ScraCOVID-19: Plataforma informativa de contenido digital mediante Scraping y almacenamiento NoSQL

ScraCOVID-19: Digital content information platform through Scraping and NoSQL storage

DOI: http://doi.org/10.17981/ingecuc.16.2.2020.18

Artículo de Investigación Científica. Fecha de Recepción: 28/07/2020. Fecha de Aceptación: 17/09/2020

Ariel Guillermo Sánchez-Paipilla
Universidad Pedagógica y Tecnológica de Colombia. Sogamoso (Colombia)
ariel.sanchez@uptc.edu.co

Mónica Katherine Durán-Vaca
Universidad Pedagógica y Tecnológica de Colombia. Sogamoso (Colombia)
monica.duran@uptc.edu.co

Angela María González-Amarillo
Universidad Nacional Abierta y a Distancia. Tunja (Colombia)
angela.gonzalez@unad.edu.co

Javier Antonio Ballesteros-Ricaurte
Universidad Pedagógica y Tecnológica de Colombia. Tunja (Colombia)
javier.ballesteros@uptc.edu.co

Para citar este artículo:
A. Sánchez Paipilla, M. Durán Vaca, A. González Amarillo & J. Ballesteros Ricaurte, "ScraCOVID-19: Plataforma informativa de contenido digital mediante Scraping y almacenamiento NoSQL", *INGE CUC*, vol. 16, no. 2, pp. 229–237, 2020. DOI: http://doi.org/10.17981/ingecuc.16.2.2020.18

Resumen

Introducción— Mantener informada a la comunidad sobre la reciente pandemia causada por el COVID-19, se ha convertido en una necesidad haciéndose indispensable el uso de canales de comunicación confiables, información precisa y basada en la evidencia.

Objetivos— Este trabajo tiene como objetivo principal crear ScraCOVID-19 una plataforma web de contenido digital dedicada a acceder a las noticias actualizadas y de manera rápida. Como caso de estudio se manejan cuatro medios digitales con licencia a nivel nacional. Las noticias se presentan de manera resumida para permitir a los lectores, en función de su interés, leer las noticias mediante algunos filtros como: desempleo, educación, maltrato, corrupción y discriminación.

Metodología— ScraCOVID-19 se crea a partir de la técnica de extracción Scraping, mediante el uso de BeautifulSoup, librería que permite extraer información en formato HTML de varios sitios web, utilizando el lenguaje de programación Python. Resultado: Se describe un modelo para realizar la categorización que extrae información útil para clasificar información en categorías haciendo referencia a las URL.

Conclusiones— A partir de técnicas de extracción utilizadas en conjunto con herramientas de almacenamiento de datos no estructurados, se obtiene información de diferentes páginas web y se administran todos los datos recogidos en una misma web generada dinámicamente.

Palabras clave— Análisis de datos; bases de datos NoSQL; comunicación digital; página web; extracción de información

Abstract

Introduction— Keeping the community informed about the recent pandemic caused by COVID-19 has become a necessity, making the use of reliable communication channels accurate and evidence-based information indispensable.

Objectives— His work has as main objective to create ScraCOVID-19 on a connected digital content web platform to access updated news quickly. As a case study, four digital media are managed with national license. The news is presented in a summarized way to allow readers, depending on their interest, to read the news through some filters such as: unemployment, education, abuse, corruption and discrimination.

Methodology— ScraCOVID-19 is created from the Scraping extraction technique, using BeautifulSoup, a library that allows information in HTML format to be extracted from various websites, using the Python programming language. Results: As a result, a categorization model is described that extracts useful information to classify information into categories by referring to the URL.

Conclusions— It is concluded that, from extraction techniques used in conjunction with unstructured data storage tools, information is obtained from different web pages and all the data collected on the same dynamically generated web is managed.

Keywords— Data analysis; NoSQL Database; digital communication; web page; information extraction

I. Introducción

Con la convergencia e integración de la tecnología de la información y la producción de contenidos, los datos han crecido rápidamente y se han convertido en los recursos estratégicos para el desarrollo de técnicas y tecnologías computacionales que manejan el uso intensivo de datos y al mismo tiempo su integración en el mundo de Big Data [1], [2]. Esta innovación tecnológica se convierte en lo más importante de la nueva era, tecnologías en constante investigación para obtener valor y mejorar la compatibilidad de los tipos de datos y fuentes. Los datos son la pieza clave de la sociedad y de la economía, por tal motivo se hace relevante las habilidades en cuanto a la captura, almacenamiento y procesamiento del dato [3]. Estos datos se encuentran dispersos y en grandes cantidades de forma estructurada o no estructurada, en diferentes tipos de formatos y contenidos web, que, al darle un tratamiento de análisis de los datos, busca organizar y dar soluciones mediante la interrogación e interpretación de dichos datos.

El contenido web que se trabajó en este artículo, está asociado al tema de la pandemia causada por COVID-19 el cual dio inicio hace no más de seis meses en Colombia, y teniendo en cuenta que la información es incesante en muchos medios, destinan la mayor parte de sus recursos a la cobertura de la emergencia sanitaria, y donde la cobertura de los medios de comunicación toma un rol importante no solo para la difusión de la información sino también para proyectar conductas sobre el aislamiento social, incidencias y recomendaciones comunicadas por las autoridades sanitarias, así como la mejora de la higiene pública, entre otros. Por tal razón, y teniendo en cuenta algunas de las asociaciones de medios del país [4], se escogieron cuatro medios digitales según tres categorías: radio, revista y periódico, los cuales por su importancia y credibilidad [5] soportan el proceso de este trabajo y permiten dar uso a técnicas que posibilitan el descubrimiento de los datos y así poder presentarlos de una manera ordenada.

Entre los mecanismos encargados de descubrir dinámicamente la información se encuentran técnicas [6], [7] que permiten recuperar contenido de una página web, dentro de los cuales se pueden mencionar los lenguajes informáticos API [8], los robots [9], los agentes inteligentes [6] y el Scraping [10], [11], siendo este último una técnica que simula la navegación humana en la web para recopilar información detallada de diferentes portales digitales. El éxito de esta técnica radica en la velocidad y en la capacidad para ser automatizado y/o programado [12].

En el escenario actual la técnica Scraping es utilizada en diferentes áreas [13]: en motores de búsqueda, sistemas colaborativos de recomendación, en el sector publicitario, en el sector de la salud y el periodismo, entre muchos otros. De cada una de estas áreas existen diversos trabajos relacionados que logran grandes resultados en el manejo de la información. Dentro de ellos se pueden citar: a) un desarrollo de Foros de discusión de salud que permiten el análisis de los datos extraídos con el objetivo de extraer la información más importante para que los profesionales en la salud se documenten y estén actualizados en diversos temas de interés [14]; b) un proyecto sobre un proceso que examina los permisos de la Comisión Costera de California utilizando varias técnicas de minería de texto, incluido extracción de información y clasificación supervisada [15]; c) una clasificación de la complejidad del texto basada en información lingüística que consiste en la construcción de un clasificador que identifica la complejidad del texto en el contexto de enseñar a leer a los estudiantes de inglés como segundo idioma (ESL por sus siglas en inglés) [16]; d) un proyecto nombrado Procircle —una plataforma de promoción que utiliza técnica de crowdsourcing y raspado de datos web, teniendo en cuenta que existen muchas promociones de marketing publicadas en varios sitios web. Se desarrolla una plataforma para recopilar noticias de promoción en un solo lugar [17]; e) un desarrollo denominado NewSone —un sistema de agregación para noticias mediante el método de scraping web, plataforma dedicada a agregar todas las últimas actualizaciones de noticias de medios nacionales e internacionales [7] y f) un proyecto FactExtract —recopilación automática y agregación de artículos y reclamos de hechos periodísticos de periódicos en línea, cuyo objetivo es mostrar en una página web, datos específicos y altamente estructurados reduciendo el esfuerzo humano [18].

Teniendo en cuenta lo anterior, se desarrolla la plataforma informativa que muestra las noticias de los cuatro medios digitales y las proporciona en un solo ambiente digital, presentando contenidos cortos, nítidos, mejorando la calidad de los resultados, con información organizada por temas, con doce filtros previamente definidos. Los usuarios finales obtienen una experiencia flexible en esta plataforma, permitiendo a los lectores leer las noticias en función del interés. Esto puede ser posible al permitirles elegir las categorías de noticias ligadas al COVID-19, dentro de ellos: desempleo, educación, ayuda humanitaria, educación virtual, maltrato infantil, maltrato de género, enfermedades psicosociales, donación de órganos, discriminación, corrupción, vacunas y pobreza, permitiendo al lector la cobertura de todas las noticias diarias y titulares de las fuentes manejadas que cuentan con licencia y confianza a nivel nacional y mundial.

El documento está estructurado de la siguiente manera: en la sección 2 se describe la metodología donde se explican las etapas que se tienen en cuenta para la extracción, almacenamiento y presentación de contenido. En la sección 3 materiales y métodos, se mencionan y explican las herramientas más importantes utilizadas en este desarrollo. La sección 4 informa los resultados y discusión. Por último, la sección 5 presenta la conclusión y trabajos futuros del trabajo realizado.

II. Metodología

El enfoque utilizado para esta investigación se tiene en cuenta a partir de lo que indica los autores [14] donde menciona que el marco de trabajo para la extracción de información es tomar como entrada las URL de los website e implica principalmente cuatro etapas: 1) Selección de las fuentes de datos, tomando algunos medios de comunicación digitales en Colombia; 2) aplicación de la técnica de Scraping; 3) almacenamiento orientado a documentos; y 4) diseño de la plataforma informativa ScraCOVID-19.

Todas las herramientas necesarias para la extracción de la información tienen la característica de ser lenguaje de marcas de hipertexto-HTML, y el sistema de base de datos manejado es NoSQL, orientado a documentos y de código abierto; el lenguaje de programación utilizado es Python. La metodología se muestra en la Fig. 1 y a continuación se explica cada una de sus etapas:

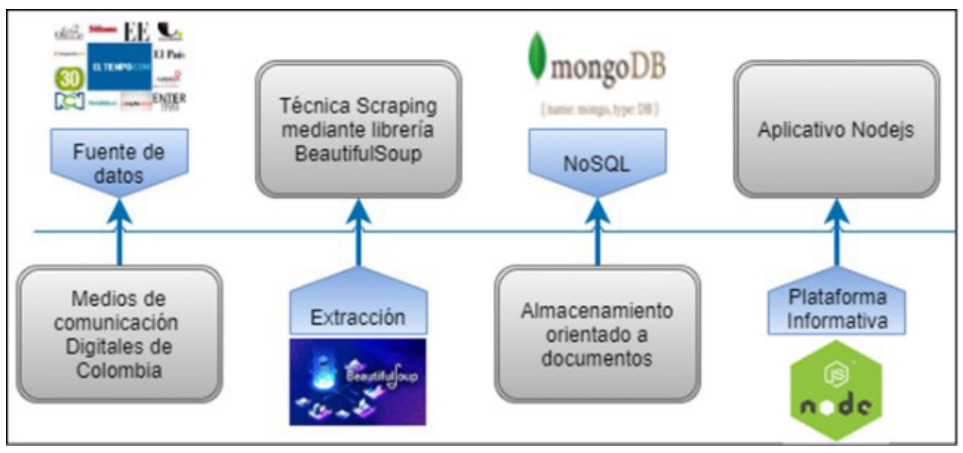

Fig. 1. Metodología.
Fuente: Autores.

A. *Fuente de datos: Medios Digitales*

Los sitios web por lo general están conformados por contenidos no estructurados a partir de etiquetas HTML que posibilitan el acceso a su información. A través de técnicas de extracción automatizada estos datos pueden ser almacenados de una forma organizada y así se pueden analizar y presentar.

Debido al auge de información que circula actualmente en redes sociales y medios de comunicación acerca del COVID-19 y dada la importancia que este tema genera para toda la ciudadanía, en el presente estudio se seleccionaron 4 de los principales medios de comunicación digitales de Colombia. Se tuvieron en cuenta 3 categorías: radio, revista y periódico, teniendo en cuenta la medición del ranking de audiencia de medios digitales que es informado a través del aplicativo Alexa [19]. En un periodo de 60 días se tomó el contenido de los portales web,

con el fin de obtener las etiquetas en HTML comunes entre ellas y de interés para la presente propuesta: referencia de la noticia, título, contenido, enlace de la noticia y posteriormente aplicar el proceso de extracción mediante la técnica de Scraping.

B. *Extracción*

En esta etapa se realizó el rastreo del contenido usando un script del lenguaje de programación Python con la librería BeautifulSoup, y otras librerías para permitir la integración con la base de datos NoSQL.

Para la extracción de los datos de los medios digitales seleccionados, se empleó la técnica de Scraping mediante el protocolo HTTP usando la librería request que se muestra en la Fig. 2. Se obtiene el código HTML por medio de métodos haciendo peticiones para obtener el código de cada una de las páginas web a fin de recolectar la información para su posterior análisis y almacenamiento.

```
31
32  def scraping_site():
33      re = requests.get(site)
34      if re.status_code == 200:
35          soup = BeautifulSoup(re.text,'html.parser')
36
37          if soup is not None:
38              client = MongoClient('localhost',27017)
39              database = client.DBNotices
40              articles = soup.find_all('article',{'class' : 'article short'})
41
42              for article in articles:
43                  #set_robot(article)
44                  robot = threading.Thread(name='set_robot',target=set_robot,args =(article,database))
45                  robot.start()
46
47  if __name__ == '__main__':
48      scraping_site()
```

Fig. 2. Petición HTTP.
Fuente: Autores.

Una vez se obtuvo el código HTML, se dio inicio al scrapeo de cada una de las páginas web mediante la librería BeautifulSoup (Fig. 3), la cual ofrece métodos find y findAll que permiten acceder desde las etiquetas HTML, a todo el contenido o porciones de éste.

```
31
32  def scraping_site():
33      re = requests.get(site)
34      if re.status_code == 200:
35          soup = BeautifulSoup(re.text,'html.parser')
36
37          if soup is not None:
38              client = MongoClient('localhost',27017)
39              database = client.DBNotices
40              articles = soup.find_all('article',{'class' : 'article short'})
41
42              for article in articles:
43                  #set_robot(article)
44                  robot = threading.Thread(name='set_robot',target=set_robot,args =(article,database))
45                  robot.start()
46
47  if __name__ == '__main__':
48      scraping_site()
```

Fig. 3. Librería Beautifulsoup.
Fuente: Autores,

C. *Almacenamiento (NoSQL): Bases de datos documentales*

Para la recolección y almacenamiento del contenido a partir de la técnica de Scraping, se utilizó MongoDB [20], una base de datos NoSQL orientada a documentos. Los registros se guardan de manera local en estructuras de datos de tipo JSON. La Fig. 4 muestra un ejemplo de la descripción del documento JSON de noticias de uno de los medios digitales. A partir de este código se pueden almacenar las etiquetas seleccionadas correspondientes a la información de la noticia: referencia, título, URL y contenido.

```
json = {'ref': ref, 'title' : title, 'href':url,'contenido' : contenido}
database.notes.insert_one(json)
```

Fig. 4. Documento JSON medio digital.
Fuente: Autores.

D. *Plataforma Informativa*

Los resultados obtenidos de la extracción y almacenamiento de los datos de cada uno de los medios de comunicación digital se muestran en una plataforma de información web (Fig. 5). Los usuarios pueden acceder en un solo ambiente digital a todas las noticias del mismo tema ofrecidas por los cuatro medios digitales, contribuyendo a mejorar la información obtenida por el lector.

Fig. 5. Plataforma Informativa.
Fuente: Autores.

III. Materiales y Métodos

Para este estudio se propone la siguiente solución como estrategia para abordar el objetivo planteado; cada paso y su resultado se explica a continuación.

Para la selección de las fuentes de datos, tomando como base el tema central que en la actualidad circula en los medios de comunicación global COVID-19, se escogieron medios de comunicación digitales de Colombia de mayor importancia, a través de la herramienta de Amazon, Ranking Alexa [18], la cual analiza el tráfico en la red de las páginas web ya sea por país, categoría, entre otros. En el presente artículo no se darán los nombres de los medios analizados, ya que el fin no es dar publicidad a ningún portal informativo sino mostrar la utilización de la técnica de extracción de datos web para posteriores análisis de información [21].

Para la selección del contenido HTML, se usaron las etiquetas con la información que contiene la noticia: id de la noticia, título, contenido y URL. Dada la estructura HTML de cada uno de los medios digitales de estudio, se tuvieron en cuenta principalmente las etiquetas article, para scrapeo de todo el contenido de la noticia y div, para el scrapeo del texto general. En la Fig. 6 se observa un ejemplo de las etiquetas utilizadas para uno de los medios digitales utilizados.

Fig. 6. Estructura HTML.
Fuente: Autores.

Los datos extraídos a partir del contenido seleccionado de cada uno de los medios digitales se almacenaron en MongoDB a través de documentos JSON [22], agregando cada registro con la referencia, el título, el contenido y URL de la noticia, utilizando la librería pymongo para hacer la conexión por medio de la línea de código import MongoClient como se aprecia en la Fig. 7.

```
31
32  def scraping_site():
33      re = requests.get(site)
34      if re.status_code == 200:
35          soup = BeautifulSoup(re.text,'html.parser')
36
37          if soup is not None:
38              client = MongoClient('localhost',27017)
39              database = client.DBNotices
40              articles = soup.find_all('article',{'class' : 'article short'})
41
42              for article in articles:
43                  #set_robot(article)
44                  robot = threading.Thread(name='set_robot',target=set_robot,args =(article,database))
45                  robot.start()
46
47  if __name__ == '__main__':
48      scraping_site()
```

Fig. 7. Conexión MongoDB.
Fuente: Autores.

Como técnica para agilizar el tiempo en los procesos de ejecución de la extracción de los datos en Python evitando exceso en los recursos de memoria, se utilizaron hilos a partir de threading código que se muestra en la Fig. 8. Con este proceso se posibilita al lenguaje de programación el lanzamiento de varias operaciones de búsqueda al mismo tiempo; permitiendo realizar las descargas de la información en paralelo, teniendo en cuenta que las páginas web son de gran tamaño y contienen gran cantidad de información; guarda la información mientras se está editando otra página y realiza el monitoreo del funcionamiento del conjunto de páginas web simultáneamente.

```
31
32  def scraping_site():
33      re = requests.get(site)
34      if re.status_code == 200:
35          soup = BeautifulSoup(re.text,'html.parser')
36
37          if soup is not None:
38              client = MongoClient('localhost',27017)
39              database = client.DBNotices
40              articles = soup.find_all('article',{'class' : 'article short'})
41
42              for article in articles:
43                  #set_robot(article)
44                  robot = threading.Thread(name='set_robot',target=set_robot,args =(article,database))
45                  robot.start()
46
47  if __name__ == '__main__':
48      scraping_site()
```

Fig. 8. Ejecución de Hilos.
Fuente: Autores.

Para la presentación de los resultados se maneja un entorno mediante la infraestructura Express del ambiente de trabajo Nodejs, que trabaja en tiempo de ejecución real. Se trabaja las conexiones bidireccionalmente lo que permite optimizar el rendimiento y la escalabilidad en aplicaciones web, motivo por el cual se muestran los resultados de una manera ágil y organizada, dirigiendo a los lectores a fuentes expertas y fiables.

IV. Resultados y Discusión

El resultado del proyecto permite comprender el procedimiento que se lleva a cabo en el análisis de datos mediante la técnica Scraping utilizando el lenguaje de programación Python.

Se crea la base de datos en Mongodb donde se almacenan todos los datos no estructurados de las fuentes seleccionadas. Los datos son analizados teniendo en cuenta las especificaciones definidas, que permiten el ensamblado, la organización de los datos con el fin de volver a realizar un análisis, mediante la implementación de modelos y algoritmos que permiten proporcionar una plataforma web como interfaz para el usuario final.

El usuario obtiene una nueva experiencia de lectura con su Interfaz de Usuario (UI) innovadora y simple, además de mostrar información responsable, veraz y de calidad. ScraCOVID-19 maneja los filtros como lo muestra la Fig. 9 en los que se tiene en cuenta como tema principal COVID-19 y coronavirus, alternándose con los demás temas asociados a esta problemática; en total se programaron doce filtros que el usuario final puede seleccionar y sobre los cuales se presentan las noticias de los cuatro medios digitales escogidos.

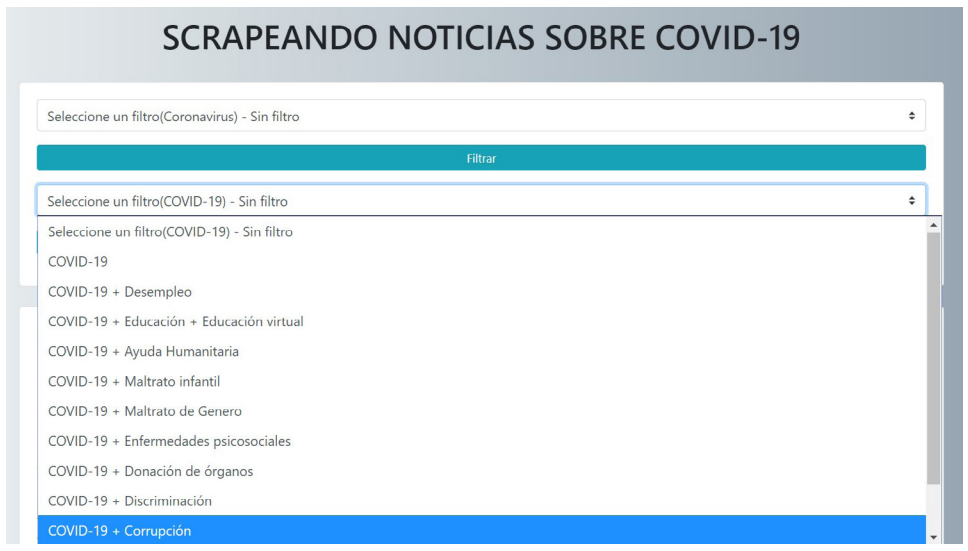

Fig. 9. ScraCOVID-19 – Filtros.
Fuente: Autores.

La Fig. 10 muestra una interfaz de usuario minimalista y fácil de usar. Presenta las noticias que se seleccionan en el momento con base en el filtro escogido, con cada uno de los temas asociados al COVID-19. El usuario puede hacer lectura del mismo tema o noticia en un tiempo dado, a partir del filtro aplicado a los diferentes medios digitales dando la posibilidad de valorar el contenido.

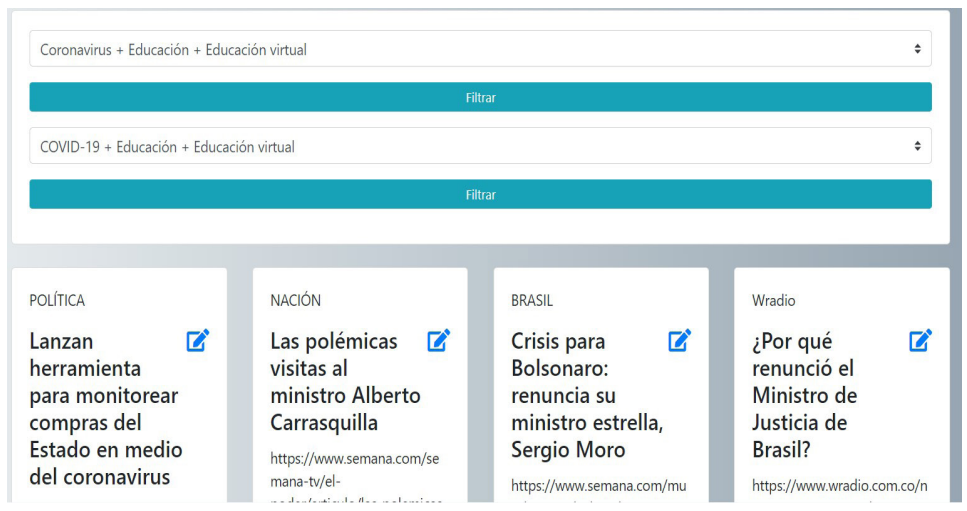

Fig. 10. ScraCOVID-19 - Noticias.
Fuente: Autores.

V. Conclusiones y Trabajos Futuros

La experimentación en tareas de extracción y almacenamiento de información que involucra técnicas de Scraping y bases de datos NoSQL, ha servido para filtrar información en diferentes páginas web y plataformas de noticias. Teniendo en cuenta que uno de los mayores desafíos a los que se está enfrentando la comunidad es a la proliferación de la información desde diferentes medios de comunicación, se crea ScraCOVID-19 como solución a la obtención de información de algunos medios digitales presentándose de manera dinámica en un solo lugar permitiéndole tener una serie de selecciones de su tema de interés.

Se puede concluir que el uso de Scraping es utilizada en diversos formatos con grandes cantidades de datos. La técnica de extracción utilizada es un campo con un avance activo que maneja varios niveles de automatización convirtiéndose en una herramienta eficaz para la obtención de información, siendo de utilidad para el análisis de contenido web. Es interesante conocer y aplicar los usos de la técnica, que para esta investigación se basó en la recopilación de un gran volumen de información de medios de comunicación digitales.

La técnica de extracción podría utilizarse para otros propósitos como la automatización de tareas, el control de estrategias en redes sociales y el análisis de opiniones mediante técnicas de minería de texto y lenguaje natural.

En el futuro, representaciones visuales deberían incluirse dentro de la plataforma de información propuesta en esta investigación, con el fin de hacer análisis sobre los datos estructurados almacenados que permitan mostrar las tendencias de los temas de noticias en el transcurso del tiempo.

Referencias

[1] A. Landers, R. N., Brusso, R. C., Cavanaugh, K. J. & A. B. Collmus, "A primer on theory-driven web scraping: Automatic extraction of big data from the Internet for use in psychological research," *Psychol. Mhetods*, vol. 21, no. 4, 475–492, 2016. https://doi.org/10.1037/met0000081

[2] R. S. Chaulagain, S. Pandey, S. R. Basnet & S. Shakya, "Cloud Based Web Scraping for Big Data Applications," presented at *2nd IEEE International Conference on Smart Cloud*, SmartCloud, NY, USA, 3-5 Nov. 2017, pp. 138–143. https://doi.org/10.1109/SmartCloud.2017.28

[3] E. Uzun, "A Novel Web Scraping Approach Using the Additional Information Obtained from Web Pages," *IEEE Access*, vol. 8, pp. 61726–61740, 2020. https://doi.org/10.1109/ACCESS.2020.2984503

[4] AMI, "AMI en los medios de comunicación," *ami.org*, 2020. https://ami.org.co/ami-en-los-medios-de-comunicacion/.

[5] ASOMEDIOS, "Medios Digitales," *asomedios.com*, 2020. http://www.asomedios.com/medios-digitales/

[6] S. C. M. de S Sirisuriya, "A Comparative Study on Web Scraping," presented at *8th International Research Conference IRS*, KDU, RML, LK, 27-28 Aug. 2015, pp. 135–140. Available from http://ir.kdu.ac.lk/bitstream/handle/345/1051/com-059.pdf?sequence=1&isAllowed=y

[7] N. R. Haddaway, "The Use of Web-scraping Software in Searching for Grey Literature," *Grey J*, vol. 11, no. 3, pp. 186–190, 2015.

[8] L. Citra, Meiliana & A. Chandra, "Social media web scraping using social media developers API and regex," *Procedia Comput Sci*, vol. 157, pp. 444–449, 2019. https://doi.org/10.1016/j.procs.2019.08.237

[9] A. Josi, L. A. Abdillah & Suryayusra, "Penerapan teknik web scraping pada mesin pencari artikel ilmiah," *SISFO*, vol. 5, pp. 1–6, 2014. Available: https://arxiv.org/abs/1410.5777

[10] D. M. Thomas & S. Mathur, "Data Analysis by Web Scraping using Python," presented at *3rd International Conference on Electronics and Communication and Aerospace Technology*, ICECA 2019, CJB, IN, 12-14 Jun. 2019, pp. 450–454. https://doi.org/10.1109/ICECA.2019.8822022

[11] D. K. Mahto & L. Singh, "A dive into Web Scraper world," presented at *3rd International Conference on Computing for Sustainable Global Development*, INDIACom 2016, New DEL, IN, 16-18 Mar. 2016, pp. 689–693.

[12] R. Mitchell, *Web Scraping with Python: Collecting More Data from the Modern Web*. Sebastopol, USA: O'Reilly Media, 2018.

[13] R. Diouf, E. N. Sarr, O. Sall, B. Birregah, M. Bousso & S. N. Mbaye, "Web Scraping: State-of-the-Art and Areas of Application," presented at *2019 IEEE International Conference on Big Data*, Big Data 2019, LA, USA, 9-12 Dec. 2019, pp. 6040–6042. https://doi.org/10.1109/BigData47090.2019.9005594

[14] U. Baskaran & K. Ramanujam, "Automated scraping of structured data records from health discussion forums using semantic analysis," *Inform Med Unlocked*, vol. 10, pp. 149–158, 2018. https://doi.org/10.1016/j.imu.2018.01.003

[15] I. Hui, "Shaping the Coast with Permits: Making the State Regulatory Permitting Process Transparent with Text Mining," *Coast Manag*, vol. 45, no. 3, pp. 179–198, 2017. https://doi.org/10.1080/08920753.2017.1303694

[16] M. Z. Kurdi, "Text Complexity Classification Based on Linguistic Information: Application to Intelligent Tutoring of ESL," *JDMDH*, pp. 1–38, 2020. Available: https://arxiv.org/abs/2001.01863

[17] L. Junjoewong, S. Sangnapachai & T. Sunetnanta, "ProCircle: A promotion platform using crowdsourcing and web data scraping technique," presented at *7th ICT International Student Project Conference*, ICT-ISPC 2018, Nakhon, TH, 11-13 Jul. 2018, pp. 1–5. https://doi.org/10.1109/ICT-ISPC.2018.8524003

[18] E. N. Sarr, O. Sall & A. Diallo, "FactExtract: Automatic C ollection and A ggregation of A rticles and J ournalistic F actual Claims from Online Newspaper," *5 International Conference on Social Networks Analysis, Management and Security*, SNAMS, VA, ES, 15-18 Oct. 2018, pp. 336–341. https://doi.org/10.1109/SNAMS.2018.8554421

[19] Alexa, "Top Sites in Colombia," *Amazom Company*, 2020. https://www.alexa.com/topsites/countries/CO

[20] J. Díez, "Aplicación para monitorización de precios para Android," *Trabajo grado*, ETS, UPNA, PNA, ES, 2019. Disponible en https://hdl.handle.net/2454/33693

[21] C. Lopezosa, L. Codina & C. Gonzalo-Penela, "Off-page SEO and link building: General strategies and authority transfer in the digital news media," *Prof Inf*, vol. 28, no. 1, pp. 1–14, 2019. https://doi.org/10.3145/epi.2019.ene.07

[22] R. Bahana, R. Adinugroho, F. L. Gaol, A. Trisetyarso, B. S. Abbas & W. Suparta, "Web crawler and back-end for news aggregator system (Noox project)," presented at *2017 IEEE International Conference on Cybernetics and Computational Intelligence*, Cybernetics, HKT, TH, 20-22 Nov. 2018, pp. 56–61. https://doi.org/10.1109/CYBERNETICSCOM.2017.8311684

Ariel Guillermo Sánchez-Paipilla es estudiante de Ingeniería de sistemas y computación en la Universidad Pedagógica y Tecnológica de Colombia. Su trabajo se centra analítica de datos, ciencias de la computación y telecomunicaciones en el grupo de investigación INFELCOM-UPTC. Su pasatiempo favorito es la investigación y el desarrollo de software. https://orcid.org/0000-0001-7181-1466

Mónica Katherine Durán-Vaca es Ingeniera de Sistemas de la Fundación Universitaria Juan de Castellanos (Colombia). Especialización en Bases de Datos de la Universidad Pedagógica y Tecnológica de Colombia. Magister en Ingeniería de la Información de la Universidad de los Andes (Colombia). Actualmente docente de la Universidad Pedagógica y Tecnológica de Colombia en el área de Bases de Datos. Grupo de investigación INFELCOM -UPTC. Interesada en temas de analítica y ciencia de datos. https://orcid.org/0000-0002-4806-683X

Angela María González-Amarillo es Ingeniero de Sistemas 2006 Fundación Universitaria Juan de Castellanos (Colombia). Especialista en educación Superior a Distancia de la Universidad Nacional Abierta y a Distancia (Colombia), Master of Business Administration de la Universidad Nacional Abierta y a Distancia (Florida, USA). Docente del programa de ingeniería de sistemas de la Universidad Nacional Abierta y a Distancia (Tunja, Colombia). Líder zonal de la Escuela de Ciencias Básicas Tecnología e Ingeniería de la Zona Centro Boyacá de la Universidad Nacional Abierta y a Distancia, Grupo de Investigación GIDESTEC. https://orcid.org/0000-0002-3568-7530

Javier Antonio Ballesteros-Ricaurte es Ingeniero de Sistemas de la Universidad de Boyacá (Tunja, Colombia). Magíster en Ciencias Computacionales (convenio de la Universidad Autónoma de Bucaramanga en Colombia y el Instituto Tecnológico de Estudios Superiores de Monterrey en México). Docente de la Escuela de Ingeniería de Sistemas y Computación e investigador del Grupo de Investigación de Manejo de Información (UPTC). https://orcid.org/0000-0001-9164-4597