Informatyka techniczna i sztuczna inteligencja
Technical informatics and Artificial Intelligence
Engineer of XXI Century'2024

Dynamic data filtering for user queries in complex datasets

Rafał Żmuda [1], Tomasz Zając [2]*

[1] University of Bielsko-Biala, Department of Computer Science and Automatics, Engineering level student: s59080@student.ubb.edu.pl
[2] mgr inż., Department of Computer Science and Automatics, University of Bielsko-Biala, 43-309 Bielsko-Biala, Poland, tzajac@.ubb.edu.pl
* *Corresponding author, tzajac@.ubb.edu.pl*

Abstract: In the era of data-driven decision-making, vast datasets pose significant challenges in identifying and extracting relevant information efficiently. This paper introduces a dynamic approach to filtering complex datasets, focusing on extracting normalized, query-specific data. The proposed method leverages structural patterns within pre-normalized datasets, applying adaptive filtering techniques that respond to the specifics of user queries. By narrowing down data to the most relevant subsets, this approach enhances the precision of query responses without the need for additional normalization. This paper explores the challenges and solutions in handling diverse data types while ensuring the scalability and flexibility of the filtering mechanism across various domains.

Keywords: Large language models; Data extraction; Data processing; Web scrapping

Dynamiczne filtrowanie danych na potrzeby zapytań użytkowników w złożonych zbiorach danych

Rafał Żmuda [1], Tomasz Zając [2]*

[1] Uniwersytet Bielsko-Bialski wydział Butowy Maszyn I Informatyki, 43-309 Bielsko-Biała, Polska, Studia Inżynierskie, s59080@studen.ubb.edu.pl
[2] mgr inż., Wydział Budowy Maszyn I Informatyki, Uniwersytet Bielsko-Bialski, 43-309 Bielsko-Biała, Polska, tzajac@.ubb.edu.pl
* *Corresponding author, tzajac@.ubb.edu.pl*

Streszczenie: W erze podejmowania decyzji opartych na danych, obszerne zbiory danych stanowią istotne wyzwanie w zakresie efektywnego identyfikowania i wyodrębniania istotnych informacji. Niniejszy artykuł przedstawia dynamiczne podejście do filtrowania złożonych zbiorów danych, koncentrując się na ekstrakcji znormalizowanych danych specyficznych dla zapytań użytkowników. Proponowana metoda wykorzystuje strukturalne wzorce wstępnie znormalizowanych zbiorów danych, stosując adaptacyjne techniki filtrowania, które dostosowują się do specyfiki zapytań. Dzięki zawężeniu zbiorów do najbardziej istotnych danych, podejście to zwiększa precyzję odpowiedzi na zapytania bez potrzeby dodatkowej normalizacji. W artykule omówiono wyzwania oraz rozwiązania związane z obsługą zróżnicowanych typów danych, zapewniając jednocześnie skalowalność i elastyczność mechanizmu filtrowania w różnych obszarach zastosowań.

Słowa kluczowe: Modele językowe; Ekstrakcja danych; Przetwarzanie danych; Scrapowanie sieci

DOI: https://doi.org/10.53052/9788367652292.08

1. Introduction

In the era of data-driven decision-making, organizations are inundated with vast amounts of data from various sources. This influx of information presents both opportunities and challenges, particularly in extracting relevant insights efficiently. As datasets become increasingly complex and heterogeneous, the ability to pinpoint specific information becomes crucial. Traditional methods of data extraction often fall short, especially when handling large, mostly normalized datasets that exhibit varying content structures.

The importance of effectively filtering these datasets cannot be overstated. Businesses and researchers alike require mechanisms that allow for quick access to pertinent information that aligns with user queries. This need drives the exploration of dynamic data filtering techniques that adapt to the unique patterns of the underlying data while maintaining efficiency.

Current research in the field has largely focused on data normalization and performance optimization of extraction processes. However, there is a notable gap in methodologies that prioritize user-centric data retrieval in complex datasets. Key publications in this area have explored various approaches to data filtering, but many do not address the intricacies of adapting to user queries within largely normalized datasets.

This study aims to bridge this gap by presenting a dynamic approach to filtering complex datasets that emphasizes user query specificity. The primary focus is on the extraction of relevant data that meets user needs without necessitating extensive normalization. The findings from this research will contribute to a more nuanced understanding of data extraction processes, ultimately enhancing the efficacy of data utilization in various applications.

2. Materials and Methods

This section discusses the data acquisition sources and provides an overview of the fundamental steps required for processing the data [10]. We will also provide a brief overview of common data types and explore current web scraping technologies that leverage Artificial Intelligence.

2.1 Data Source

The datasets utilized in this study were sourced from the United Nations Statistics Division, which offers a publicly available repository encompassing a diverse range of subjects, including economic indicators, demographic statistics, health, and environmental data. This repository was selected due to its relatively high level of data normalization. Each dataset consists of numerous records, which exhibit varying structures and attributes [11].

2.2 Other Possible Data Sources

Given our approach to the presented problem, we are not limited to accessing information from a single data source. By leveraging the capabilities of Large Language Models (LLMs), we can efficiently process a wide variety of datasets with minimal modifications to the code. In this approach, the only component that requires adjustment is the initial step of processing the downloaded data. By applying a predefined interface to new data scrapers that access different data sources, we can ensure that all requirements for subsequent processing are consistently met.

2.3 Available Data Formats

Datasets available on the internet exhibit a wide range of formatting structures. Due to the absence of widely accepted standards for data normalization, one data provider may structure their datasets in a drastically different manner than another. Furthermore, two different providers might utilize two or more distinct data structures. The most common data structures include:

2.3.1 CSV

A CSV file is a simple text format used for tabular data representation. Each line in the file represents a single record, while fields within each record are separated by commas. Key futures are:

1. Human-Readable: Easy to read and edit with basic text editors or spreadsheet software like Microsoft Excel.
   - Limited Data Types: Primarily supports plain text and numeric values; lacks support for more complex data types.
   - No Schema: Does not enforce any schema; the first row is often used as a header for field names.

2.3.2 XML

XML is a markup language that defines rules for encoding documents in a format that is both human-readable and machine-readable. It uses a hierarchical structure of elements, each defined by tags. Key futures are:

- Custom Tags: Users can define their own tags, providing flexibility in data representation.
- Hierarchical Structure: Supports complex nested data and relationships.
- Verbose: Tends to be larger in file size compared to JSON due to its markup nature.

2.3.3 JSON

JSON is a lightweight data interchange format that uses a text format based on a subset of the JavaScript programming language. It is built on key-value pairs and can represent hierarchical data structures. Key futures are:

- Flexible Data Structures: Supports nested objects and arrays, allowing for complex data representation.
- Ease of Use: Readable and easy to manipulate in web applications and APIs.
- Language Agnostic: Widely supported across programming languages, making it a standard format for data exchange.

2.3.4 Parquet

Parquet is a columnar storage file format designed for efficient data storage and processing, especially for analytical workloads. It is part of the Apache Hadoop ecosystem. Key futures are:

- Columnar Storage: Stores data column-wise, which improves query performance for analytical operations.
- Efficient Compression: Utilizes advanced compression techniques to minimize storage space.
- Schema Evolution: Supports changes to the data schema over time without requiring full data migration.

2.4 Data Scrapping

Web scraping refers to the automated extraction of data from websites, enabling efficient collection of large-scale datasets for analysis [3]. In this study, web scraping was implemented to gather structured information from online repositories. By utilizing specialized frameworks such as Playwright, the process involved navigating web pages, retrieving the necessary data, and transforming it into a format suitable for further analysis. This approach allowed for systematic and scalable data acquisition, essential for processing diverse datasets in our methodology.

2.5 Artificial Intelligence in Web Scrapping

AI-powered web scraping introduces a new dimension to data extraction by leveraging machine learning algorithms and natural language processing (NLP) capabilities. Unlike traditional web scraping methods that rely on predefined rules and rigid scraping patterns, AI-based approaches can adapt dynamically to changing website structures and content. By understanding the context of the data through NLP models, AI can identify patterns, extract relevant information more intelligently, and even handle inconsistencies like missing data or varying formats. Furthermore, AI can assist in automating tasks such as data cleaning, classification, and transformation, streamlining the overall process and reducing the need for constant human oversight [1-2]. This allows for more efficient and scalable web scraping, capable of handling complex data sources across various domains with greater precision and minimal code adjustments. Some examples of this approach can be seen in companies like Kadoa and Browse Ai. By utilizing Large Language Models, they are able to create automated web scrappers that can be easily accessible and modifiable by users without prior coding knowledge.

2.6 Data Processing Steps

This section outlines the fundamental steps involved in data acquisition and processing. To illustrate the methodology, we have provided examples based on the query: "Average Annual Temperature." All data presented in the following sections will be contextualized with respect to this query. The dataflow diagram below (Figure 1) shows the basic frow of data in the application.

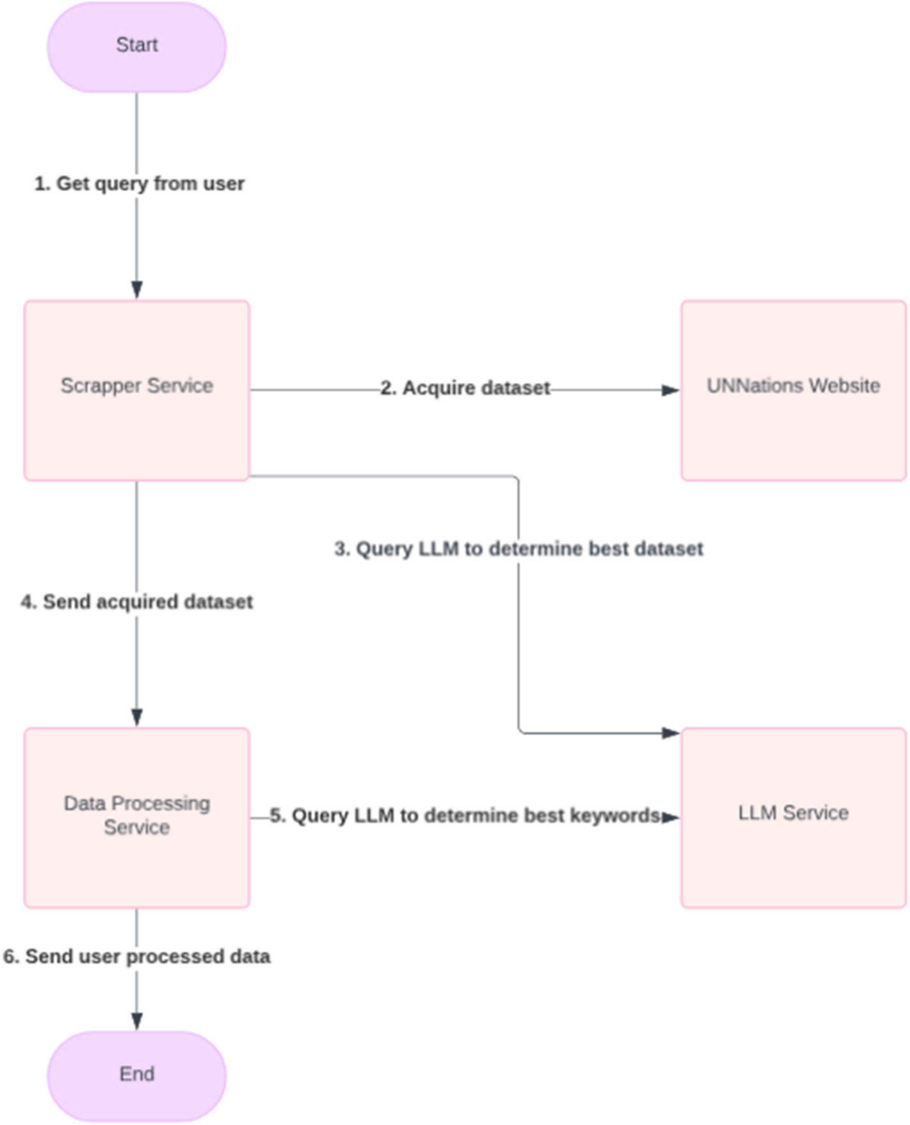
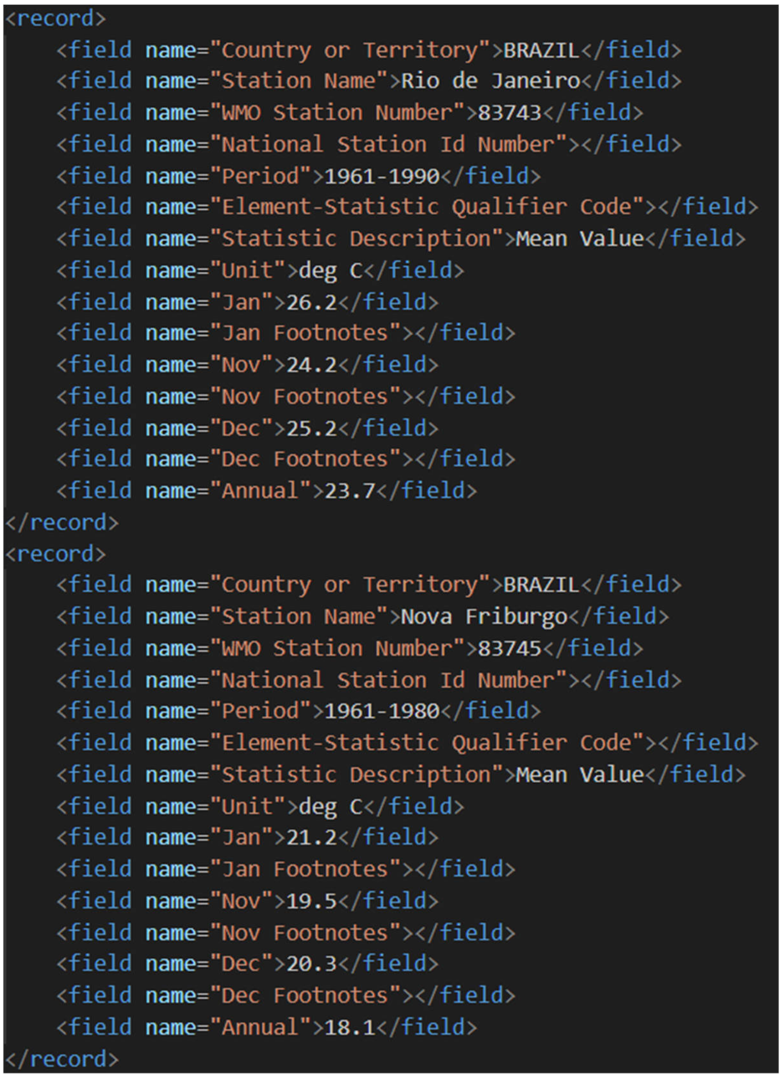


Figure 1. Shortened dataflow diagram of application

2.6.1 Scraping Correct Dataset

The first step involved identifying and scraping the relevant datasets. This process was accomplished using automated web scraping frameworks, specifically Playwright, which navigates to specified URLs and extracts the required information. The initial phase of scraping focused on querying and retrieving dataset titles. In the subsequent step, the scraped titles were input into a Large Language Model (LLM) with specific instructions to determine the dataset that best matched the provided user query [4-9]. Upon identifying the most appropriate dataset, we proceeded to scrape the data in XML format for further processing. Example of the data downloaded from the provider website (Figure 2). Some of the records were not included for the clarity.



```
<record>
    <field name="Country or Territory">BRAZIL</field>
    <field name="Station Name">Rio de Janeiro</field>
    <field name="WMO Station Number">83743</field>
    <field name="National Station Id Number"></field>
    <field name="Period">1961-1990</field>
    <field name="Element-Statistic Qualifier Code"></field>
    <field name="Statistic Description">Mean Value</field>
    <field name="Unit">deg C</field>
    <field name="Jan">26.2</field>
    <field name="Jan Footnotes"></field>
    <field name="Nov">24.2</field>
    <field name="Nov Footnotes"></field>
    <field name="Dec">25.2</field>
    <field name="Dec Footnotes"></field>
    <field name="Annual">23.7</field>
</record>
<record>
    <field name="Country or Territory">BRAZIL</field>
    <field name="Station Name">Nova Friburgo</field>
    <field name="WMO Station Number">83745</field>
    <field name="National Station Id Number"></field>
    <field name="Period">1961-1980</field>
    <field name="Element-Statistic Qualifier Code"></field>
    <field name="Statistic Description">Mean Value</field>
    <field name="Unit">deg C</field>
    <field name="Jan">21.2</field>
    <field name="Jan Footnotes"></field>
    <field name="Nov">19.5</field>
    <field name="Nov Footnotes"></field>
    <field name="Dec">20.3</field>
    <field name="Dec Footnotes"></field>
    <field name="Annual">18.1</field>
</record>
```

Figure 2. Raw data downloaded from data repository

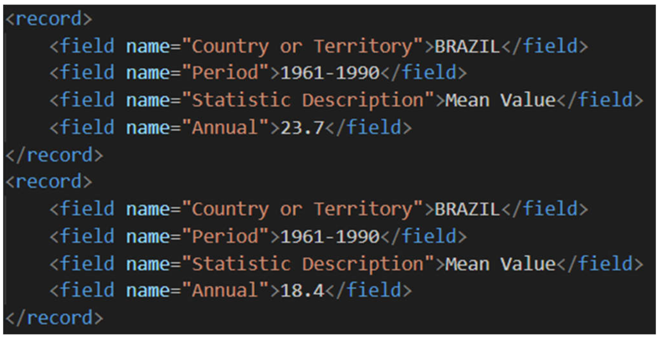
2.6.2 Initial Data Processing

Once the datasets were scraped, the initial data processing step aimed to prepare the data for analysis. This included:

- Data Transformation: During this phase, the dataset was transformed from XML format into a dictionary structure. This ensures data consistency and ease of use in future step of processing
- Removing Unnecessary Data: Our approach emphasized providing users with the most recent data available. Consequently, records were grouped by country, and only data from recent years were utilized in further processing.

2.6.3 Removing Unnecessary Records

Following the initial processing, unnecessary fields within the records were identified with the assistance of the LLM. A list of all record key values was submitted to the LLM with instructions to select the two most relevant keys corresponding to the user query. The model is instructed to choose one key corresponding to the dataset description and one key with highest probability of storing the numerical data that could answer this query. The chosen datasets consistently included fields pertaining to the country of origin and the year or period of data collection; therefore, these fields were excluded from consideration when selecting the most relevant keys. After determining the key attributes, we proceeded to remove all unnecessary fields from each record in the dataset. In example shown below (Figure 3) the most important properties determined with the help of LLM were “Statistic Description” and “Annual”

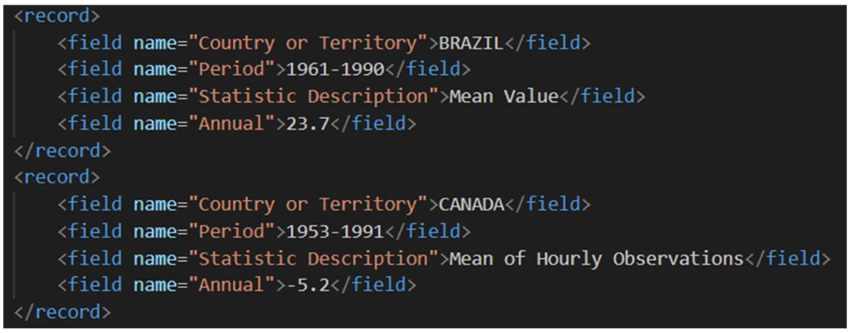


```
<record>
    <field name="Country or Territory">BRAZIL</field>
    <field name="Period">1961-1990</field>
    <field name="Statistic Description">Mean Value</field>
    <field name="Annual">23.7</field>
</record>
<record>
    <field name="Country or Territory">BRAZIL</field>
    <field name="Period">1961-1990</field>
    <field name="Statistic Description">Mean Value</field>
    <field name="Annual">18.4</field>
</record>
```

Figure 3. Data after three processing steps shown in XML format

2.6.4 Finding Best Answers to User Query

With the dataset comprising records containing only four fields: “Country,” “Year,” “Value Description,” and “Value,” the final processing step focused on identifying the best matching “Value Description” for the user’s query [7]. Given that the datasets often contain extensive information within a single file, it is common for a single “Country” key to have the same value associated with multiple records. To streamline the information presented per country key, it is essential to select the “Value Description” that best matches the user query while removing all non-essential records. Following these steps, we are left with a dictionary of records, ensuring that there is exactly one record per country containing the most accurate information relevant to the user. In example show below (Figure 4) it can be seen that the key “Statistic Description” can have different values and not all of them the user query. In this case the LLM is provided with the values of the description field and is instructed to choose the best answer for query “Average Annual Temperature”, In this case the model chooses the fields “Mean Value”. With this information we are able to delete all records with “Statistic Description” not matching our selected value and delete them.



```
<record>
    <field name="Country or Territory">BRAZIL</field>
    <field name="Period">1961-1990</field>
    <field name="Statistic Description">Mean Value</field>
    <field name="Annual">23.7</field>
</record>
<record>
    <field name="Country or Territory">CANADA</field>
    <field name="Period">1953-1991</field>
    <field name="Statistic Description">Mean of Hourly Observations</field>
    <field name="Annual">-5.2</field>
</record>
```

Figure 4. Records with all unnecessary fields removed.

3. Results

The dynamic data filtering approach introduced in this study was evaluated on a set of datasets extracted from the United Nations Statistics Division. The main objective was to assess the system’s efficiency in identifying the most relevant, normalized data that best matched user queries. The datasets, which were pre-normalized but varied in structure, provided an ideal testbed to analyse the flexibility and accuracy of the filtering method.

3.1 Dataset Filtering and Optimization

Upon extracting the raw data, the system demonstrated a high degree of accuracy in selecting relevant datasets. Out of multiple potential datasets, the automated scraping process, aided by the Large Language Model (LLM), successfully narrowed the choices to those most relevant to the query provided. The LLM was particularly effective in interpreting and matching the dataset titles to the user queries, ensuring that the correct dataset was selected for further processing. This resulted in a reduction of manual filtering tasks, enhancing operational efficiency.

3.2 Data Reduction through Key Selection

The removal of unnecessary records played a pivotal role in streamlining the datasets. By grouping records by country and selecting only the most recent data, the number of records per dataset was reduced significantly. The LLM successfully identified the most relevant fields, while discarding irrelevant or redundant information. As a result, we observed large reduction in dataset size, which directly contributed to faster query processing and less computational overhead.

3.3 Query-Specific Data Accuracy

To evaluate the precision of the final filtered dataset, several user queries were tested against the processed records. In all cases, the system effectively returned data points that accurately matched the query intent. For example, when tasked with extracting "average annual temperature" for a specific country, the LLM correctly identified and retained only the most relevant fields, such as "Mean Value." This process consistently provided high-fidelity results, with no extraneous records being presented in the final dataset.

3.4 Potential Usage of Presented Algorithm

Creating standardized datasets is essential for processing large volumes of data, but processing data solely for the sake of it offers limited value. The algorithms developed in this project are designed with a user-centric approach, focusing on making data easily accessible and interpretable for everyday users. With the vast amount of data available today, there is a wealth of information that can answer many of our daily queries. However, few people are inclined to download and process datasets containing thousands of records just to find out the average temperature in a particular country.

While certain datasets are analysed and presented on specific websites, many are unaware of the extensive, credible data sources available to them, often at no cost. This project aims to bridge that gap by providing data from reputable sources in a more accessible and visually appealing format. Through the use of infographics, complex datasets can be transformed into easily digestible, visually engaging presentations. This approach not only simplifies data interpretation but also encourages broader public engagement with data-driven insights, making valuable information more available to everyone. Image below (Figure 5) shows possible usage of processed data.

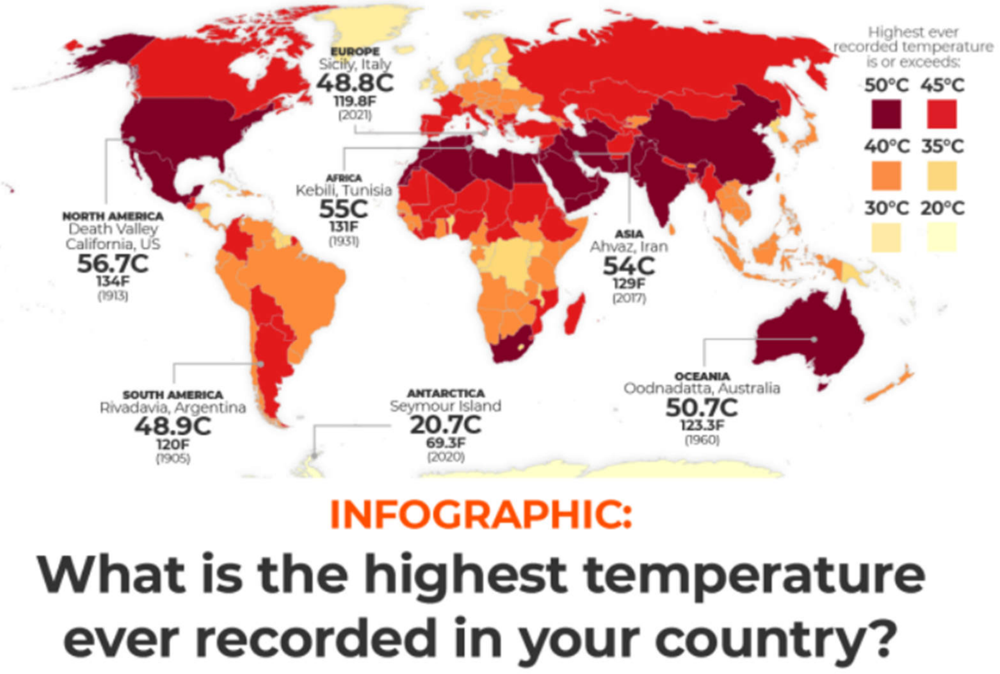


Figure 5. Example of infographic that could be created with processed data

3.5 Future Directions

While the current research presents a robust approach to dynamically filtering normalized data from complex datasets, there are several areas where further improvements and exploration could be beneficial. One potential direction is to refine the adaptability of the algorithm to handle even more diverse and unstructured datasets. This would involve enhancing the flexibility of the filtering mechanism to work across domains where data normalization is less consistent or where data comes from unconventional sources such as social media or sensor networks.

Additionally, integrating more advanced AI models for deeper data analysis and pattern recognition could further enhance the precision of the filtered results. For instance, exploring how Large Language Models (LLMs) can be used not just for filtering but for actively transforming and summarizing data could unlock new capabilities, making the datasets even more user-friendly and accessible.

Finally, developing a more intuitive user interface and visualization tools could expand the practical applications of the system, making it accessible to non-technical users who need to query and interpret complex data. This would enable broader adoption across industries and enhance the system's utility in everyday decision-making processes.

4. Conclusions

Efficient processing and standardization of big data have become critical to advancements across nearly all fields. With the growing influence of AI and large language models (LLMs) on everyday life, the demand for vast amounts of training data has surged exponentially [12]. As billions of people access the internet daily, the volume of available data continues to expand at an unprecedented rate. In response, major companies are investing in the development of innovative techniques for more efficient data collection and processing.

While the collection of data for AI training is undoubtedly one of the most crucial tasks of our time, it's equally important to ensure that the processed data does not remain solely within the confines of AI models. By leveraging modern technology, we can make data more understandable and accessible, not just for machine learning purposes, but for broader public use. This shift toward making data more open and interpretable has the potential to unlock new insights and drive progress across numerous sectors.

In this paper, we successfully developed an application capable of processing large datasets based on user requests. The system effectively generates concise, standardized datasets that contain specified data, making them easier to analyse and process further. With continued research and development, it is possible to create a fast and user-friendly interface that provides users with the precise data they require.

Reference

1. Khder, Moaiad Ahmad. Web scraping or web crawling: State of art, techniques, approaches and application. International Journal of Advances in Soft Computing & Its Applications 13.3 (2021).
2. Weerasinghe, M., M. W. P. Maduranga, M. V. T. Kawya. Enhancing Web Scraping with Artificial Intelligence: A Review. (2023).
3. Bar-Ilan, Judit. "Data collection methods on the Web for infometric purposes—A review and analysis." Scientometrics 50.1 (2001): 7-32.
4. O'Leary, Daniel E. Artificial intelligence and big data. IEEE intelligent systems 28.2 (2013): 96-99.
5. Rahmani, Amir Masoud, et al. "Artificial intelligence approaches and mechanisms for big data analytics: a systematic study." PeerJ Computer Science 7 (2021): e488.
6. Gandomi, Amir H., Fang Chen, and Laith Abualigah. Big Data Analytics Using Artificial Intelligence. Electronics 12.4 (2023): 957.
7. Chew, Robert, et al. "LLM-assisted content analysis: Using large language models to support deductive coding." arXiv preprint arXiv:2306.14924 (2023).
8. Dai, Shih-Chieh, Aiping Xiong, and Lun-Wei Ku. LLM-in-the-loop: Leveraging large language model for thematic analysis. arXiv preprint arXiv:2310.15100 (2023).
9. Qiu, Junfei, et al. A survey of machine learning for big data processing. EURASIP Journal on Advances in Signal Processing 2016 (2016): 1-16.
10. Fouad, Mohamed Mostafa, et al. Data mining and fusion techniques for WSNs as a source of the big data. Procedia Computer Science 65 (2015): 778-786.
11. Erl, Thomas, Wajid Khattak, and Paul Buhler. Big data fundamentals: concepts, drivers & techniques. Prentice Hall Press, 2016.
12. Jan, Bilal, et al. Deep learning in big data analytics: a comparative study. Computers & Electrical Engineering 75 (2019): 275-287.