Extraction and processing of web content for corpus creation: a systematic literature review.

Jair Alfredo Flores Luna[1], Miguel Hidalgo Reyes[1], Virginia Lagunes Barradas[1,2],

[1] Tecnológico Nacional de México/ITS de Xalapa,
C.P.91096 Xalapa, Veracruz, México
[2] Universidad Veracruzana,
C.P. 91020 Xalapa, Veracruz, México
227O02703@itsx.edu.mx
{miguel.hr, virginia.lb}@xalapa.tecnm.mx

Abstract. The processes and methods of text extraction and pre-processing for corpus generation are not widely documented, especially when it comes to Spanish texts. Most of the documents that collect this information are in English and focus on research carried out in the United States or in Asian countries. The aim of this systematic literature review is to know the state of the art of the technologies and methods used for the extraction of texts from web platforms and their pre-processing to generate a specific corpus. Thanks to this review, the issues defined by the research questions have been addressed and an area of opportunity has been identified for the development of new projects focused on the extraction of web information and the creation of corpora to obtain new knowledge.

Keywords: text preprocessing, web platforms, corpus, text extraction, natural language.

1 Introduction

Thanks to the internet, there are various sources of information, be they web pages or social networks, among others, where most data is currently generated minute by minute in an impressive manner. However, much of this data is stored in an unstructured way, which complicates its extraction and analysis (Vanden Broucke, S., & Baesens, B., 2018).

Currently, making decisions to improve strategies in companies or public organizations requires extracting information that is available on the internet.

The literature review conducted in this article aims to explore techniques for extracting and preprocessing web information to highlight the most commonly used techniques, programming languages, and libraries for this purpose, through the analysis of previous works or projects that have been carried out. Likewise, the difficulties they have faced, the discoveries made, and the conclusions they have reached are identified.

The findings obtained will allow us to understand the state of the art of web content extraction methods and their preprocessing, as well as to identify areas of opportunity for exploring new fields of knowledge, proposing innovative alternatives to achieve better results, or even working on new projects in specific areas that have not yet been developed.

2 Methodology

This systematic literature review follows the methodology outlined by Kitchenham (2007), which involves three phases: planning, conducting, and documenting the review. Based on this methodology, a series of steps are proposed to gather the necessary information for this review (see Figure 1).

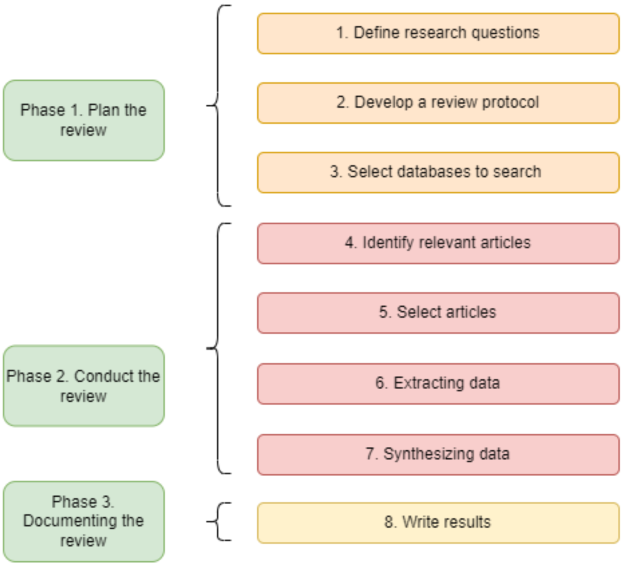

Phase 1. Plan the review
1. Define research questions
2. Develop a review protocol
3. Select databases to search
Phase 2. Conduct the review
4. Identify relevant articles
5. Select articles
6. Extracting data
7. Synthesizing data
Phase 3. Documenting the review
8. Write results

Fig. 1. Methodology of systematic review used for this review.

2.1 Planning

Initially, in the review planning stage, the topic or area of research interest was selected. Based on this selection, the following stages were developed, which narrow down and define what is desired to be known or verified.

2.1.1 Define research questions

For this systematic literature review, the research questions were first defined, which guided the information search to obtain the expected results.

The focus of this research is to understand methods for extracting content from a web platform, specifically text. Secondly, the tools, programming languages, libraries, or frameworks employed for this purpose are identified.

It was also necessary to understand the most commonly used formats for creating a text corpus, along with natural language processing techniques applicable to corpora. Lastly, the individuals or entities who will utilize the corpus and for what purpose needed to be determined.

Based on the above, the four research questions shown in Table 1 were formulated.

Table 1. Research questions.

RQ	Research questions
RQ1	What are the steps to extract text from a web page?
RQ2	What are the main languages and libraries used for text extraction, as well as the most common structured formats for the corpus?
RQ3	What are the most common natural language processing techniques applied to corpora?
RQ4	Who will use the corpus?

2.1.2 Development of the review protocol

The review protocol allows for the establishment of the search strategy and the selection of information that will be used to address the research questions, aiming to reduce information bias in the obtained results.

2.1.2.1 Definition of search strings

Once the research questions were obtained, the search strings were generated. To do this, we began by selecting the following keywords: processing, text, web, corpus, method, format, natural language.

Based on these keywords, the search strings shown in Table 2 were generated.

Table 2. Search strings.

S	Strings
S1	"web scraping" AND extraction AND (text OR content OR information) NOT document
S2	corpus AND (build OR construction) AND process AND text AND web NOT (extracting OR extraction) NOT translation
S3	corpus AND "natural language processing" AND (method OR technique) NOT speech NOT extraction NOT translation

2.1.2.1 Validation of articles

This step allowed for the establishment of the article validation method and the application of exclusion and inclusion criteria (see Table 3). Subsequently, the steps shown in Figure 2 were systematically followed.

Table 3. Search criteria.

Inclusion criteria	Exclusion criteria
Less than 5 years old	More than 5 years old
In both English and Spanish languages	The title does not include or relate to the keywords
That are research articles	The abstract does not relate to the keywords
That are specialized books on the topic	The article does not contribute to the answer of any research question

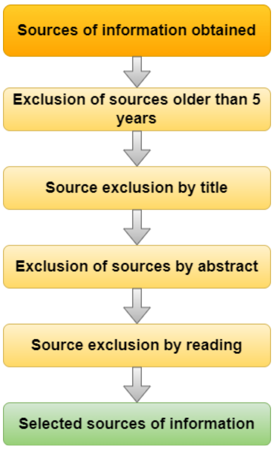

Sources of information obtained
Exclusion of sources older than 5 years
Source exclusion by title
Exclusion of sources by abstract
Source exclusion by reading
Selected sources of information

Fig. 2. Validation criteria for article filtering.

2.1.3 Selection of databases

The databases used to obtain information (see Table 4) were selected according to the questions aimed to be addressed, which are directly related to the fields of computing and data science.

Specifically, ACM was chosen for its extensive collection of works in engineering and, especially, computer science. IEEE, on the other hand, covers a wide range of publications related to all engineering areas, particularly those connected with information technologies.

Additionally, the Redalyc database was considered due to its collection of articles primarily contributed by Latin American countries, which contributes to the state of the art in the Spanish language. Likewise, ScienceDirect, with its extensive catalog of articles related to computer science, was selected for the ease of article retrieval.

Lastly, Springer was also considered given its significance as a repository for scientific articles and various types of documents that can be consulted.

Table 4. Selected databases.

No.	Databases
1	ACM
2	IEEE
3	Redalyc
4	ScienceDirect
5	Springer

2.2 Review's execution

In this phase, the article search was conducted, and a log was maintained to record the database used, the link or site's URL, the date of consultation, and the search string employed.

2.2.1 Identification of sources

In this step, the results obtained from each database were tallied (see Figure 3).

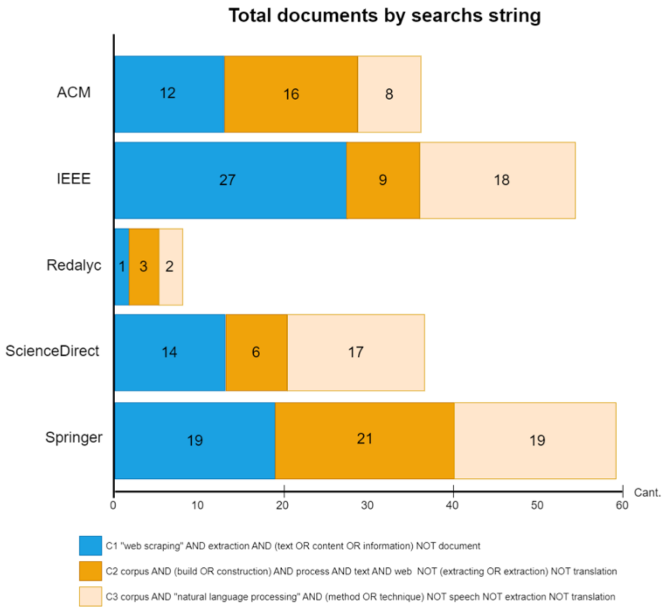


Fig. 3. Results obtained after applying the validation method and exclusion criteria.

2.2.2 Source selection

To select the articles, the validation criteria shown in Figure 2 were employed. This process involved excluding articles based on their age, title, abstract, and ultimately, their full reading. Next, the number of articles found through the search strings was observed, followed by filtering through exclusion to obtain the validated articles, as depicted in Figure 4.

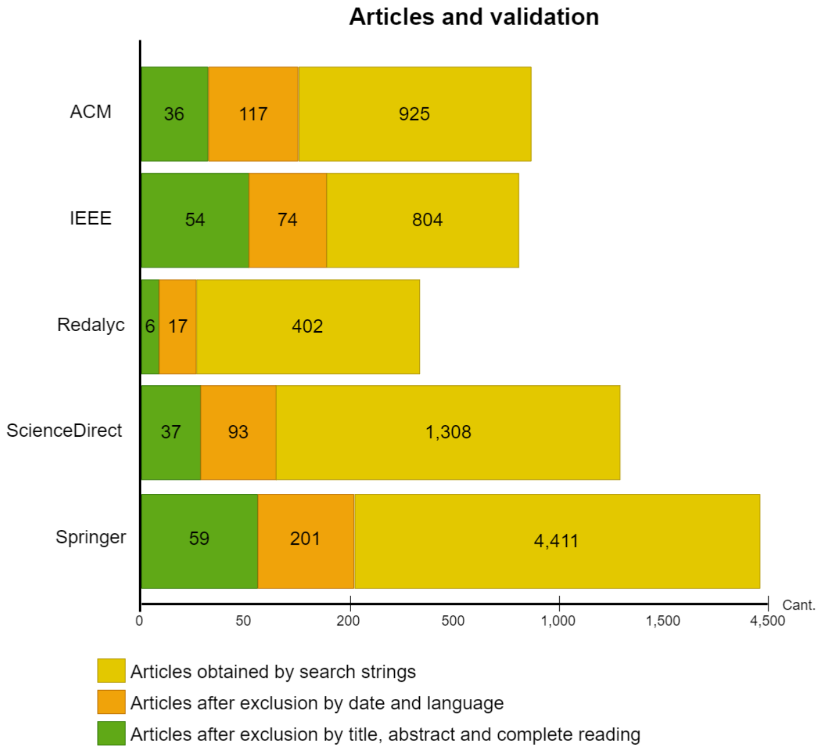
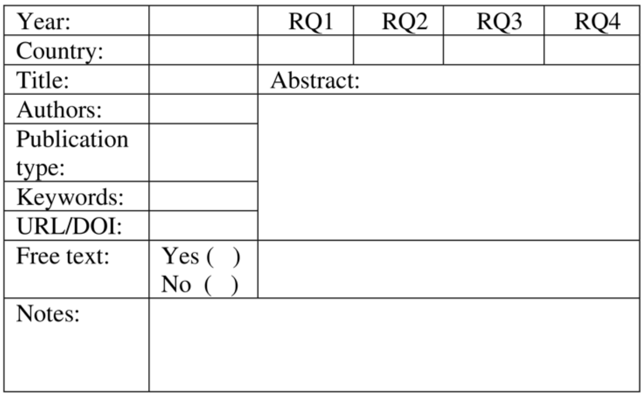


Fig. 4. Selection of articles according to the inclusion and exclusion results.

2.2.3 Data extraction

A format (Figure 5) was used where information summarizing the data obtained from the articles and other documents found is placed. The objective was to keep the information organized and facilitate the quick information reference source that one may wish to cite.



Year:		RQ1	RQ2	RQ3	RQ4
Country:					
Title:		Abstract:			
Authors:					
Publication type:					
Keywords:					
URL/DOI:					
Free text:	Yes ( ) No ( )				
Notes:					

Fig. 5. Format example for the information source summary.

3 Results

Based on the analysis of the obtained documents, answers are provided to the research questions posed at the beginning of this systematic literature review.

3.1 RQ1 What are the steps to extract text from a web page?

Evaluating the results, it was determined that any activity involving the extraction of information, regardless of the type of data, is referred to as web scraping. This process involves techniques through which various resources are used to gather useful information, enabling individuals or companies to utilize these resources and information for research purposes (Parvez et al., 2018).

Web scraping consists of two components: the crawler and the data extractor (Parvez et al., 2018). The crawler is responsible for inputting the list of URLs proposed by the user, from which information will be extracted. These URLs are called seeds, and the crawler, upon visiting each of these URLs, identifies the links present on each page and adds them to the list of URLs to be visited. The data extractor is tasked with obtaining the required information from each visited webpage.

To perform data extraction from a webpage, it is important to analyze the platform structure from which information is to be extracted. This allows the required information identification and a strategy development for its retrieval. This is necessary because web pages are encoded in markup languages like HTML or XML, and data is intermixed with tags and other webpage elements, such as CSS styles or fragments of programming code like Javascript.

Regarding methods for extracting text from web pages, three were found: 1) text manual copy and paste, 2) using APIs provided by website administrators or owners, and 3) using programming languages and libraries. The latter method, utilizing programming languages and libraries, is the most employed. This is due to its capacity to handle the extraction of substantial amounts of text and to circumvent the limitations of manual extraction, often favoring the use of APIs. It's important to note that through the systematic review, no formal methodologies that are universally applicable across all web pages were identified, as webpage designs and structures always vary from one site to another.

Finally, following the method applied by (Diouf et al., 2019), the steps for the complete extraction of a webpage are as follows:

a) Web platform crawling
b) Perform content selection.
c) List the selected page.
d) Collect links from the web platform.
e) Obtain the content.
f) Remove stop words.
g) Store the information.

3.2 RQ2 What are the main languages and libraries used for text extraction, as well as the most common structured formats for the corpus?

According to the obtained results, the most used programming languages for text extraction are Python, accounting for over half of the identified works, followed by R and Java, with PHP being used in a smaller proportion. Finally, there were cases where the use of proprietary software tools, browser extensions, or APIs for web text extraction was documented. This is shown in Figure 6.

As for the most used libraries, these are categorized by the programming language used for information extraction. In Java, "jsoup" was used; in R, "rvest" was employed for this purpose. For Python, various libraries were found to be used, primarily "requests," which enables HTTP requests to be made, and through other libraries, information was retrieved from server responses.

Among the most used are "scrapy" and "beautifulsoup," while other libraries specialized in specific web platforms or social media, such as "newspaper," "tweepy," "twint," or "ARGUS," were also utilized.

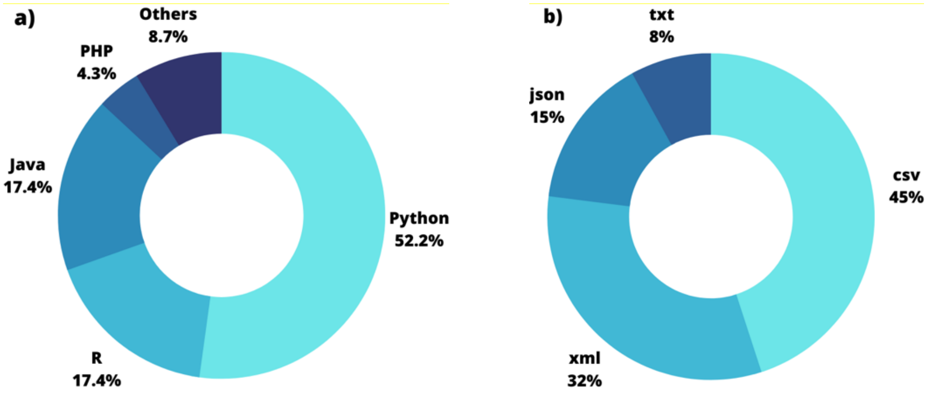


Fig. 6. a) Most used programming languages according to obtained results. **b)** Most used file formats for corpora according to obtained results.

For the most used structured formats for the corpus, it was found that the most used format is CSV, followed by XML. JSON is also used, and there are instances where a simple plain text file, TXT, can be employed. The corresponding percentages are shown in Figure 6b.

3.3 RQ3 **What are the most common natural language processing techniques applied to corpora?**

According to the obtained results analysis, natural language processing techniques applied to corpus research can be mainly categorized into three types: machine learning, deep learning, and neural network-based approaches (see Figure 7).

An example of natural language processing techniques application is reported in the work of (Moodley & Marivate, 2019), which employs the topic modeling technique to group similar documents for identifying the main themes in a set of texts extracted from news pages. This work yielded significant insights into electoral periods, and the authors suggested extending the technique to texts extracted from social media platforms.

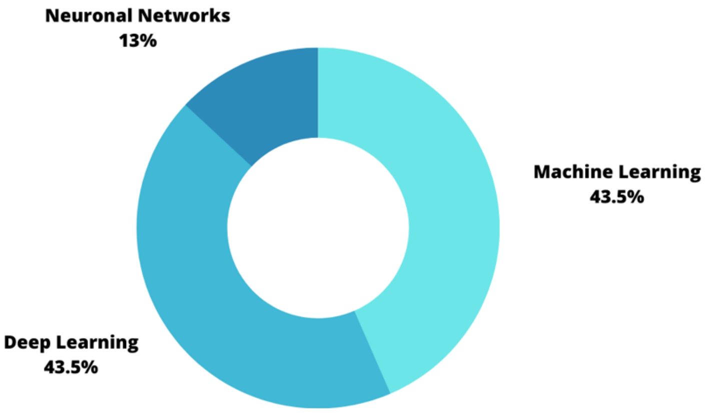


Fig. 7. Most used natural language processing techniques as tallied from the read articles.

3.4 RQ4 Who will use the corpus?

According to the selected sources reading, individuals from the academic and research field are the ones who utilize a corpus to create new projects or knowledge through various techniques (see Figure 8).

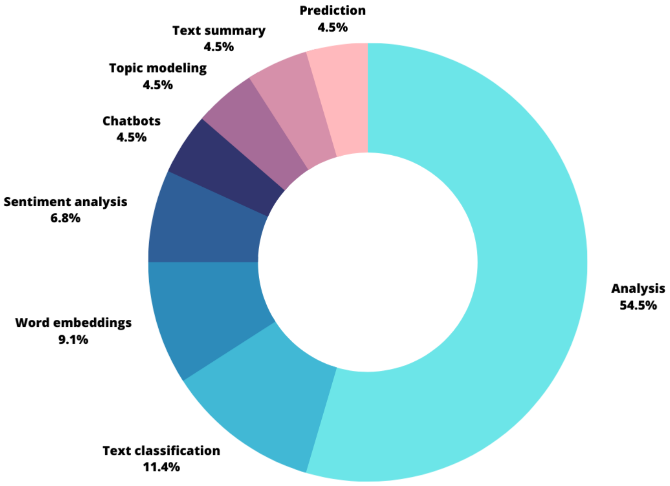


Fig. 8. Corpus main uses according to the count reported in the analyzed articles.

It can be observed that the primary use of corpora was analysis. Specifically, notable examples include the analysis of urban logistics (Tamayo et al., 2020), what patients learn about psychotropic drugs from the web (Hart et al., 2020), analysis of a corpus of fraudulent emails targeting universities (Ciambrone & Wilson, 2023), word sorting and linguistic conventions (Van Koevering et al., 2020), and context-aware recommendation systems (Yang et al., 2020). Through this, it can be discerned that corpus analysis is applied across a wide range of topics to discover or obtain new insights.

4 Conclusions

Thanks to the completion of this systematic literature review, it can be concluded that, while various projects and research on methods of extracting text from web pages have been conducted, it should be noted that only six found works follow the complete process of creating a corpus from the extracted text.

Among these found works that adhere to this process is (Moghadasi et al., 2020), detailing the design and creation of a chatbot providing advice to opiate-addicted patients. This work follows a process that addresses the inquiries guiding this systematic review, starting with web scraping for information extraction from a web platform, followed by corpus generation, and finally the application of natural language processing techniques for producing a valuable derivative product.

Another relevant article is by Boonmatham and Meesad (2020), who conducted a stock quotation analysis following the same basic steps of web scraping, corpus creation, and implementing a natural language processing technique, achieving favorable outcomes.

Upon reviewing the various selected articles, it was established that the most utilized programming language for text extraction and preprocessing is Python. This is due to the language's versatility and its extensive set of libraries designed specifically for handling and analyzing large datasets. Additionally, the library Pandas, which is used for creating and managing dataframes, was employed.

Concerning corpora, it is noteworthy that their primary usage is for language study, although they can also serve to acquire new insights or create new tools facilitating more natural interactions with computers.

Finally, an opportunity area can be identified for developing new projects related to web data extraction for corpus generation for research purposes. The majority of the found projects were developed for languages such as English, Chinese, French, among others, leaving those directed at the Spanish language significantly behind.

References

1. Alayiaboozar, E., Asghar, A. (2022). Steps for Creating Two Persian Specialized Corpora. En International Journal of Information Science and Management, 231-243. https://dorl.net/dor/20.1001.1.20088302.2022.20.4.14.3
2. Beysolow, T., II. (2018). Applied Natural Language Processing with Python. En Apress eBooks. Apress. https://doi.org/10.1007/978-1-4842-3733-5
3. Boonmatham, S., & Meesad, P. (2020). Stock Price Analysis with Natural Language Processing and Machine Learning. Proceedings of the 11th International Conference on

Advances in Information Technology (IAIT2020). Association for Computing Machinery, New York, NY, USA, Article 47, 1–6. https://doi.org/10.1145/3406601.3406652

4. Ciambrone, G., & Wilson, S. (2023). Creation and Analysis of a Corpus of Scam Emails Targeting Universities. https://doi.org/10.1145/3543873.3587303
5. Desagulier, G. (2017). Corpus Linguistics and Statistics with R. En Springer eBooks. Springer Nature. https://doi.org/10.1007/978-3-319-64572-8diouf
6. Diouf, R., Sarr, E. N., Sall, O., Birregah, B., Bousso, M., & Mbaye, S. N. (2019). Web Scraping: State-of-the-Art and Areas of Application. En HAL (Le Centre pour la Communication Scientifique Directe). Le Centre pour la Communication Scientifique Directe. https://doi.org/10.1109/bigdata47090.2019.9005594
7. Faty, L., NDiaye, M., Sarr, E. N., & Sall, O. (2020). OpinionScraper: A News Comments Extraction Tool for Opinion Mining. En 2020 Seventh International Conference on Social Networks Analysis, Management and Security (SNAMS). https://doi.org/10.1109/snams52053.2020.9336576
8. Gorro, K. D., Ali, M. F., Gorro, K. D., & Ancheta, J. M. (2020). Exploring Natural Language Processing Techniques in Social Media Analysis during a Pandemic. En International Conference on Information Technology. https://doi.org/10.1145/3446999.3447012
9. Hart, K. L., Perlis, R. H., & McCoy, T. P. (2020). What do patients learn about psychotropic medications on the web? A natural language processing study. Journal of Affective Disorders, 260, 366-371. https://doi.org/10.1016/j.jad.2019.09.043
10. Hu, M., Benson, R., Chen, A., Zhu, S., & Conway, M. (2021). Determining the prevalence of cannabis, tobacco, and vaping device mentions in online communities using natural language processing. Drug and Alcohol Dependence, 228, 109016. https://doi.org/10.1016/j.drugalcdep.2021.109016
11. Kulkarni, A., & Shivananda, A. (2019). Natural Language Processing Recipes: Unlocking Text Data with Machine Learning and Deep Learning using Python. Apress.
12. Moghadasi, M. N., Zhuang, Y., & Gellban, H. (2020). Robo: A Counselor Chatbot for Opioid Addicted Patients. 2020 2nd Symposium on Signal Processing Systems (SSPS 2020). https://doi.org/10.1145/3421515.3421525
13. Moodley, A. & Marivate, V., (2019). Topic Modelling of News Articles for Two Consecutive Elections in South Africa. 6th Intl. Conference on Soft Computing & Machine Intelligence: (ISCMI 2019) : November 19-20, 2019, Johannesburg, South Africa.
14. Parvez, M. S., Tasneem, K. S. A., Rajendra, S. S., & Bodke, K. (2018). Analysis Of Different Web Data Extraction Techniques. 2018 International Conference on Smart City and Emerging Technology (ICSCET). https://doi.org/10.1109/icscet.2018.8537333
15. Pineda-Jaramillo, J., Fazio, M., Pira, M. L., Giuffrida, N., Inturri, G., Viti, F., & Ignaccolo, M. (2023). A sentiment analysis approach to investigate tourist satisfaction towards transport systems: the case of Mount Etna. Transportation research procedia, 69, 400-407. https://doi.org/10.1016/j.trpro.2023.02.188
16. Tamayo, S., Combes, F., & Gaudron, A. (2020). Unsupervised machine learning to analyze City Logistics through Twitter. Transportation research procedia, 46, 220-228. https://doi.org/10.1016/j.trpro.2020.03.184
17. Van Koevering, K., Benson, A. R., & Kleinberg, J. (2020). Frozen Binomials on the Web: Word Ordering and Language Conventions in Online Text. https://doi.org/10.1145/3366423.3380143
18. Vanden Broucke, S., & Baesens, B. (2018). Practical Web Scraping for Data Science. En Apress eBooks. Apress. https://doi.org/10.1007/978-1-4842-3582-9
19. Yang, J., Yi, X., Cheng, D. Z., Hong, L., Li, Y., Wang, S., Taibai, X., & Chi, E. H. (2020). Mixed Negative Sampling for Learning Two-tower Neural Networks in Recommendations. En Companion Proceedings of the Web Conference 2020. https://doi.org/10.1145/3366424.3386195