Cloud Computing and Data Science
https://ojs.wiserpub.com/index.php/CCDS/

UNIVERSAL WISER
PUBLISHER

Research Article

Automating Data Collection to Support Conflict Analysis: Scraping the Internet for Monitoring Hourly Conflict in Sudan

Yahya Masri[1], Anusha Srirenganathan Malarvizhi[1,2], Samir Ahmed[1], Tayven Stover[1], Zifu Wang[1,3*], Daniel Rothbart[4], Mathieu Bere[4], David Wong[2], Dieter Pfoser[2], Chaowei Yang[1,2]

[1]NSF Spatiotemporal Innovation Center, George Mason University, Fairfax, VA, 22030, USA
[2]Department of Geography and Geoinformation Science, George Mason University, Fairfax, VA, 22030, USA
[3]Center for Geographic Analysis, Harvard University, Cambridge, MA, 02138, USA
[4]Jimmy and Rosalynn Carter School for Peace and Conflict Resolution, George Mason University, Arlington, VA, 22201, USA
E-mail: zifu_wang@fas.harvard.edu

Received: 12 August 2025; **Revised:** 7 November 2025; **Accepted:** 13 November 2025

Abstract: The ongoing conflicts in Sudan have escalated rapidly, highlighting the critical need for timely and accurate data to inform humanitarian responses, policy decisions, and research needs. While existing datasets such as the Armed Conflict Location & Event Data Project (ACLED) and the Uppsala Conflict Data Program Georeferenced Event Dataset (UCDP GED) provide valuable insights into conflicts, they suffer from update delays and lack source transparency, which hinders timely incident reporting and comprehensive analysis. To address these limitations, we developed a web scraping toolset that collects hourly data from the Internet, deploying the tools to support Sudan conflict analysis. The scraped data was used to build an open-access database that houses 6,946 articles as of October 25, 2024, from national, regional, and international sources, offering a transparent and easily accessible resource for further analysis. A case study is presented to demonstrate the scraper's practical application in covering the siege of Sinjah, successfully capturing spatial and temporal events within a conflict zone. The scraped data outperformed the UCDP GED in capturing incidents but missed some smaller-scale incidents recorded by ACLED, highlighting areas for improvement through expanding source diversity. Overall, the scraper demonstrates great potential for improving conflict monitoring and could be further enhanced by incorporating additional sources and automation techniques.

***Keywords*:** web scraping, conflict data collection, sudan conflict, data automation, news mining, spatiotemporal data

1. Introduction

Several ongoing conflicts worldwide, such as those in Ukraine, Sudan, and the Middle East, have highlighted the urgent need for timely and comprehensive conflict monitoring to understand and respond to the resulting complex humanitarian crises [1]. Conflict monitoring involves systematically observing and analyzing conflicts to track their development, understand their dynamics, and assess their impacts [2]. This process is crucial for humanitarian organizations, researchers, and policymakers, as it enables them to assess needs, allocate resources, and develop strategies for relief and resolution [3-4]. It relies on gathering data from a variety of sources, including news reports, social media, government statements, and local observers [5].

DOI: https://doi.org/10.37256/ccds.7120268226

One of the most pressing conflicts being monitored today is the ongoing war in Sudan. Since the outbreak of violence in April 2023 between the Sudanese Armed Forces (SAF) and the Rapid Support Forces (RSF), the situation has rapidly escalated, creating a complex civil war crisis. The violence has not only drawn regional and international attention but has also highlighted the urgent need for the effective and timely gathering of conflict data to inform humanitarian responses.

The Sudan conflict's complexity and rapidly evolving nature underscore the broader challenges in gathering reliable data from conflict zones. In such volatile environments, obtaining timely and accurate information is particularly difficult due to restricted access, government censorship, and the dangers journalists and observers face [6]. In addition to government efforts, organizations such as Armed Conflict Location & Event Data Project (ACLED) [7], Uppsala Conflict Data Program (UCDP) [8], the Global Database of Events, Language, and Tone (GDELT) [9], and the Integrated Crisis Early Warning System (ICEWS) [10] offer extensive datasets on global conflicts.

These initiatives enhance the availability of reliable conflict data, enabling a more comprehensive analysis of conflict dynamics and facilitating better-informed decisions by policymakers, humanitarian organizations, and researchers. While ACLED, UCDP, GDELT, and ICEWS provide valuable datasets and insights into global conflicts, there are limitations to relying solely on these sources. Existing datasets often depend on specific data collection methods and sources, which may lead to gaps in coverage, particularly in fast-evolving conflict situations or in regions with limited reporting [11]. Additionally, the time lag between data collection, analysis, and publication can result in delays that hinder timely response efforts.

To address these challenges, web scraping from news articles has emerged as a powerful tool to support conflict monitoring, particularly in fast-evolving situations where access to timely data is critical [12]. Web scraping involves the automated extraction of data from websites, enabling researchers to gather and organize large volumes of information efficiently [13]. In situations where data sources are fragmented or updated frequently, web scraping proves valuable, allowing for continuous, real-time updates of conflict data from a variety of reputable sources [14]. In addition to web scraping, Application Programming Interfaces (APIs) offer another important method for accessing data. Data providers design APIs to enable structured access to their datasets, allowing users to query specific information in a clean and well-documented fashion [13]. By combining the precision of API data with the flexibility of web scraping, researchers can obtain more robust and comprehensive datasets [15].

While existing conflict datasets such as ACLED, UCDP, GDELT, and ICEWS are valuable, they often need revision regarding timeliness and coverage, especially in rapidly evolving conflict situations. Furthermore, the lag between data collection and publication delays crucial decision-making, underscoring the need for more agile and immediate conflict monitoring tools. Web scraping can address these gaps by providing continuous updates from various sources, ensuring immediate and comprehensive conflict data. Considering these advantages, this research focuses on the Sudan conflict and aims to develop an extensive web scraping toolset capable of efficiently extracting, organizing, and storing conflict data. The toolset is used to build an open-access conflict dataset, providing transparent, timely, and reliable data for analysis. The current study has the following four objectives:

1. To design and implement a web scraping toolset to retrieve conflict data from credible sources, including APIs and dynamic and static websites, ensuring broad and diverse hourly data collection.

2. To build and maintain an open-access database that houses the conflict data collected, providing researchers and policymakers with a reliable and accessible resource for conflict analysis and decision-making.

3. To apply the toolset to a real-world use case involving the siege of Sinjah, the capital of Sennar state in Sudan, demonstrating its effectiveness in collecting and organizing conflict data to offer valuable insights for further analysis.

4. To validate the accuracy and comprehensiveness of the toolset's output through comparison with established conflict datasets, such as ACLED and UCDP, by focusing on the number of incidents reported within a specific time interval, ensuring their reliability in capturing conflict data for research and decision-making.

The structure of the paper is organized as follows. Section 2 surveys conflict data resources and webscraping foundations, including the theoretical framing that guides our approach. Section 3 describes prescraping source discovery and selection. Section 4 details the methodology spanning the end-to-end workflow, automation, post-processing, and data-quality controls and Section 5 reports experiments and results, including descriptive statistics, a Sinjah siege case study, comparisons with ACLED and UCDP Georeferenced Event Dataset (UCDP GED), and performance testing. Section 6 offers a discussion of implications and limitations and finally, Section 7 provides the conclusion along with considerations for future research.

2. Literature review

This section situates our work within two strands of prior research. Section 2.1 reviews conflict event datasets (e.g., ACLED, UCDP GED, GDELT, ICEWS), emphasizing strengths and known limitations in transparency and update frequency. Section 2.1.1 outlines the theoretical framing that guides the source-attributed, event-ready acquisition. Section 2.2 then surveys web-scraping foundations, tooling, automation patterns, data-quality controls, and ethical/legal considerations, relevant to high-frequency collection from diverse news outlets. Together, these strands inform our source selection strategy and the end-to-end methodology presented later in the paper.

2.1 *Conflict data*

Existing datasets from ACLED, UCDP GED, GDELT, and ICEWS are critical in supporting the efforts of governments, policymakers, and humanitarian organizations to understand conflict dynamics and respond effectively to crises worldwide [16-17]. Introduced by Raleigh et al. [7], ACLED is a leading conflict dataset that reports data on political violence, riots, and protests worldwide. As of 2024, ACLED continues to provide detailed information on various forms of conflict, including battles, civilian targeting, and protests in regions such as Africa, Asia, and the Americas. Similarly, UCDP GED is recognized for its comprehensive data on organized violence, covering state-based conflicts, non-state conflicts, and one-sided violence. The UCDP GED is a valuable tool for conflict analysis at substate and country-year levels, known for its historical depth, documenting conflicts from the 1970s onwards [8]. In contrast, GDELT offers a different approach, using machine learning and natural language processing to analyze global news coverage. GDELT categorizes events by location and tone, providing multiple daily updates for real-time analysis [9]. While GDELT is known for its vast, real-time collection of global event data, ICEWS focuses on forecasting political instability through an advanced computational social science approach that includes agent-based and logistic regression models [10]. Recent advancements in automated event extraction and categorization, as evaluated, have demonstrated the efficiency of such tools in producing reliable data at a fraction of the cost of human coding [18]. These methods are particularly beneficial for large-scale event data collections like GDELT, which aim to provide near realtime insights into global conflicts.

However, ACLED and UCDP GED face significant limitations, particularly regarding the transparency of some of their sources. A critical issue is the inaccessibility of the original article or the news media link from which their data is derived [11]. This lack of access makes it difficult for users to verify the reported incidents or delve deeper into the content, thereby limiting the potential for independent analysis and validation. UCDP attributes the absence of Uniform Resource Locator (URLs) to copyright restrictions, a reasoning also shared by ACLED [11]. This constraint prevents these organizations from sharing URLs in their datasets, thus limiting comprehensive verification efforts and complete transparency. Furthermore, neither ACLED nor UCDP provide full article content, except a few sentences or a single-sentence summarization, which restricts the depth of analysis that can be conducted and may limit the ability to fully capture the context and complex nuances of the reported incidents.

ACLED updates its data every week, with occasional data publication pauses, resulting in delays in postevents in near real-time. Consequently, the timeliness of ACLED's data can be compromised, presenting challenges to tracking rapidly evolving conflicts and developments. In addition, UCDP updates its data monthly, likely compromising the timeliness, immediacy, and relevance of the data, and thus potentially impacting the effectiveness of using the data in real-time analysis and decision-making. On the other hand, GDELT's data can be considered near real-time as it integrates the flow of new data into their database every 15 minutes [12]. Although GDELT's data is openly accessible, its vast size and complexity present challenges in parsing and cleaning. Unlike curated datasets like ACLED and UCDP, GDELT's data requires additional effort to retrieve and analyze relevant information [12]. Accessing fulltext articles through GDELT involves a process called're-hydration,' where users must use metadata (URLs) to retrieve original articles, adding another layer of complexity [19]. This makes large-scale or in-depth analysis more difficult and time-consuming.

In response to the critical limitations in transparency, timeliness, and data accessibility identified in existing datasets like ACLED, UCDP GED, and GDELT, we developed the web scraper toolset.

2.1.1 *Theoretical framing of conflict event data*

Rather than a single "conflict data model," event datasets are developed under different inclusion rules and purposes [7-8, 21-22]. Broad, all-incident datasets (e.g., ACLED) aim to capture diverse political-violence events [7], while specialized datasets (e.g., Global Terrorism Database; GTAC Terrorist Incidents Record) focus on particular incident types and actors under narrower criteria [21-22]. The Sudan scraper follows the specialized model: it is designed to systematically collect and archive source-attributed reporting within a specific conflict domain, improving temporal resolution, reproducibility, and transparency for subsequent event analysis. Although the current system focuses on data acquisition rather than direct event extraction, its design draws on theoretical principles from event-data research and spatiotemporal conflict analysis [17]. These frameworks emphasize that observable reports can serve as proxies for underlying conflict dynamics when appropriately filtered and verified [23]. By capturing detailed temporal and spatial metadata, the scraper enables future analyses of escalation, diffusion, and clustering once event-level parsing is implemented. In this sense, the system is theory-informed, serving as an infrastructure that supports the empirical testing of models addressing how conflict intensity and conflict spread evolve over time.

Finally, this scraper operates within a broader ethical data-collection framework grounded in transparency, legality, and accountability [24]. Only publicly accessible articles are collected, and all source URLs are stored to facilitate manual, independent verification. These practices align with emerging digital-conflict informatics and responsible webdata acquisition [25].

2.2 *Web scraping*

Web scraping has become an indispensable tool in the era of big data, enabling rapid, large-scale data collection from online sources [26-28]. Web scraping automates the extraction of information from websites, enabling researchers and organizations to gather large amounts of data compared to traditional manual collection methods [29-31]. This method is particularly advantageous in fields requiring high-frequency and immediate data collection, such as the social sciences, media analysis, and conflict monitoring [32-33]. For example, researchers leverage web scraping to track evolving logistics, public sentiment and monitor ongoing geopolitical conflicts or shifting opinions on media releases [34-35]. In logistics, Tee et al. [36] demonstrated how web scraping can efficiently collect and process timedependent traffic data, providing a low-cost alternative to paid services for improving vehicle routing, highlighting the potential and current applicability of web scraping for broad industry adoption. Data collected through web scraping of socially relevant platforms, such as Twitter (X) or news outlets, can provide real-time data that can be scraped during critical events, such as conflicts or crises, enabling researchers to monitor developments, public reactions, and shifts in sentiment as these situations unfold [37]. Additionally, web scraping is a robust method for extracting structured and unstructured data, facilitating the analysis of trends, behaviors, and patterns across diverse sectors. For example, Lan et al. [38] utilized cloud-based web scraping to automatically collect and standardize COVID-19 case data from various global sources, overcoming challenges related to inconsistent formats and publication methods.

In recent years, this ability to gather vast amounts of data has become increasingly important in Machine Learning (ML) and Artificial Intelligence (AI), where diverse and structured data are essential for building robust models. Web scraping supports the creation of robust models by providing diverse datasets needed for predictive tasks [39]. Natural Language Processing (NLP) has particularly benefited from web scraping, as it provides vast amounts of unstructured text data for training models. Qi and Shabrina [40] emphasized the importance of data preprocessing in sentiment classification, demonstrating how web-scraped data provides valuable insights into public sentiment during global events like the COVID-19 pandemic. Similarly, Miranda et al. [41] explored the evolution of sentiment in Spanish COVID-19 pandemic tweets using web-scraped data and a fine-tuned BERT model, further highlighting the role of web scraping in ML in understanding public emotions during crises.

Konstantinidis et al. [42] used web scraping in the financial sector to collect over 550,000 news articles, which were analyzed using NLP models to develop sentiment-based investment portfolios. Zaytoon et al. [43] utilized web scraping to create multilingual datasets, like the AMINA dataset for Arabic news articles, which supports NLP tasks such as classification and topic modeling.

While web scraping offers immense potential for data collection in various fields, the effectiveness of these applications heavily depends on the choice of tools used to gather and process the data. Many studies have highlighted the critical role of web scraping tools in efficiently gathering data from diverse web sources. Beautiful Soup is commonly used for its simplicity in parsing Hypertext Markup Language (HTML) and eXtensible Markup Language (XML), making it ideal for navigating inconsistent webpage structures [44]. Selenium excels in automating interactions with dynamic content rendered by JavaScript, which is beneficial for scraping dynamic web pages like stock market platforms [45]. Scrapy is favored for large-scale scraping due to its asynchronous downloading and built-in pipelines, demonstrated in e-commerce data extraction and ontology building [46]. Lxml stands out for its fast and efficient parsing of structured websites, making it highly scalable for complex web structures [47]. Puppeteer and Playwright offer modern alternatives to Selenium, with enhanced headless browsing and interaction capabilities for scraping JavaScript-heavy sites [48].

In this review, several critical challenges associated with web scraping were identified. Ethical and legal concerns arise from the need to balance automated data collection with respect for privacy, intellectual property, and website regulations [49-50]. Ensuring data quality can be difficult, as the lack of human verification in web scraping introduces potential errors, necessitating post-collection reviews and quality control mechanisms [51]. Additionally, the rise of bot traffic complicates data integrity by obscuring meaningful patterns, requiring advanced algorithms to detect and mitigate unauthorized scraping [52].

To address the challenges of web scraping, we incorporated rigorous data validation protocols and implemented quality control reviews to ensure the accuracy and reliability of the collected data. Ethical and legal considerations were also prioritized to ensure compliance with website terms and respect for privacy, minimizing the risk of violating intellectual property rights. This approach ensures that the data collected is both high-quality and ethically sourced. The scraper automatically extracts and stores full articles with links, overcoming the challenge of limited access to the source material. Unlike traditional datasets with delayed updates, our scraper operates hourly, providing frequent information as events unfold. Overall, the scraper streamlines data collection, allowing for the extraction of key information such as source URLs, articles, and images, which can then be used for analysis.

3. Pre-scraping data collection

The objective of the pre-scraping data collection stage was to compile a reliable list of sources reporting on the Sudan conflict for integration into the scraper. To achieve this, a dual approach of web scanning and LexisNexis filtering was followed to build a reliable and comprehensive list of sources reporting on the Sudan conflict for integration into the scraper. An extensive web scan was conducted utilizing various techniques and search engines to locate relevant information on the Sudan conflict over a specific period. For instance, tailored Google searches, such as “Sudan AND (kill OR wound OR injure) after: 2024-08-01 before: 2024-08-08” and Bing searches, such as “Sudan” AND “kill” AND “injure” were employed. By manually filtering the date using the filter option, articles within a defined time frame were targeted, focusing on incidents matching specific conflict-related keywords. This method helped pinpoint articles likely to contain reports of significant events.

Upon gathering potential articles, each was manually reviewed for their relevance and direct connection to the conflict. Once potential articles were identified, the next step involved examining the content in detail.

Subsequently, additional articles published by the same source were explored to assess internal consistency in reporting. However, to ensure not only internal validity but also the reliability of the information, we cross-referenced key events from the articles with other independent and reputable sources. By triangulating these reports with external data, we confirmed whether multiple outlets reported the same incidents, thus validating the reliability of the original source. This combination of internal consistency checks and external triangulation ensured that only sources demonstrating both accuracy and consistency were selected for the final list. The goal was to prioritize sources that consistently reported detailed and verifiable updates on the conflict.

Additionally, sources were selected for web scraping only if legally permitted, ensuring ethical compliance. An evaluation was conducted to determine if a website had substantial coverage of Sudan’s conflict and if it focused mainly on Sudanese events or global events. Compliance is maintained as an ongoing process, with periodic reviews of each source’s access policies. If a site changes its terms or restricts automated access, it is promptly removed from the pipeline. Only publicly available information is collected, and attribution is always preserved to maintain transparency and uphold ethical data use.

The second step involved leveraging the LexisNexis database as a secondary tool to refine our source selection. Searches were filtered by date, specifically targeting publications from the beginning of the conflict, April 15, 2023, to the present day. Using LexisNexis's filtering options, results were narrowed down by source name, automatically sorting the sources in descending order by the number of articles from the filtered search.

These two approaches, web scanning and LexisNexis filtering, were part of continuous evaluation and refinement by identifying patterns in coverage during the scraping and analysis of articles. The refinement helped us recognize reliable sources for reporting on the Sudan conflict and expand our list by incorporating new, valuable contributors. To ensure quality and relevance, consultation with a conflict resolution professional was conducted to verify source credibility and recommend additions. By regularly assessing the performance and credibility of sources, a comprehensive and reliable list was maintained, prioritizing those that provided frequent, detailed updates. This approach, combined with web scanning and LexisNexis filtering, built an effective and robust source list for the scraper. Much like the modular design of CoviRx [53], which allows for changes in components without altering the overall system, the Sudan scraper is designed to easily integrate new data sources and adapt to evolving conflict data requirements.

Table 1. Overview of sources used in the Sudan scraper, classified by coverage scope, data retrieval method, and website dynamics

Source	Source coverage	Web scraped or API	Website type
Al Jazeera	International	Web scraped	Static
Al Taghyeer	National	Web scraped	Dynamic
Arab news	Regional	Web scraped	Dynamic
BBC news	International	Web scraped	Dynamic
Darfur 24	National	Web scraped	Static
France 24	International	Web scraped	Dynamic
Ground news	International	Web scraped	Dynamic
Middle East Monitor	Regional	Web scraped	Dynamic
Radio Dabanga	National	Web scraped	Static
Radio Tamazuj	Regional	Web scraped	Static
Sudan News Agency	National	Web scraped	Static
Sudan Tribune	National	Web scraped	Dynamic
Sudan War Monitor	National	Web scraped	Static
The Guardian	International	API	Dynamic
Xinhua	International	Web scraped	Dynamic

Table 1 shows the sources in the scraper and their respective scales, identified through data collection. The classification was based on the geographic coverage of each source, which is categorized into three distinct scales:

1. International: This category encompasses sources that report on global news, extending beyond the African continent. While these sources lack the specific, highly detailed information relevant to the situation in Sudan, they often incorporate overall worldwide statistics, valuable for our data curation.

2. Regional: Sources classified as regional primarily cover the general geographic area surrounding Sudan, including neighboring countries like South Sudan, and broader regional dynamics such as the Sahel Region or the Middle East. Notable examples of regional sources include major outlets such as Middle East Monitor and Arab News.

3. National: National sources exclusively cover news within Sudan. These sources offer the most localized and

detailed information, often reporting on incidents and strategic developments with detailed specificities. Some notable examples of national sources incorporated into the scraper include Sudan News Agency (SUNA), Sudan Tribune, and Radio Dabanga.

By classifying sources based on scale, data retrieval method, and website type, a broad and diverse coverage of the Sudan conflict was achieved. This integration of international, regional, and national perspectives enabled us to capture diverse information, from high-level geopolitical trends to detailed local reports. This layered approach enhances contextual understanding of the conflict, allowing for more comprehensive analyses.

4. Methodologies

This section outlines the end-to-end workflow used to acquire, process, and maintain hourly conflict data. Section 4.1 introduces the scraping pipeline, including source-specific extraction for static and dynamic webpages, keyword-guided relevance filtering, structured text parsing, and database insertion. Section 4.2 details the automation framework for continuous execution, version synchronization, and dependency management. Section 4.3 describes post-processing steps, including database architecture, spatial linkage, and duplicate-resolution logic.

Section 4.4 presents data-quality safeguards addressing completeness, consistency, recency, and misinformation risks. Together, these components provide a reproducible, scalable pipeline for fast-updating, source-attributed conflict data.

4.1 *Workflow diagram*

Figure 1 illustrates the complete web scraping workflow, from selecting data sources and determining scraping methods to filtering, extracting relevant information, and storing results in a database.

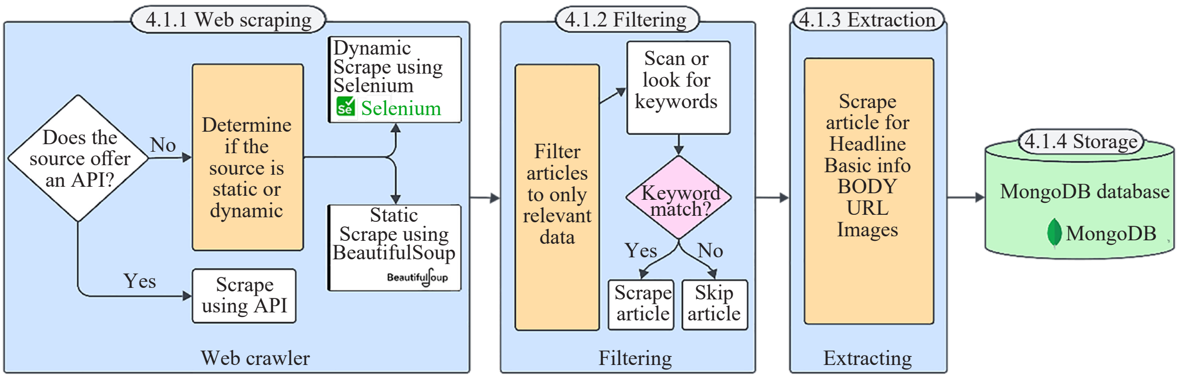

4.1.1 Web scraping
Does the source offer an API?
No
Yes
Determine if the source is static or dynamic
Dynamic Scrape using Selenium
Selenium
Static Scrape using BeautifulSoup
BeautifulSoup
Scrape using API
Web crawler
4.1.2 Filtering
Filter articles to only relevant data
Scan or look for keywords
Keyword match?
Yes
No
Scrape article
Skip article
Filtering
4.1.3 Extraction
Scrape article for Headline Basic info BODY URL Images
Extracting
4.1.4 Storage
MongoDB database
MongoDB

Figure 1. Web scraping workflow includes four steps

4.1.1 *Web scraping*

To gather the necessary data, web scrapers were developed for all chosen sources to extract specific aspects of articles from both structured sources, like APIs and diverse, dynamic sources, like websites. APIs were preferred when available because they provided structured, reliable data that were frequently updated, reducing errors and making analysis simpler. For websites without an API, websites were directly scraped. Websites could be loaded in two ways: statically, with fixed content, or dynamically with content that changed with user interactions. For static websites, Python packages like BeautifulSoup were used to parse HTML and extract content. For dynamic websites, where content was loaded via JavaScript, BeautifulSoup wasn’t suitable. Instead, Selenium, which sent web drivers like ChromeDriver to simulate user interactions, triggered content loading and allowed the content to be collected.

4.1.2 *Filtering*

Several measures were implemented to exclude unrelated content to focus solely on collecting data about the Sudan conflict. First, a list of keywords associated with the Sudan conflict, such as "Sudan," "war," and "destruction," along with other relevant terms, was compiled. The full keyword list used for filtering is publicly available in the project's GitHub repository (keywords.md) to ensure transparency and reproducibility. Headlines of articles from each source were scanned for these keywords. If a keyword was present, the entire article was scraped. If not, it was skipped. Priority was given to scraping from dedicated sections on websites that specifically covered the Sudan conflict, ensuring more targeted data collection. Even within these sections, each article's headline was checked for relevant keywords to avoid collecting unrelated articles, such as those that included South Sudan, which did not align with the focus. This dual approach helped ensure the relevance and specificity of the gathered data.

4.1.3 *Extracting*

To keep the database organized, only specific elements from each article were scraped to efficiently extract relevant data and prevent clutter. This method made the data easy to access and analyze. For each article, only the essential components, such as the source, headline, date, body text, images, and URL, were collected. After scraping, the data was formatted and timestamped. This systematic approach ensured that the database remained current and well-organized. Furthermore, retaining full text alongside URLs enables subsequent Large Language Model (LLM) pipelines for incident classification and evidence-linked location/date extraction on the same corpus [54-55].

4.1.4 *Storage*

Essential components were stored in a MongoDB database, which facilitates access and analysis of retrieved data. Additionally, the database recorded the exact time each scraper operated, which was crucial for monitoring its performance by tracking the number of articles scraped daily and quickly identifying any issues that might arise.

4.2 *Automation of data scraping*

Figure 2 illustrates the automation of data scraping, from synchronization with the repository to article collection and filtering. The automated web scraping system was structured to manage data collection hourly. Crawler scripts are maintained in a GitHub repository, enabling easy updates and deployment as new requirements emerge. The server uses a secure Secure Shell (SSH) key to synchronize with the repository, ensuring that the latest versions of the scrapers are consistently available for deployment. A CRON job is set to automatically execute the scrapers hourly, ensuring regular and uninterrupted data collection without manual intervention. Before each task, the system verifies that all necessary libraries and dependencies are installed, reducing errors due to missing or outdated components. During data collection, the server gathers articles, ensures relevance, and filters duplicates to prevent redundancy, improving the efficiency and accuracy of the final dataset.

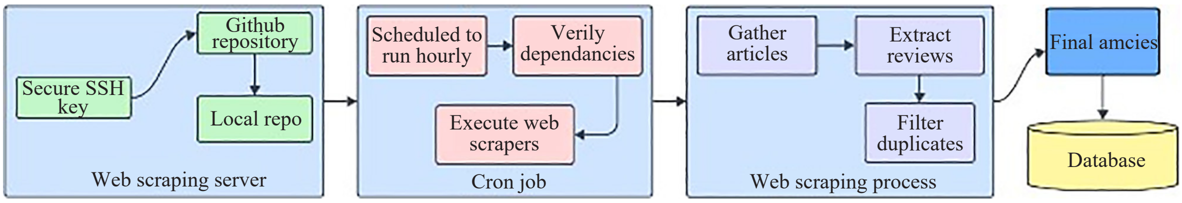


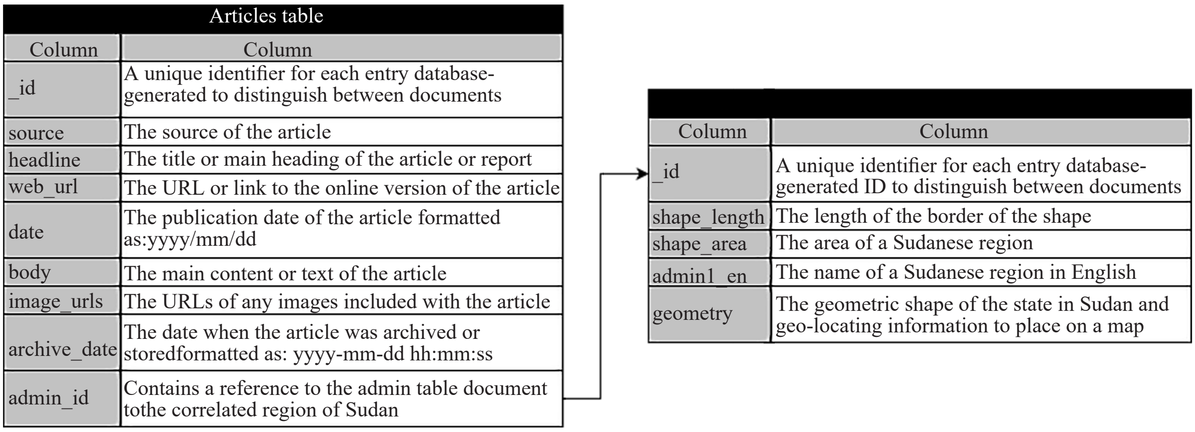
Figure 2. A flowchart representing the process taken to automate the scraping of web pages

4.3 *Post-data processing*

After completing the automated web scraping, the scraped data is stored in the database, and additional checks are performed to remove duplicates to ensure that the final dataset is accurate and free of redundant entries.

4.3.1 *Database design*

Figure 3 presents the document-based structure of a NoSQL database (MongoDB), showing how data is stored in two collections: articles and admin. The "articles" collection stores document fields such as the article's source, headline, web URL, publication date, and content. Each article includes an admin_id field, which refers to a document in the admin collection. The "admin" collection contains Sudan's state-level administrative data (Administrative level 1); the adm1_en row correlates to state names, the geometry column specifies a set of *x*-*y* coordinates depicting the shape of the polygon of a state, the shape_length row indicates the length of the spatial entity if all borders were laid flat, and the shape_area row indicates the area of the polygon. The admin table's relational approach allows each article to be linked to a specific geographic region, facilitating geographic analysis of conflict-related events within distinct Sudanese regions.



Articles table	
Column	Column
_id	A unique identifier for each entry database-generated to distinguish between documents
source	The source of the article
headline	The title or main heading of the article or report
web_url	The URL or link to the online version of the article
date	The publication date of the article formatted as:yyyy/mm/dd
body	The main content or text of the article
image_urls	The URLs of any images included with the article
archive_date	The date when the article was archived or storedformatted as: yyyy-mm-dd hh:mm:ss
admin_id	Contains a reference to the admin table document tothe correlated region of Sudan

Column	Column
_id	A unique identifier for each entry database-generated ID to distinguish between documents
shape_length	The length of the border of the shape
shape_area	The area of a Sudanese region
admin1_en	The name of a Sudanese region in English
geometry	The geometric shape of the state in Sudan and geo-locating information to place on a map

Figure 3. "Articles" and "admin" collections in the DB

4.3.2 *Duplicate handling*

Duplication handling is an important mechanism in the practice of storing and maintaining a database of unique documents and values. Figure 4 illustrates the duplication handling process used for incoming articles.

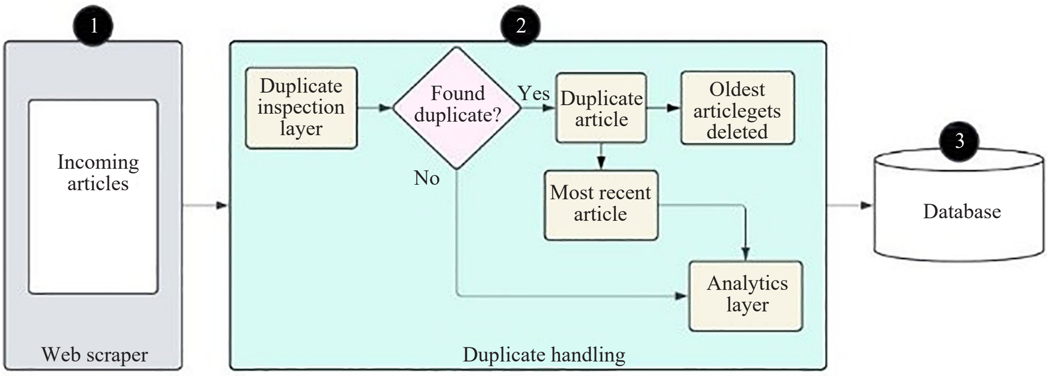


Figure 4. Duplication handler system

Incoming data is first processed through a duplicate detection module, which uses web URLs as the primary identifier for each entry. This approach ensures that all incoming articles are efficiently cross-referenced with existing records in the database to prevent redundancy. If a duplicate is found, the system updates the stored entry with the latest information, ensuring that the most recent version of the article is preserved. When no duplicate is found, the article moves through the analytics layer, the analytics layer keeps track of the number of articles scraped from each source every day, allowing researchers to gain further insights into collecting data related to the Sudan conflict. After passing through the analytics layer, the article is then inserted into the database for long-term storage, ensuring that new and updated information is systematically preserved for future retrieval and analysis. This process ensures that the database remains current, efficient, and free of redundant entries, supporting research on the evolving situation in Sudan.

4.4 *Data quality check*

A robust data quality control process has been established to ensure the reliability and accuracy of data collected by the scraper [56]. This process addresses several key data quality aspects, including completeness, consistency, duplicate handling, recency, and relevance, as well as error detection and correction, as outlined below.

• Completeness: The scraper is configured to ensure that all essential information, such as publication date, image URLs, headline, and content, is captured from each article. Any missing data is then manually reviewed and investigated. This step guarantees that each article's metadata is fully captured for monitoring the Sudan conflict.

• Consistency: Given the dynamic nature of online news reporting, articles are often updated after their initial publication. To maintain the accuracy of stored data, the scraper is designed to revisit previously scraped content to verify consistency with the latest version. This process ensures that any updates to critical details-such as casualty numbers, location details, or changes in the presumed perpetrator-are reflected in the database, preserving the integrity of the information.

• Conflict resolution and misinformation control: Although the current system does not automatically resolve conflicting accounts, a manual validation process is planned to address this limitation. When multiple sources report the same event with differing details, these cases will be manually reviewed and compared across independent, reputable outlets to identify the most consistent version. Reports that remain unverified will be flagged and excluded from aggregated analyses until further corroboration is available. Additionally, when articles are updated or corrected by publishers, these revisions will be manually reviewed to ensure the database reflects the most accurate version. This planned procedure will strengthen the reliability and transparency of the dataset as validation efforts continue.

• Duplication Detection: To maintain dataset integrity and prevent redundancy, the scraper employs a duplication detection system, discussed in Section 4.3.2, using web URLs as unique identifiers. When a duplicate is identified, the existing record is updated with the latest data, ensuring that the database holds only the most recent version of each article. This duplication detection process ensures the database maintains a single version of each news article, preserving dataset integrity.

• Timeliness and Relevance: The scraper is configured to run at regular intervals, capturing new data as soon as it becomes available, ensuring that the database provides an up-to-date representation of the Sudan conflict. Additionally, articles are systematically filtered to ensure relevance; only those articles pertaining to the conflict are processed and stored. This relevance filtering is critical to ensure that the resulting dataset remains focused on conflict-related incidents, avoiding unrelated data.

• Error Detection and Correction: Given the variability in source structures and potential website format changes, the scraper may occasionally encounter errors such as missing data fields or other inconsistencies that affect data collection. The research team regularly reviews outputs and identify issues. Once detected, adjustments are made to the scraper's configuration to reflect the new structure, ensuring it continues to collect data as intended. This manual intervention allows for quick fixes, minimizing downtime.

To improve accessibility for readers from non-technical backgrounds, Table 2 summarizes the main technical features of the Sudan scraper and their direct benefits for conflict analysis.

Table 2. Summary of technical features and their benefits for conflict analysis

Technical feature	What it does	Benefit for conflict analysis	Where described
Source acquisition via APIs & web scrapers (BeautifulSoup for static; Selenium for dynamic)	Pulls articles from structured APIs and JavaScript-rendered pages.	Maximizes coverage across outlet types; reduces missed incidents due to site tech.	§4.1.1
Hourly automation (CRON + repo sync)	Runs all crawlers every hour with versioned scripts.	Improves timeliness; captures fast-moving sequences around key events (e.g., sieges).	§4.2
Relevance filtering (keywords + section targeting)	Screens headlines/sections before full scrape.	Focuses dataset on Sudan conflict; lowers noise for incident detection.	§4.1.2
Structured extraction (source, title, date, body, images, URL)	Normalizes essential fields per article.	Enables reliable event parsing, dating, and traceable sourcing.	§4.1.3
MongoDB storage with admin linkage	Stores articles; links records to ADM1 geography.	Supports map-ready spatiotemporal analysis by state/region.	§4.3.1
Duplicate handling by URL with update-on-match	De-dupes incoming items; updates when stories change.	Preserves a single, most-current record; avoids overcounting incidents.	§4.3.2
Data-quality controls (completeness, consistency, conflict-resolution protocol, timeliness, error fixes)	Periodic rechecks; planned manual review for conflicting reports; routine scraper fixes.	Increases reliability, reduces misinformation risk, keeps data current.	§4.4
Analytics layer for source throughput	Tracks per-source scrape counts/ dates.	Reveals coverage gaps and outlet performance over time.	§4.3.2
Transparency (full text + URLs stored)	Retains original content and links.	Enables verification, reproducibility, and audit trails versus black-box datasets.	§4.1.3, §6
Performance profiling (static vs dynamic pages)	Benchmarks scrape time by site type.	Informs scaling strategy and runtime expectations for multi-country expansion.	§5.4

5. Experiments and discussion

This section evaluates the system’s empirical performance and practical utility. Section 5.1 reports descriptive characteristics of the collected dataset, highlighting distribution across source types. Section 5.2 demonstrates operational use through a spatiotemporal case study centered on the Sinjah siege, including manual incident extraction and event mapping. Section 5.3 compares scraper coverage against ACLED and UCDP GED to assess completeness and recall under real-world conditions. Section 5.4 presents performance benchmarks for static and dynamic scraping at scale. Together, these analyses assess data quality, timeliness, scalability, and comparative coverage, illustrating the scraper’s effectiveness for high-frequency conflict monitoring.

5.1 *Statistics*

Figure 5 illustrates the total number of articles scraped from various sources by the Sudan scraper, categorized by their source coverage: International, Regional, and National. A total of 6,946 articles have been scraped, with publication dates ranging from April 4, 2023, to October 25, 2024. National sources account for 54.56% of the total articles, emphasizing the depth of local reporting on the conflict. Regional sources account for 22.13%, offering potential insight into the conflict’s impact on surrounding regions, while international sources contribute 23.31%, providing broader, global perspectives. This distribution highlights the importance of combining national, regional, and international news outlets, to gain an understanding of the conflict and its nuances and complexities from multiple perspectives.

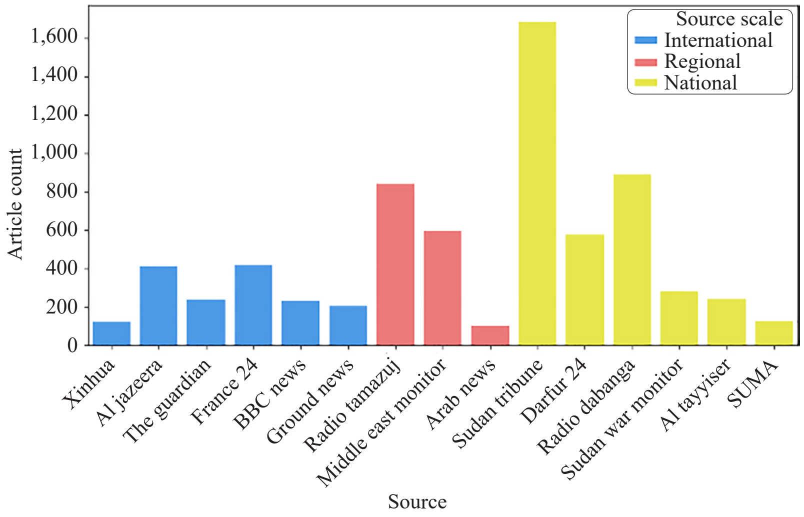


Figure 5. Total number of articles scraped, with publication dates spanning from April 4, 2023, to October 25, 2024

5.2 *Use case-spatiotemporal event detection with scraped data*

The selection of our use case, focusing on spatiotemporal event detection in Sudan between June 21 and July 5, 2024, was driven by several critical factors. The period surrounding the siege of Sinjah, the capital of the state of Sennar, is particularly important due to its role as a significant trading hub and flashpoint in the ongoing Sudan conflict. While the siege of Sinjah itself occurred on June 29-30, 2024, the analysis was expanded to include the dates preceding and following this event to uncover the buildup of military actions leading to the siege, as well as the aftermath. This timeframe was characterized by intense conflict dynamics that provide valuable insights into the operational movements of the paramilitary RSF and their territorial expansion strategies, making it an ideal window for evaluating the proposed event detection systems.

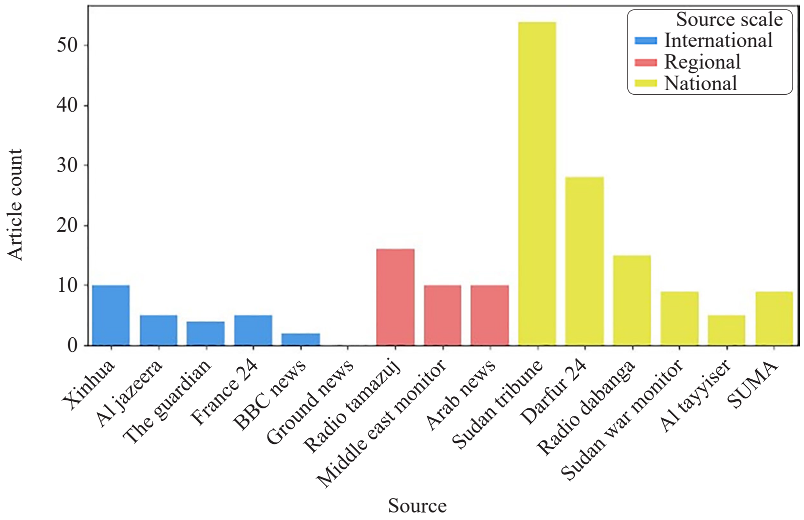


Figure 6. Total number of articles scraped from each source and stored in the MongoDB from June 21, 2024, to July 5, 2024

Figure 6 illustrates the effectiveness of the scraper in collecting a large number of articles from various sources during the selected period between June 21, 2024, and July 5, 2024. During this period, the scraper retrieved 120 articles from national sources, which comprised most of the total collection, highlighting the depth of national reporting on the conflict. It also captured 36 articles from regional and 26 from international sources, offering a broader global perspective. In total, the scraper collected 182 articles during this time interval chosen for the use case, demonstrating the scraper's efficiency in providing frequent and multi-scale coverage for conflict monitoring and its capability to quickly capture various sources, thus including local and global perspectives in the conflict analysis.

5.2.1 *Use case data preparation*

The data preparation for this use case involved three essential filtering steps to refine the dataset for further analysis.

1. Firstly, articles within the date range of June 21 to July 5, 2024, were selected from the MongoDB database, followed by a manual review to exclude articles unrelated to Sudan or focused on other countries.

2. The selected but uncategorized articles were read to assess their focus, ensuring that only articles about the conflict were retained.

3. Finally, articles that offered statistical summaries or broad overviews of events were excluded. Only those containing detailed information about specific incidents were kept for further analysis. This structured approach ensured the retention of only the most relevant, incident-specific articles, providing a more focused dataset for subsequent use.

Once the relevant incident-based articles were identified, key spatiotemporal data, including each incident's location and date, was extracted. The data was then organized sequentially based on the occurrence data, with geographic coordinates assigned to each location. This process enabled the construction of maps visualizing the sequence of events over time, providing a clear representation of the spatial and temporal dynamics of the concerned incident.

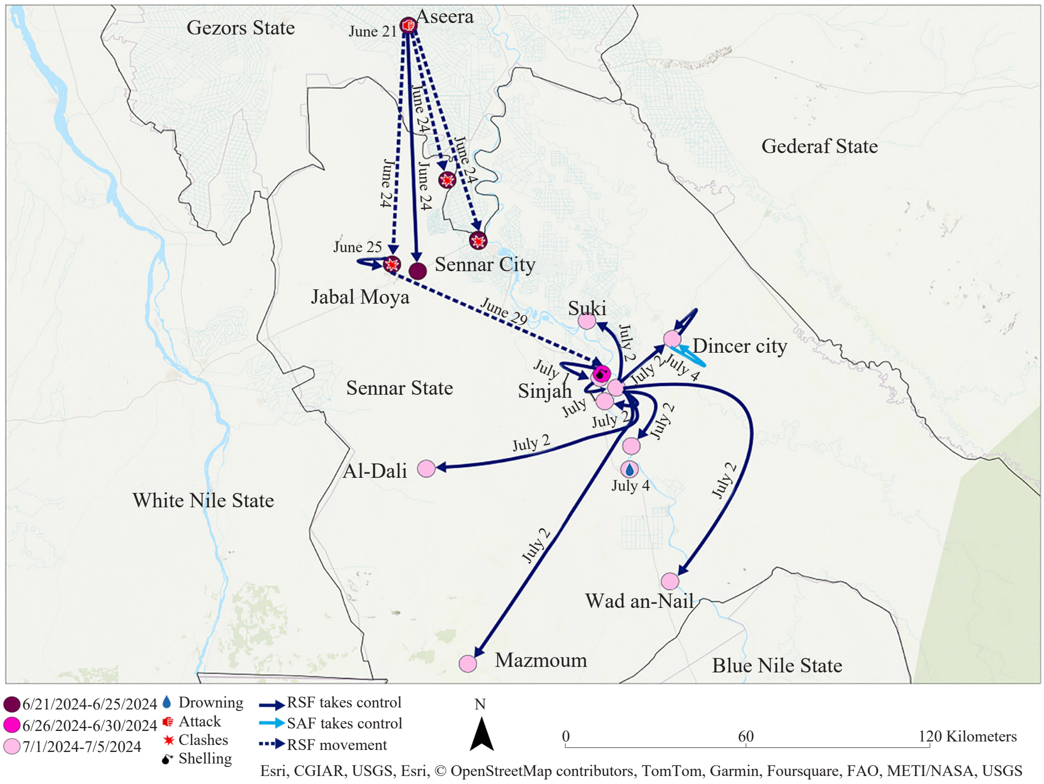
5.2.2 *Use case map*

Figure 7 illustrates the progression of conflict in Sennar from June 21 to July 5, 2024, while capturing events in neighboring states such as Gezira. The use case map highlights the strategic importance of Sennar as a key region in the ongoing Sudan conflict, while also showcasing the evolving nature of military engagements, including territorial control shifts and the broader humanitarian impacts of the conflict. By tracking these events both spatially and temporally, the map offers critical insights into the dynamics of the conflict, focusing on the operational strategies of the RSF and their confrontations with the SAF during this critical period. The map uses color-coded points to depict the timeline of incidents. Various symbols are used to indicate different natures of events, including clashes, shelling, attacks, and instances where the RSF or the SAF took control of a specific location. Dashed dark blue arrows represent RSF movements, indicating the general direction of their expansion and military activities.

Solid dark blue arrows mark locations where the RSF took control, showing key areas where they gained dominance, with little or minimal retaliation from the SAF. Additionally, solid light blue arrows indicate places where the SAF took control, reflecting successful counteractions by the Sudanese forces. These directional markers do not reflect exact troop movement or military maneuvers but provide a visual representation of the overall flow of the conflict, focusing on territorial control shifts rather than specific routes.

Several key locations are highlighted on the map beginning with Aseera in Gezera State, where the events started on June 21. Sinjah, the capital of Sennar State, is a focal point of significant conflict throughout the timeline. Other important locations include Sennar City, Jabal Moya, Dinder City, and Suki, where various incidents such as attacks, clashes, and shelling occurred.

The importance of this case study lies in its ability to showcase how our scraped data directly supports the effectiveness of fast-evolving event detection tools within a conflict zone, where rapid response and accurate information are essential. As events unfolded during and after the siege of Sinjah, understanding the locations, timings, and sequences of military engagements provided crucial context for humanitarian and strategic interventions. This approach underscores the effectiveness of database-driven filtering, manual curation, and verification in isolating incident-specific data, supporting the creation of detailed mapping of conflict. It also highlights the challenges and opportunities of extracting actionable insights from conflict data, emphasizing the need for precise geospatial and temporal analysis in conflict monitoring.



Esri, CGIAR, USGS, Esri, © OpenStreetMap contributors, TomTom, Garmin, Foursquare, FAO, METI/NASA, USGS

Figure 7. Map of the Sudan Conflict progression through Sennar from June 21, 2024, to July 5, 2024. This map was created manually using data extracted mentioned in Section 5.2.1

5.3 *Comparison with ACLED and UCDP conflict data sources*

To assess the effectiveness of our approach, this section compares the scraped data against two key datasets-ACLED and UCDP GED-using incidents preceding and following the siege of Sinjah, from June 21 to July 5, 2024, as featured in our case study. GDELT was not chosen for comparison due to its uncurated nature [32] and data quality concerns [57]. Furthermore, given that the comparison focused on the time interval of June 21 to July 5, 2024, ICEWS was excluded from this comparison due to its discontinuation on April 11, 2023. This timeframe was selected to ensure a consistent basis for comparison across datasets. We also test how excluding social media sources, such as Twitter (X), from the ACLED dataset impacts the dataset's coverage and explore the scope for improvement.

It should be noted that differences between ACLED, UCDP GED, and our scraper stem largely from how each dataset is collected and updated. ACLED combines curated open sources with human coding, partner networks, and select social media inputs, offering broad coverage but limited transparency and slower update cycles. UCDP GED relies mainly on reports from news agencies, NGOs, and intergovernmental organizations, supported by manual validation and occasional checks through local or social media when traditional coverage is scarce. In contrast, our scraper operates fully automatic, collecting articles hourly from attributable news outlets and storing both the full text and URLs. These distinctions in source type, human involvement, and update frequency account for many of the overlaps and gaps observed across the three datasets.

5.3.1 *Comparison with ACLED data*

Figure 8 reveals that the ACLED dataset captured 50 incidents, whereas the Sudan scraper captured 32 incidents, with 28 overlapped. While the ACLED dataset captured 22 incidents more than the Sudan scraper did, the Sudan scraper identified 4 incidents that ACLED did not capture.

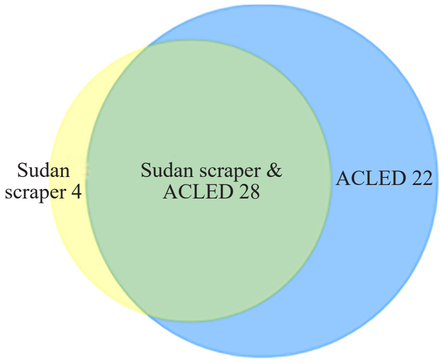


Figure 8. Venn diagram of incidents covered by ACLED from June 21, 2024, to July 5, 2024, related to events preceding and following the siege of Sinjah, the capital city of Sennar state, Sudan

However, the four articles that ACLED missed suggest that the Sudan scraper has enhanced coverage. ACLED's broader coverage can largely be attributed to its integration of social media platforms such as Twitter (X) and its reliance on strictly Arabic-language news outlets, including Al Rakoba and Sudan Akhbar, as additional data sources. Furthermore, ACLED relies on local collaborators who provide on-the-ground reports, adding another layer of granularity to its dataset. This approach allows ACLED to capture incidents that may otherwise go unreported in open-source data accessible on the Internet. These sources enable ACLED to capture incidents that may not be reported by the English-language or international outlets utilized by the scraper.

5.3.2 *Comparison with ACLED data excluding social media*

Figure 9 compares the incidents captured by the Sudan scraper and those recorded in the ACLED dataset, excluding Twitter (X), from June 21, 2024, to July 5, 2024. In this analysis, the smaller ACLED dataset recorded 40 incidents, while the Sudan scraper captured 32. ACLED reported 12 incidents that the Sudan scraper did not, whereas the Sudan scraper identified 4 incidents that ACLED did not. Both datasets share 28 incidents, indicating substantial overlap. While the Sudan scraper missed 12 incidents reported by ACLED, it is important to note that many were not completely missed due to a lack of source inclusion but because of limitations in source accessibility. For instance, Al Taghyeer is included as a source in our scraper, but the Arabic version of this site is updated more frequently than its English counterpart. Since many of the articles are not reported or translated into English, the scraper can sometimes miss incidents reported solely in Arabic.

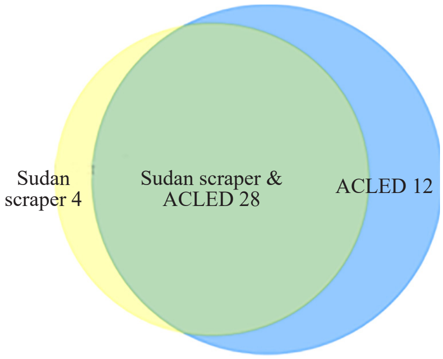


Figure 9. Venn diagram of incidents covered by ACLED excluding social media from June 21, 2024, to July 5, 2024, related to events preceding and following the siege of Sinjah, the capital city of Sennar state, Sudan

Notably, the 12 incidents missed by the Sudan scraper generally reflect follow-up developments rather than entirely new or critical incidents. For example, the scraper misses resistance clashes at Jabal Moya after June 25, 2024, and fighting reported by Al Rakoba on June 27, 2024. These incidents reflect continued activity but do not represent major shifts in the conflict. Furthermore, most missed incidents stem from fully Arabic-language sources such as Beam Reports, Al Rakoba, Sudan Akhbar, and Alnilin, which need to be included in the scraper's future scope. While these follow-up incidents are valuable for providing context, their absence does not significantly alter the broader conflict analysis, as both datasets captured the primary trends and major incidents. In conclusion, the incidents that the Sudan scraper missed highlight opportunities for improving the scraper by incorporating a broader range of national Arabic-language news sources to ensure a more comprehensive tracking system.

Additionally, it is worth noting that the scraper is capable of accessing and scraping Arabic-language news sites, whether dynamic or static. However, the primary challenge lies in accurately translating the articles. The translation process introduces complexities that can affect the overall integrity of the scraped information. To address this issue in the future, one potential solution is identifying an accurate and reliable method of translating text, such as LLMs, which have been shown to provide accurate and reliable translations [58]. This future approach aims to bridge the gap in translation and ensure that incidents reported in Arabic are fully captured, contributing to a more complete source list and database.

5.3.3 *Comparison with UCDP data*

Figure 10 shows that UCDP GED dataset recorded 8 incidents during the time interval, while the Sudan scraper captured 32. The UCDP GED reported 2 incidents not reported by the Sudan scraper, whereas the scraper identified 26 incidents that UCDP missed. Six incidents were shared, indicating minimal overlap. The Sudan scraper captures a significantly broader range of incidents than UCDP, suggesting its potential to provide more comprehensive coverage of conflict events in this region.

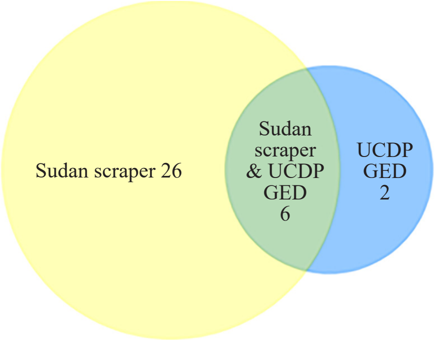


Figure 10. Venn diagram of incidents covered by UCDP from June 21, 2024, to July 5, 2024, related to events preceding and following the siege of Sinjah, the capital city of Sennar state of Sudan

5.4 *Performance testing*

At full scale, the performance of web scraping is a necessary consideration when developing a web scraper [59]. Through performance testing, we aimed to identify potential bottlenecks or issues that could arise when handling a larger dataset to ensure the scraper operates efficiently under realistic conditions. Performance testing intended to evaluate execution time was conducted on four representative news sources chosen from the 15 sources in our scraper. To ensure a balanced evaluation, two static and two dynamic sources were included, providing a simplified yet representative sample for the analysis. Dynamic sources, characterized by asynchronously loaded content, posed challenges such as waiting for server responses, particularly for images. In contrast, static sources, with non-asynchronous content, enabled faster data extraction. As shown in Figure 11, the distribution of scraping durations clearly demonstrates that dynamic sources consistently required significantly more time to process compared to static sources, reflecting the additional overhead introduced by asynchronously loaded content.

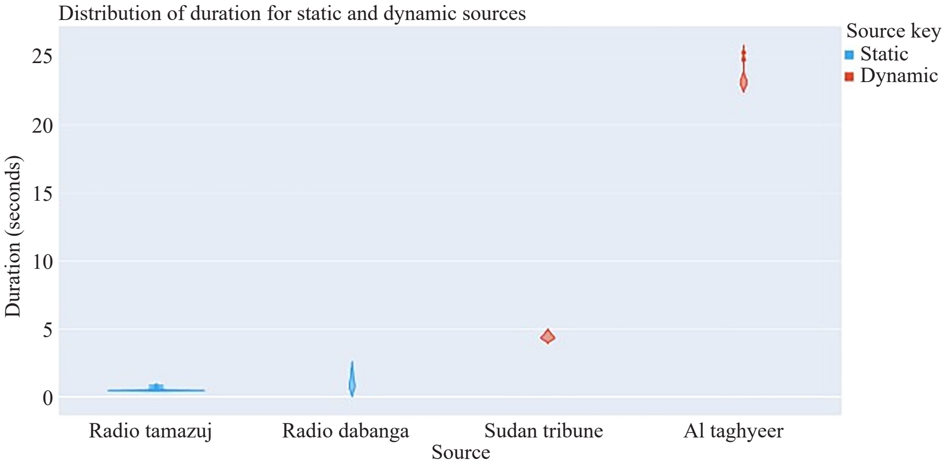


Figure 11. Violin plot documenting the difference in time taken to scrape static and dynamic web pages

Each test included up to 500 web-scraping executions with 10 trials per test, resulting in over 25,000 articles scraped. This extensive testing allowed for accurate performance analysis. From the observed results, scraping static sources took 0.802 seconds per scrape, while scraping dynamic sources extended the process to 10.597 seconds. This disparity in time highlights the additional computational cost introduced when handling dynamic content. For example, scraping articles from Radio Tamazuj, a static web news source, averaged 0.522 seconds. In contrast, scraping articles from Al Taghyeer, a dynamic web news source, required 23.554 seconds due to server-side delays in loading images.

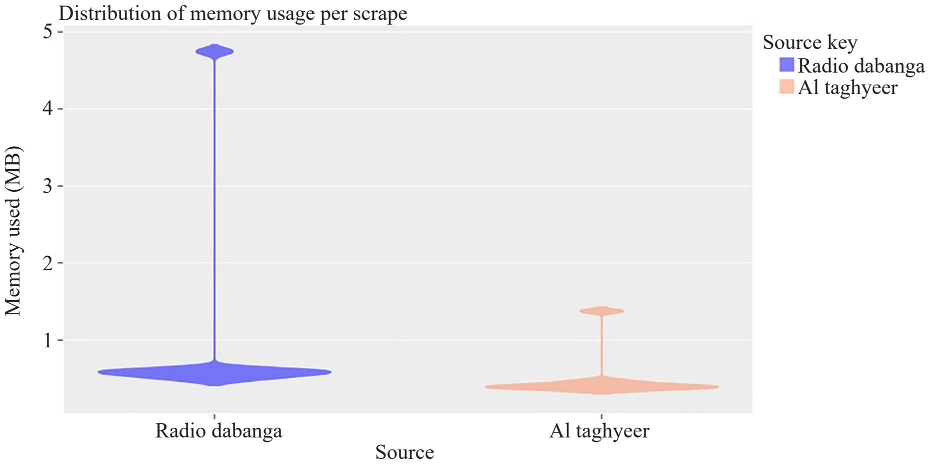


Figure 12. Violin plot documenting the difference in memory used to scrape static and dynamic web pages

In addition to measuring execution time, we also assessed memory efficiency to pinpoint any potential resource limitations when scraping at full capacity. We employed the same testing methodology as before but focused specifically on the memory consumed during each scraping operation. Radio Dabanga was chosen as the static source and Al Taghyeer as the dynamic source. Each test consisted of up to 500 scrapes across 10 independent trials for both sources, totaling around 10,000 scrapes overall. This method yielded consistent and relevant performance data.

Figure 12 shows a disparity in memory consumption between static and dynamic scraping. Extracting static content from Radio Dabanga used an average of 0.985 MB per scrape, while scraping dynamic content from Al Taghyeer used around 0.499 MB per scrape. This difference in memory usage can be explained by how memory efficiency scales with page complexity [60]. For smaller and less complex websites, lightweight parsers are likely to keep more in-memory data structures, while parsers driven by browsers optimize memory management as complexity grows. As a result, the static scraper demonstrated higher memory usage because it continuously stored text and image data, while the dynamic scraper handled resource allocation more effectively between scrapes.

6. Discussion

In this paper, we developed a web scraping toolset specifically designed to efficiently extract, organize, and store hourly conflict data from various reliable sources over the Internet, focusing on the Sudan conflict. In addition, an open-access database of online media content on the Sudan conflict was built and maintained, storing the collected data and offering a transparent and easily accessible resource for further analysis and decision support. The toolset addresses the limitations of existing conflict datasets by providing hourly, multi-scale coverage for more comprehensive and up-to-date insights into evolving conflict dynamics. The Sudan scraper successfully scraped and compiled a dataset of 6,946 articles with publication dates ranging from April 4, 2023, to October 25, 2024, from national (54.56% of total), regional (22.13%), and international sources (23.31%), reflecting detailed local reporting, the conflict's impact on neighboring countries, and broader global context. This distribution illustrates the scraper's ability to capture multi-scale coverage, ensuring an analysis of both local and global perspectives.

The scraper's practical application was evident in its coverage of the Sinjah siege, illustrating its utility in event detection. The scraper captured key information such as source URLs, content, and dates related to military engagements and humanitarian crises during the Sinjah siege, showcasing its utility in gathering data for event detection. These data were critical for tracking the sequence of events and understanding the broader implications for potential humanitarian interventions. The use case highlights the scraper's potential to support conflict observatories and decision-makers who require up-to-the-minute, location-specific information in rapidly evolving situations. Additionally, data-driven filtering methods and manual curation helped isolate incident-specific data, improving the accuracy of the conflict maps generated from the data.

A comparison between the scraper's performance and established datasets, such as ACLED and UCDP GED, further illustrated its strengths and areas for improvement. The analysis revealed that while the scraper captured many key incidents, it missed several due to the challenge of accessing Arabic-language sources. This highlights the critical role of language in event detection and suggests that improving integration with Arabic-language sources could significantly enhance the scraper's coverage. Despite this limitation, the scraper demonstrated its value by capturing local incidents that broader datasets, like UCDP GED, often overlook, underscoring its potential to complement traditional conflict datasets by offering more localized insights.

A key feature of the toolset is its use of static and dynamic scraping techniques, leveraging BeautifulSoup for static content and Selenium for dynamic content. This dual-technique approach enables consistent data collection across diverse platforms, ensuring timeliness and relevance. However, performance testing revealed that scraping dynamic sources took significantly longer (an average of 10.597 seconds per page compared with 0.802 seconds per static page), underscoring the challenge of efficiently processing dynamic content.

The scraper's hourly data collection enables reconstruction of incident sequences at short intervals, helping distinguish same-day surges from multi-day developments and identify localized extensions of activity across neighboring villages and towns. Preserving full text and URLs supports cross-checking when information changes or information overlaps across multiple sources, allowing comparisons of timing and intensity across states. Even without automated event extraction, these features improve timeline reconstruction, clarify escalation patterns, and provide a stronger foundation for future event analysis and for examining how the pace and clustering of violence evolve over short timeframes. These capabilities are not Sudan-specific; the workflow can be replicated in other areas of conflict to reconstruct short-interval timelines, compare reporting across outlets, and support incident mapping under rapidly evolving conditions.

7. Conclusion and future work

This study presented an automated, hourly web-scraping framework for conflict event monitoring in Sudan.

By integrating static and dynamic scraping techniques, keyword-based filtering, database architecture, and quality-control mechanisms, the system demonstrated the capacity to collect timely and source-attributed data at scale.

Comparative analyses with established datasets such as ACLED and UCDP GED confirmed the scraper's reliability and transparency advantages, while the Sinjah case study illustrated its operational effectiveness in capturing rapid conflict dynamics. Together, these results highlight the potential of automated data pipelines to enhance situational awareness and support conflict informatics research through reproducible, open-source methodologies.

Future development will focus on expanding language coverage, particularly Arabic-language integration, to close gaps in local reporting. Additional work will address improved entity extraction, temporal normalization, and spatial disambiguation using advanced LLM pipelines. Integration with live dashboards and machine-learning modules will enable real-time visualization and incident classification, while cross-regional extensions beyond Sudan will test scalability and adaptability across conflict contexts. Ethical and verification protocols will continue to guide data handling, ensuring transparency and accountability in automated monitoring systems. Collectively, these enhancements will strengthen the framework's applicability for researchers, humanitarian organizations, and policymakers tracking evolving conflict patterns.

Acknowledgement

Colleagues from MITRE and the Department of State provided invaluable comments and suggestions via the Sudan Conflict Observatory Project.

Funding support

The research is supported by the Department of State/MITRE Inc. (Conflict Observatory Project), NSF IUCRC and the START Programs (1841520).

Data availability statement

The data that support the findings of this study are openly available in the GitHub repository at https://github.com/stccenter/sudan_web_scraper.

Author contribution statement

Yahya Masri: Methodology, Validation, Formal analysis, Investigation, Data curation, Visualization, Writing-original draft. Anusha Srirenganathan Malarvizhi: Conceptualization, Investigation, Writing-original draft. Samir Ahmed: Methodology, Formal analysis, Software, Visualization, Writing-original draft. Tayven Stover: Methodology, Formal analysis, Software, Visualization, Writing-original draft. Zifu Wang: Data curation, Resources. Daniel Rothbart: Writing-review & editing, Project administration. Mathieu Bere: Validation, Writing-review & editing, Resources. David Wong: Writing-review & editing. Dieter Pfoser: Writing-review & editing. Chaowei Yang: Conceptualization, Writing-review & editing, Supervision, Project administration, Funding acquisition.

Conflicts of interest

The authors declare that they have no conflicts of interest to this work.

References

[1] Brosché J. Conflict over the commons: Government bias and communal conflicts in Darfur and Eastern Sudan. *Ethnopolitics*. 2023; 22: 199-221. Available from: https://doi.org/10.1080/17449057.2021.2018221.

[2] Tadesse H. Modelling conflict dynamics: Evidence from Africa-what do the data show via spatiotemporal global ACLED dataset? *Applied Spatial Analysis and Policy*. 2023; 16(4): 1541-1559. Available from: https://doi.org/10.1007/s12061-023-09522-1.

[3] Broussard G, Rubenstein LS, Robinson C, Maziak W, Gilbert SZ, DeCamp M, et al. Challenges to ethical obligations and humanitarian principles in conflict settings: A systematic review. *Journal of International Humanitarian Action*. 2019; 4(1): 1-3. Available from: https://doi.org/10.1186/s41018-019-0063-x

[4] Doocy S, Lyles E. Humanitarian needs in government-controlled areas of Syria. *PLoS Currents*. 2018; 10. Available from: https://doi.org/10.1371/currents.dis.f510b7b5f473a260a215744b4b85c38b.

[5] Höglund K, Öberg M. *Understanding Peace Research*. London: Routledge; 2011.

[6] Zwijnenburg W, Ballinger O. Leveraging emerging technologies to enable environmental monitoring and accountability in conflict zones. *International Review of the Red Cross*. 2023; 105(924): 1497-1521. Available from: https://doi.org/10.1017/S1816383123000383.

[7] Raleigh C, Linke R, Hegre H, Karlsen J. Introducing ACLED: An armed conflict location and event dataset. *Journal of Peace Research*. 2010; 47: 651-660. Available from: https://doi.org/10.1177/0022343310378914.

[8] Sundberg R, Melander E. Introducing the UCDP georeferenced event dataset. *Journal of Peace Research*. 2013; 50: 523-532.

[9] Leetaru K, Schrodt PA. GDELT: Global data on events, location, and tone, 1979-2012. In: *Proceedings of the International Studies Association Annual Convention*. San Francisco: International Studies Association; 2013. p.1-49.

[10] O'Brien SP. Crisis early warning and decision support: Contemporary approaches and thoughts on future research. *International Studies Review*. 2010; 12(1): 87-104.

[11] Eck K. In data we trust? A comparison of UCDP GED and ACLED conflict event datasets. *Cooperation and Conflict*. 2012; 47: 124-141.

[12] Polzin B. *Leveraging big data and machine learning to identify and forecast factors that influence the war in ukraine*. Master's Thesis. Monterey: Naval Postgraduate School; 2023.

[13] Sirisuriya SDS. Importance of web scraping as a data source for machine learning algorithms-review. In: *Proceedings of the 2023 IEEE 17th International Conference on Industrial and Information Systems*. Peradeniya, Sri Lanka: IEEE; 2023. Available from: https://doi.org/10.1109/ICIIS58898.2023.10253502.

[14] Rennie S, Buchbinder M, Juengst E, Brinkley-Rubinstein L, Blue C, Rosen DL. Scraping the web for public health gains: Ethical considerations from a "big data" research project on HIV and incarceration. *Public Health Ethics*. 2020; 13: 111-121. Available from: https://doi.org/10.1093/phe/phaa006.

[15] Tjaden J. Web scraping for migration, mobility, and migrant integration studies: Introduction, application, and potential use cases. *International Migration Review*. 2023; 59(3): 1367-1384. Available from: https://doi.org/10.1177/01979183231208428.

[16] Carboni A, Raleigh C. Collecting conflict data worldwide: ACLED's contribution. In: *Open Source Investigations in the Age of Google*. Singapore: World Scientific Publishing; 2024. p.171-187.

[17] Donnay K, Dunford ET, McGrath EC, Backer D, Cunningham DE. Integrating conflict event data. *Journal of Conflict Resolution*. 2019; 63: 1337-1364. Available from: https://doi.org/10.1177/0022002718790814.

[18] King G, Lowe W. An automated information extraction tool for international conflict data with performance as good as human coders: A rare events evaluation design. *International Organization*. 2003; 57: 617-642. Available from: https://doi.org/10.1017/S0020818303573064.

[19] Halkia M, Ferri S, Papazoglou M, Van Damme MS, Thomakos D. Conflict event modelling: Research experiment and event data limitations. In: *Proceedings of the Workshop on Automated Extraction of Socio-Political Events from News 2020*. Marseille, France: European Language Resources Association (ELRA); 2020. p.42-48.

[20] El Ouadi A, Beskow D. Comparison of common crawl news and GDELT. In: *Proceedings of the 2024 IEEE International Systems Conference (SysCon)*. Montreal, QC: IEEE; 2024. Available from: https://doi.org/10.1109/SysCon61195.2024.10553540.

[21] LaFree G, Dugan L. Introducing the global terrorism database. *Terrorism and Political Violence*. 2007; 19(2): 181-204. Available from:https://doi.org/10.1080/09546550701246817.

[22] Global Terrorism Trends and Analysis Center (GTTAC). *Methodology*. 2025. Available from: https://gttac.com/

methodology/ [Accessed 22th October 2025].

[23] Weidmann NB. On the accuracy of media-based conflict event data. *Journal of Conflict Resolution*. 2015; 59(6): 1129-1149.

[24] Qiu Y, Hu Z. Progress and recommendations in data ethics governance: A transnational analysis based on data ethics frameworks. *Humanities and Social Sciences Communications*. 2025; 12(1): 1354.

[25] Brown MA, Gruen A, Maldoff G, Messing S, Sanderson Z, Zimmer M. Web scraping for research: Legal, ethical, institutional, and scientific considerations. *arXiv:2410.23432*. 2024. Available from: https://doi.org/10.48550/arXiv.2410.23432.

[26] Wahed MA, Alzboon MS, Alqaraleh M, Ayman J, Al-Batah M, Bader AF. Automating web data collection: Challenges, solutions, and python-based strategies for effective web scraping. In: *2024 7th International Conference on Internet Applications, Protocols, and Services (NETAPPS)*. Kuala Lumpur, Malaysia: IEEE; 2024. Available from: https://doi.org/10.1109/NETAPPS63333.2024.10823528.

[27] Han S, Anderson CK. Web scraping for hospitality research: Overview, opportunities, and implications. *Cornell Hospitality Quarterly*. 2020; 62: 89-104. Available from: https://doi.org/10.1177/1938965520973587.

[28] Liang H, Zhu JJ. Big data, collection of (social media, harvesting). In: *The International Encyclopedia of Communication Research Methods*. Hoboken: John Wiley & Sons; 2017. p.1-18.

[29] Gheorghe M, Mihai FC, Dârdală M. Modern techniques of web scraping for data scientists. *International Journal on User-System Interaction*. 2018; 11(1): 63-75.

[30] Singrodia V, Mitra A, Paul S. A review on web scraping and its applications. In: *2019 International Conference on Computer Communication and Informatics*. Coimbatore, India: IEEE; 2019. Available from: https://doi.org/10.1109/ICCCI.2019.8821809.

[31] Zhao B. Web scraping. In: *Encyclopedia of Big Data*. Cham: Springer; 2017. p.1-3.

[32] Ünver HA. Computational international relations: What can programming, coding and internet research do for the discipline? *arXiv:1803.00105*. 2019. Available from: https://doi.org/10.48550/arXiv.1803.00105.

[33] Zhao B. Web scraping. In: Schintler LA, McNeely CL, (eds). *Encyclopedia of Big Data*. Cham: Springer; 2022. p.951-953.

[34] Rao NK, Naseeba B, Challa NP, Chakrvarthi S. Web scraping (IMDb) using Python. *Telematique*. 2022; 21(1): 235-247.

[35] Thota P, Ramez E. Web scraping of COVID-19 news stories to create datasets for sentiment and emotion analysis. In: *Proceedings of the 14th PErvasive Technologies Related to Assistive Environments Conference*. Greece: Association for Computing Machinery (ACM) and Special Interest Group on Accessible Computing (SIGACCESS); 2021. p.306-314. Available from: https://doi.org/10.1145/3453892.34613.

[36] Tee HL, Liew SY, Wong CS, Ooi BY. An estimated travel time data scraping and analysis framework for time-dependent route planning. *Data*. 2022; 7: 54. Available from: https://doi.org/10.3390/data7040054.

[37] EzZahout A, Chakouk S, Mitouilli S, Bouni MAE. A numerical real-time web tracking and scraping strategy applied to analysing COVID-19 datasets. In: *2021 7th Annual International Conference on Network and Information Systems for Computers*. Guiyang, China: IEEE; 2021. p.536-542. Available from: https://doi.org/10.1109/ICNISC54316.2021.00102.

[38] Lan H, Sha D, Malarvizhi AS, Liu Y, Li Y, Meister N, et al. COVIDScraper: An open-source toolset for automatically scraping and processing global multiscale spatiotemporal COVID-19 records. *IEEE Access*. 2021; 9: 84783-84798. Available from: https://doi.org/10.1109/ACCESS.2021.3085682.

[39] Kumar S, Roy UB. A technique of data collection: Web scraping with Python. In: *Statistical Modeling in Machine Learning*. Amsterdam: Elsevier; 2023. p.23-36.

[40] Qi Y, Shabrina Z. Sentiment analysis using Twitter data: A comparative application of lexicon and machine-learning-based approaches. *Social Network Analysis and Mining*. 2023; 13: 31. Available from: https://doi.org/10.1007/s13278-023-01030-x.

[41] Miranda CH, Sanchez-Torres G, Salcedo D. Exploring the evolution of sentiment in Spanish pandemic tweets: A data analysis based on a fine-tuned BERT architecture. *Data*. 2023; 8: 96. Available from: https://doi.org/10.3390/data8060096.

[42] Konstantinidis T, Farres AB, Xu YL, Constantinides TG, Mandic DP. Financial news classification model for NLP-based bond portfolio construction. In: *Proceedings of the 2023 24th International Conference on Digital Signal Processing*. Rhodes (Rodos), Greece: IEEE; 2023. Available from: https://doi.org/10.1109/DSP58604.2023.10167921.

[43] Zaytoon M, Bashar M, Khamis MA, Gomaa W. Amina: An Arabic multipurpose integral news articles dataset.

Neural Computing and Applications. 2024; 36: 22149-22169. Available from: https://doi.org/10.1007/s00521-024-09464-y.

[44] Abodayeh A, Hejazi R, Najjar W, Shihadeh L, Latif R. Web scraping for data analytics: A BeautifulSoup implementation. In: *Proceedings of the 2023 Sixth International Conference of Women in Data Science at Prince Sultan University*. Riyadh, Saudi Arabia: IEEE; 2023. p.65-69. Available from: https://doi.org/10.1109/WiDS-PSU57071.2023.00025.

[45] Mehta S, Pandi G. An improving approach for fast web scraping using machine learning and Selenium automation. *International Journal of Advanced Research in Computer Engineering and Technology*. 2019; 8(10): 434-438.

[46] El Asikri M, Knit S, Chaib H. Using web scraping in a knowledge environment to build ontologies using Python and Scrapy. *European Journal of Molecular & Clinical Medicine*. 2020; 7(3): 433-442.

[47] Thivaharan S, Srivatsun G, Sarathambekai S. A survey on Python libraries used for social media content scraping. In: *Proceedings of the 2020 International Conference on Smart Electronics and Communication*. Trichy, India: IEEE; 2020. p.361-366.

[48] García B, Del Alamo JM, Leotta M, Ricca F. Exploring browser automation: A comparative study of Selenium, Cypress, Puppeteer, and Playwright. In: Bertolino A, Faria JP, Lago P, Semini L. (eds.) *Quality of Information and Communications Technology*. Cham: Springer; 2024. p.142-149. Available from: https://doi.org/10.1007/978-3-031-70245-7_10.

[49] Krotov V, Johnson L, Silva L. Tutorial: Legality and ethics of web scraping. *Communications of the Association for Information Systems*. USA: Murray State University; 2020.

[50] Luscombe A, Dick K, Walby K. Algorithmic thinking in the public interest: Navigating technical, legal, and ethical hurdles to web scraping in the social sciences. *Quality & Quantity*. 2022; 56: 1023-1044. Available from: https://doi.org/10.1007/s11135-021-01164-0.

[51] Dietrich A, Rehs A. Quality assurance in web scraping of exchange rates. In: *European Conference on Quality in Official Statistics*. Vilnius: Eurostat; 2022.

[52] Parikh K, Singh D, Yadav D, Rathod M. Detection of web scraping using machine learning. *Open Access International Journal of Science and Engineering*. 2018; 3: 114-118.

[53] Jain HA, Agarwal V, Bansal C, Kumar A, Mohammed MUR, Murugesan S, et al. CoviRx: A user-friendly interface for systematic downselection of repurposed drug candidates for COVID-19. *Data*. 2022; 7(11): 164. Available from: https://doi.org/10.3390/data7110164.

[54] Wang Z, Masri Y, Malarvizhi AS, Stover T, Ahmed S, Wong D, et al. Optimizing context-based location extraction by tuning open-source LLMs with RAG. *International Journal of Digital Earth*. 2025; 18(1): 2521786. Available from: https://doi.org/10.1080/17538947.2025.2521786.

[55] Masri Y, Wang Z, Malarvizhi AS, Ahmed S, Stover T, Wong DWS, et al. Comparative analysis of BERT and GPT for classifying crisis news with Sudan conflict as an example. *Algorithms*. 2025;18(7): 420. Available from: https://doi.org/10.3390/a18070420.

[56] Sha D, Liu Y, Liu Q, Li Y, Tian Y, Beaini F, et al. A spatiotemporal data collection of viral cases for COVID-19 rapid response. *Big Earth Data*. 2021; 5: 90-111. Available from: https://doi.org/10.1080/20964471.2020.1844934.

[57] Ward MD, Beger A, Cutler J, Dickenson M, Dorff C, Radford B. Comparing GDELT and ICEWS event data. *Analysis & Politics*. 2013; 21: 267-297.

[58] Wang S, Tu Z, Tan Z, Wang W, Sun M, Liu Y. Language models are good translators. *arXiv:2106.13627*. 2021. Available from: https://doi.org/10.48550/arXiv.2106.13627.

[59] Dikilitaş Y, Çakal Ç, Okumuş AC, Yalçın HN, Yıldırım E, Ulusoy ÖF, et al. Performance analysis for web scraping tools: Case studies on BeautifulSoup, Scrapy, HtmlUnit and Jsoup. In: Márquez FPG, Jamil A, Hameed AA, Ramírez IS. (eds.) *Emerging Trends and Applications in Artificial Intelligence*. Cham: Springer; 2023. p.471-480. Available from: https://doi.org/10.1007/978-3-031-56728-5_39.

[60] Morina V, Sejdiu S. Evaluating and comparing web scraping tools and techniques for data collection. In: *1th UBT Annual International Conference on Computer Science and Engineering*. Prishtine, Kosovo: UBT Knowledge Center; 2022.