape," but "how to avoid being blocked." Contemporary research needs a proxy management and cloud scaling layer that was not as important in the "Legacy" era of 2021.

5.1 *Critical Discussion*

The proposed system effectively addresses the "Redundancy Challenge" identified in the literature. Social media data is plagued by redundancy (quoted replies, identical links). By implementing a processing layer *during* the scraping phase—rather than after—the project minimizes the storage of noise.

Furthermore, the choice of PRAW over direct HTML scraping for the primary acquisition layer aligns with the "API-Scraping Hybrid Model" suggested by recent research. This ensures compliance with Reddit's rigorous API terms while maintaining the ability to process the data flexibly using Python's data science stack.

The integration of Selenium (as noted in the comparative studies) is reserved for scenarios where API access might be insufficient or where visual rendering is required, though the primary proposal focuses on PRAW for its efficiency and structured JSON output. This decision matrix (Fig 4.1) demonstrates a sophisticated understanding of resource management—choosing the "lighter" API method where possible and the "heavier" browser automation only when necessary.

6. Conclusion and Future Scope

6.1 *Conclusion*

The Reddit Web Scraper project represents a robust implementation of modern data engineering techniques. It successfully automates the Extract, Transform, Load (ETL) process for social media data, converting unstructured user-generated content into high-quality, structured datasets. By leveraging libraries like PRAW, BeautifulSoup, and pandas, the system overcomes manual data collection limitations and provides a scalable solution for academic and market research. The project not only demonstrates technical proficiency in handling APIs and data parsing but also adheres to the ethical necessities of digital research.

6.2 *Future Work*

To enhance the utility of this tool, several future developments are proposed:

- NLP Integration: Directly integrating machine learning pipelines for automated sentiment analysis and topic modeling within the scraper.
- Visualization: Developing an interactive front-end dashboard for real-time trend visualization, making the insights accessible to non-technical users.
- Cloud Deployment: Automating deployment as a cloud service to allow for continuous, server-side monitoring of trends without local machine dependencies.
- Cross-Platform Expansion: Extending the scraping logic to other platforms (e.g., X/Twitter, Meta) to allow for comparative social trend analysis.

ACKNOWLEDGMENT

The authors would like to express their sincere gratitude to the faculty and mentors of the Department of Computer Science and Engineering for their guidance and support throughout this research. Special thanks are extended to the project supervisor for valuable insights and continuous encouragement during the development of this work. also acknowledge the use of open-source tools, research resources, and publicly available dataset that contributed to the successful implementation of Scrape-IT

REFERENCES

1. Hegadi, R. S. (2010). Image Processing: Research Opportunities and Challenges. *National Seminar on Research in Computers*, Bharathiar University.
2. Jünger, J. (2023). Scraping social media data as platform research: A data hermeneutical perspective. *Digital Communication Research*, 12, 427-441.
3. Dewi, L. C., et al. (2019). Social Media Web Scraping: using Social Media Developers API and Regex. *Procedia Computer Science*, 157, 444–449.
4. Lotfi, C. (2021). Web Scraping: Techniques and Applications. *Procedia Computer Science*.
5. Muthuseshan, G. (2019). Personalized Content Extraction and Text Classification Using Effective Web Scraping Techniques. *International Journal of Web Portals*, 11(2), 41–52.
6. Glez-Peña, D., et al. (2013). Web scraping technologies in an API world. *Briefings in Bioinformatics*, 15(5), 788-797.
7. Proferes, N., et al. (2021). Studying Reddit: A Systematic Overview of Disciplines, Approaches, Methods, and Ethics. *Social Media + Society*.
8. Arık, K. (2022). Social Media Content Review of Popular MMORPG Games: Reddit Comment Scraping and Sentiment Analysis. *Journal of Emerging Computer Technologies*, 2(1), 13-21.
9. Puthea, K., et al. (2024). Leveraging web scraping and stacking ensemble machine learning techniques to enhance detection of major depressive disorder. *Social Network Analysis and Mining*, 14(1), 239.
10. Thomas, D. M., & Mathur, S. (2019). Data Analysis by Web Scraping using Python. *ResearchGate/Amity University*.
11. Dogucu, M. (2020). Web Scraping in the Statistics and Data Science Curriculum. *Journal of Statistics and Data Science Education*, 28(3), 324-334.
12. Barbera, G., et al. (2023). The Value of Web Data Scraping: An Application to TripAdvisor. *Marketing and Management of Innovations*, 2, 209-220.
13. Khder, M. A. (2021). Web Scraping or Web Crawling: State of Art, Techniques, Approaches and Application. *International Journal of Applied Science and Computation*.
14. Kahlon, N. K., & Singh, W. (2024). Comparative Analysis of Web Scraping Tools for Low-Resource Language Text. *International Journal of Engineering Trends and Technology*, 72(1), 284-299.

15. Pereira, R. C., & Vanitha, T. (2015). Web Scraping of Social Networks. *International Journal of Innovative Research in Computer and Communication Engineering*, 3.
16. Matta, P., et al. (2020). Web Scraping: Applications and Scraping Tools. *International Journal of*
17. *Advanced Trends in Computer Science and Engineering*, 9(5).
18. Zhao, B. (2017). Web scraping. In *Encyclopedia of Big Data*. Springer International Publishing.Khan, N., & Khan, M. F. (2015). Web Scraping Techniques and Applications: A Literature Review. *International Journal of Engineering and Applied Sciences*.
19. Shivane, A., et al. (2023). Issues and Challenges of Web Scraping. *The Online Journal of Distance Education and eLearning*, 11(1).
20. Krishna, N., et al. (2022). A Study Of Web Scraping. *International Journal of Engineering Research in Computer Science and Engineering*.

.

