Extractor de noticias para el análisis integral del impacto de la pandemia en la provincia del Chubut

Emanuel Balcazar and Leo Ordinez

Laboratorio de Investigación en Informática (LINVI), FI - UNPSJB
Bvd. Brown 3051, Puerto Madryn, Argentina
{emanuelbalcazar13,leo.ordinez}@gmail.com

Resumen En el marco de la pandemia por COVID-19, con el objetivo de analizar la situación social que se estaba atravesando, se propone la extracción y análisis de información periodística, de forma automática, para su tratamiento y explotación. Para ello, se desarrolló una herramienta que recupera notas periodísticas de los principales medios de la Provincia del Chubut, generando un dataset de alta riqueza en términos de su potencial impacto.

Keywords: extracción automática de información, medios digitales, NLP

1. Introducción

En el marco de la COVID-19, la vida de las personas y el curso de las actividades y servicios no esenciales sufrieron cambios drásticos a partir del Decreto DECNU-2020-297-APN-PTE y sus normativas anexas de nivel provincial y/o municipal en Chubut. Las medidas de Aislamiento Social, Preventivo y Obligatorio (ASPO) y Distanciamiento Social, Preventivo y Obligatorio (DISPO) generaron desencadenantes de impacto social y económico que necesitan ser identificados y monitoreados para proveer información a los formuladores de políticas públicas.

En este trabajo se propone la construcción de conocimiento a partir de diferentes estrategias y herramientas de relevamiento de información y datos, que favorezcan un análisis, procesamiento y ponderación de la situación social dada en una circunstancia sanitaria como la generada por el COVID-19. En particular se considerará la región delimitada por los límites geográficos de la provincia del Chubut, ajustando la escala territorial a nivel de ciudades, pueblos y comunas rurales. La construcción de conocimiento, se hará a partir de la extracción, procesamiento y análisis automático de información periodística publicada en la prensa provincial, la cual luego será presentada de manera acorde, a fin de poder evaluarse de manera indirecta la evolución de diferentes temáticas, que impactan en la sociedad.

Los resultados serán obtenidos mediante la aplicación de técnicas de Procesamiento de Lenguaje Natural (NLP) y extracción de datos de sitios web (web

scraping), para luego ser presentados en una aplicación web que permitirá su análisis y/o difusión, así como su explotación más profunda. En el transcurso de la pandemia se han elaborado propuestas similares, con distintos objetivos [2,3,4,5,10,12]. En todos los casos, se busca la construcción de conocimiento nuevo a partir de la minería de información.

2. Materiales

En base a experiencias anteriores de trabajos similares [1,8], muchas de las estrategias utilizadas sirvieron de insumo para este trabajo.

Como primera aproximación a la solución presentada, se realizaron una serie de experimentos utilizando el buscador de Google con la intención de obtener artículos periodísticos que, en un principio, tocaran el tópico del COVID-19 en su contenido. Seguido a lo anterior, se procedió a seleccionar cuales serian las fuentes de datos (sitios webs), que resultaban más acordes para el tipo de artículos que se esperaba. Asimismo, esta decisión se vio influenciada por la intención de solo obtener los artículos que cubran la provincia de Chubut. Luego, se hicieron una serie de extracciones de prueba analizando el contenido de los sitios web periodísticos para conocer la estructura que poseían y cómo esto afectaba a la forma de extracción del contenido. Esto terminó de definir la decisión respecto a utilizar el buscador de Google como *hub* para las búsquedas y aprovechar su parametrización.

A fin de realizar búsquedas por sitios específicos y fechas particulares, se optó por incluir en la ecuación de búsqueda una fecha en el formato utilizado por el sitio. De esta manera, cada búsqueda, dependiendo de en qué sitio se está realizando, tiene en su contenido una fecha y así Google al ver que la fecha se encuentra dentro del artículo, lo devuelva como un resultado de búsqueda. Este experimento dio buenos resultados por lo que fue la alternativa elegida.

Una vez analizada la factibilidad de utilizar el buscador de Google como *hub* para la extracción de artículos, se pasó a la versión prográmatica de dicho buscador. Esta aplicación, denominada *Google Custom Search Engine (CSE)*, ofrece una API para la realización de búsquedas, mediante ecuaciones. A la vez, posee limitaciones de acuerdo a la versión de uso gratuito y la paga, las cuales determinaron ciertas decisiones arquitectónicas.

2.1. Selección de las fuentes

En este paso, se realizó un análisis de todos los sitios web de noticias disponibles de los cuales se buscó donde extraer la información. Dicho análisis fue limitado a solo los sitios que brindaran noticias informativas de cualquier tema dentro de la provincia de Chubut dado a que esta restricción forma parte de los requerimientos y limitantes del proyecto.

Para la selección de las fuentes se tuvo en cuenta un criterio de territorialidad y uno de volumen. El primero tiene que ver con las características geográficas de la provincia del Chubut, la cual tiene una gran extensión territorial y una

alta concentración en tres zonas principalmente. La primera de ellas alrededor de las ciudades de Rada Tilly, Comodoro Rivadavia y Sarmiento, al sur de la provincia. Una segunda, al noreste, conformada por la región del Valle Inferior del Río Chubut (VIRCH) y Península Valdés, donde se destacan las ciudades de Puerto Madryn, Trelew, Rawson, Gaiman y localidades menores. Finalmente una tercera zona, ubicada en el oeste cordillerano, en la que se destacan las ciudades de Esquel y Trevelin. En cuanto a volumen, se buscó que los medios tuvieran un caudal considerable de artículos, a fin de que el análisis sea lo mas amplio y detallado posible. Los medios seleccionados fueron:

- `https://diariocronica.com.ar` (sur)
- `https://www.eldiarioweb.com` (noreste)
- `https://www.diariojornada.com.ar` (noreste)
- `https://www.elpatagonico.com` (sur)
- `https://www.elchubut.com.ar` (noreste)
- `https://radio3cadenapatagonia.com.ar` (noreste)
- `https://diariolaportada.com.ar` (oeste)
- `https://www.red43.com.ar` (oeste)

Una vez seleccionado los sitios webs, se procedió a analizar cómo se estructuraban las noticias, puntualmente el análisis se realizó investigando el HTML resultante con el fin de identificar por cada sitio el formato utilizado y las etiquetas disponibles para poder extraer la información requerida, determinando los selectores CSS involucrados.

3. Métodos

Como se mencionó antes, para el diseño e implementación de la solución, se tuvieron en cuenta experiencias previas de proyectos similares. En términos metodológicos, el presente se basó en conceptos de enfoques mayores [11,9,6,7], pero adaptados a la escala de este trabajo y, sobre todo, a su urgencia. Las herramientas utilizadas para este desarrollo fueron: *NodeJS 14.16.0*, *NPM 6.14.4*, *Python 3.9.4*, *AdonisJS 4.0.13*, *VueJS 4.3.1*, *RabbitMQ 3.8.2*, control de versiones: *Github*, editor de código: *Visual Studio Code*, *PostgresQL 12.4*, *DBeaver 6.2.0*, *Docker 20.10.5* y sistemas operativos: *Ubuntu desktop 20.4* 64-bits y *Windows 11*

3.1. Arquitectura

La arquitectura se diseñó basada en microservicios con el fin de poder separar en módulos más pequeños cada parte del procesamiento. En la Figura 1 se muestra el diseño general de la misma.

El componente **cliente** es la parte visual del sistema. Se encarga de mostrar al usuario una interfaz web con el fin de brindarle la información recopilada de manera sencilla mediante gráficos y diagramas. Además permite el acceso a los artículos extraídos y normalizados.

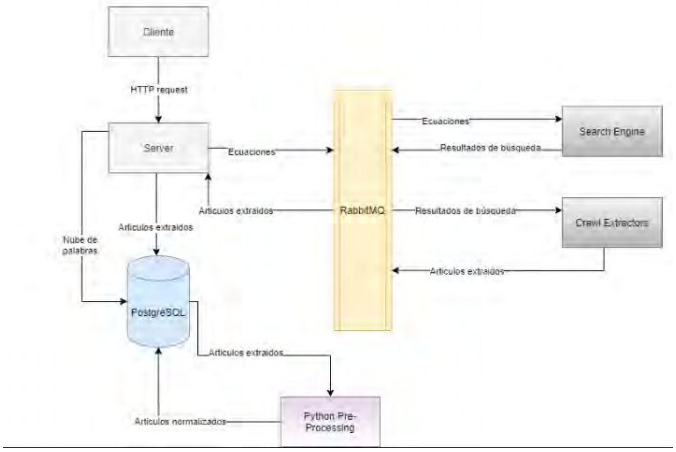


Figura 1: Arquitectura de la aplicación basada en microservicios

El componente **servidor** es el intermediario entre el componente cliente, la base de datos y la cola de mensajes. Se encarga de recibir las peticiones del cliente y obtener los datos solicitados desde la base de datos. También se conecta a la cola de mensajería para recibir los artículos que fueron extraídos y así persistirlos en la base de datos. Asimismo, el servidor tiene dos *planificadores* funcionando en simultáneo: uno se encarga de ejecutar las extracciones cada cierto período configurable; y el otro administra la construcción de nubes de palabras, a partir de los artículos procesados. El primer planificador se incorporó a la arquitectura por las restricciones de pago y correspondiente uso del servicio Google CSE. Se optó por utilizar una **base de datos relacional** para la persistencia de los datos requeridos.

La **cola de mensajería** es el componente central en el sistema, debido a que gracias a ella se comunican todos los demás componentes. La decisión de haber utilizado una cola de mensajería se justifica en la posibilidad de implementar un esquema de productor-consumidor. Esto permite, además la paralelización de los procesamientos, dado a que a través de la cola de mensajería transitan los artículos, logs del sistema, ecuaciones de búsqueda, instrucciones a otros componentes, entre otros.

El componente **motor de búsqueda** se encarga de recibir a través de la cola de mensajería una ecuación de búsqueda para luego hacer la llamada a la API haciendo uso de Google CSE. Los resultados de búsquedas se vuelven a colocar en la cola de mensajería para ser tomados por los extractores.

El componente **extractores** recibe a través de la cola de mensajería los resultados de las búsquedas y los procesa extrayendo de cada resultado el artículo en formato HTML. Luego se le aplican los selectores correspondientes al sitio y por último se envía a la cola de mensajería el articulo extraído, con datos adicionales como la ecuación utilizada, selectores, links originales y demás *metadatos*.

Por último, el componente desarrollado en **Python** se encarga de pre-procesar los artículos para permitir el posterior armado de las nubes de palabras. En este componente se normalizan los artículos extraídos, aplicando una serie de funciones de *stemming*, eliminación de caracteres inválidos, eliminación de preposiciones y demás, con el fin de obtener una versión del artículo que resume en términos neutrales el contenido del mismo. De esta manera, se puede utilizar para el armado de las nubes de palabras ya que sintetizan en conceptos y frecuencias las temáticas tratadas.

Para realizar las búsquedas, se optó por crear un componente independiente que obtenga las ecuaciones de búsqueda entrantes, ejecute la búsqueda y devuelva los resultados de la misma.El componente se divide en una serie de sub-módulos que se encargan de realizar el flujo de datos desde que llega una ecuación de búsqueda hasta que se devuelven los resultados. Para esto, se cuenta con *workers* que reciben una ecuación cada uno, donde cada worker recibirá la ecuación asociada al tópico al cual está suscrito.

Al iniciar el componente, se levantan tantos workers como sitios web configurados haya, cada worker se encargará de recibir de RabbitMQ las ecuaciones de búsqueda específicos del sitio que le corresponda, cada worker trabaja con un sitio web en particular, esto se logra utilizando el ruteo disponible de RabbitMQ el cual permite asignar un consumidor a un tópico en particular (en este caso, el tópico es el sitio web).

Una vez finalizado el proceso se obtiene un texto reducido y normalizado, que se utiliza como base para poder crear las nubes de palabras.

4. Resultados

La aplicación desarrollada dispone de un tablero de control general, que muestra estadísticas del sistema. En la Figura 2 se muestran, al mes de julio del 2022, los datos correspondientes a la recuperación de artículos desde el 1 de enero de 2020.

Un total de 87.684 artículos extraídos y normalizados, compuestos por 146.390 palabras obtenidas (muchas de las cuales son palabras de baja frecuencia que no poseen ningún significado).

Como puede apreciarse, el sitio *elchubut.com.ar* es el que más artículos publica por día, con casi 50.000, en todo el período; seguido por *diariojornada.com.ar* con 15.883 de a diferencia de los demás sitios contemplados, debido a que abarca una mayor área incluyendo noticias de otras regiones de Argentina.

4.1. Palabras más frecuentes

En la Figura 3, se presentan de las cinco palabras más frecuentes, agrupadas por la cantidad de veces que aparecen en cada sitio:

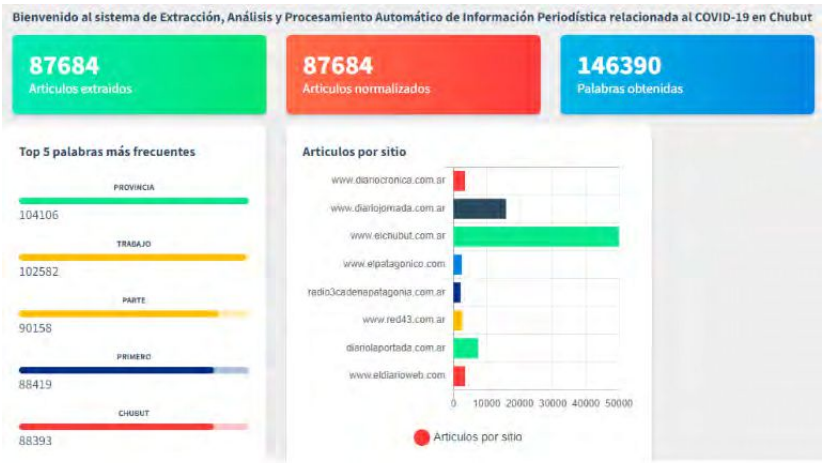


Figura 2: Tablero de control implementado.

4.2. Nube de palabras por sitio

En la Figura 4 se muestran nubes de palabras, donde la frecuencia de las palabras de un sitio particular se ve representada por el tamaño de la palabra en comparación a las demás. Esto permite ver rápidamente cuáles son las palabras obtenidas con mayor frecuencia según el sitio del que se obtuvieron. Las palabras abarcan todos los artículos de su respectivo medio (*dataset* completo).

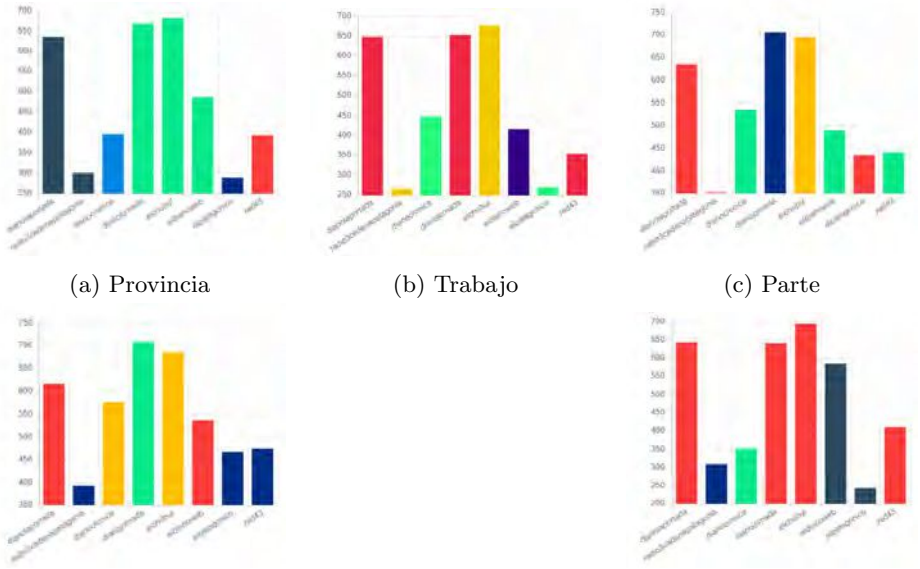
4.3. Nubes de palabras por fecha

Otro tipo de gráfico que se incluyó fueron las nubes de palabras clasificadas por rangos de fechas, el cual permite tener una visión general de todas las palabras y como va evolucionando la frecuencia de las temáticas a lo largo del tiempo.

5. Conclusiones y Trabajos Futuros

En este trabajo se presentó el desarrollo de un sistema para la recuperación, clasificación y análisis de notas periodísticas publicadas en medios digitales. El trabajo se fundamentó en la necesidad urgente, surgida por la pandemia, de contar con información acerca de "lo que ocurría en la sociedad". El desarrollo técnico involucró un análisis de factibilidad tecnológica y del entorno en el que se pensaba operar. En este sentido, la Provincia del Chubut ofició de marco y por ello se buscaron fuentes que tengan representatividad territorial y, a la vez, un volumen considerable de producción.

El desarrollo de este proyecto se dio en el marco del convocatoria COVID Federal del MINCYT. En ese contexto, el proyecto formaba parte de uno mayor,



(a) Provincia (b) Trabajo (c) Parte

(d) Primero (e) Chubut

Figura 3: Frecuencias de palabras por sitio.

que involucraba distintas aristas en el abordaje situacional de la Provincia del Chubut, durante la pandemia. Esa multidimensionalidad y multiescalaridad, se manifestaron en un trabajo interdisciplinario, que marcó los requerimientos de negocio del trabajo aquí presentado. El desarrollo informático posibilitó materializar y explorar las potencialidades de un análisis de este tipo, lo cual definió nuevos requerimientos y demandas.

En línea con los anterior, lo producido en este trabajo será utilizado como base para un futuro trabajo. En particular, el dataset obtenido de más de dos años de notas periodísticas de ocho medios de comunicación de la Provincia del Chubut, será explotado mediante técnicas de Procesamiento de Lenguaje Natural. Especificamente se trabajará sobre modelado de tópicos y modelado dinámico de tópicos, así como técnicas basadas en grafos, a fin de analizar la evolución temporal de distintos temas de interés para los medios de comunicación y, por consiguiente, para la ciudadanía.

Reconocimiento: Los autores quieren expresar un sentido recordatorio y reconocer la labor de la Dra. Florencia del Castillo, quien falleció al tiempo de publicar este artículo, en la dirección del proyecto que posibilitó este trabajo.

(a) Diario Jornada (b) Diario el Chubut (c) Diario el Patagónico

(d) Portal Radio 3 Cadena Patagonia (e) Diario Crónica (f) Diario Red43

(g) Diario La Portada (h) El Diario Web

Figura 4: Nubes de palabras de cada sitio.

Referencias

1. Balcazar, E., Bobadilla, L., Fusiman, W., Ordinez, L.: Geoperfil profesional: una herramienta automática de información sobre profesionales. In: XXVI Congreso Argentino de Ciencias de la Computación (CACIC)(Modalidad virtual, 5 al 9 de octubre de 2020) (2020)
2. Chipidza, W., Akbaripourdibazar, E., Gwanzura, T., Gatto, N.M.: A topic analysis of traditional and social media news coverage of the early covid-19 pandemic and implications for public health communication. medRxiv (2020), `https://www.medrxiv.org/content/early/2020/07/07/2020.07.05.20146894`
3. Chukwusa, E., Johnson, H., Gao, W.: An exploratory analysis of public opinion and sentiments towards covid-19 pandemic using twitter data. (2020)
4. Dousset, B., Mothe, J.: Getting insights from a large corpus of scientific papers on specialisted comprehensive topics-the case of covid-19. Procedia computer science 176, 2287–2296 (2020)
5. Hosseini, P., Hosseini, P., Broniatowski, D.A.: Content analysis of persian/farsi tweets during covid-19 pandemic in iran using nlp. arXiv preprint arXiv:2005.08400 (2020)
6. Martinez, I., Viles, E., Olaizola, I.G.: Data science methodologies: Current challenges and future approaches. Big Data Research 24, 100183 (2021)
7. Nadkarni, P.M., Ohno-Machado, L., Chapman, W.W.: Natural language processing: an introduction. Journal of the American Medical Informatics Association 18(5), 544–551 (09 2011)

(a) Enero 2020 (b) Febrero 2020 (c) Marzo 2020

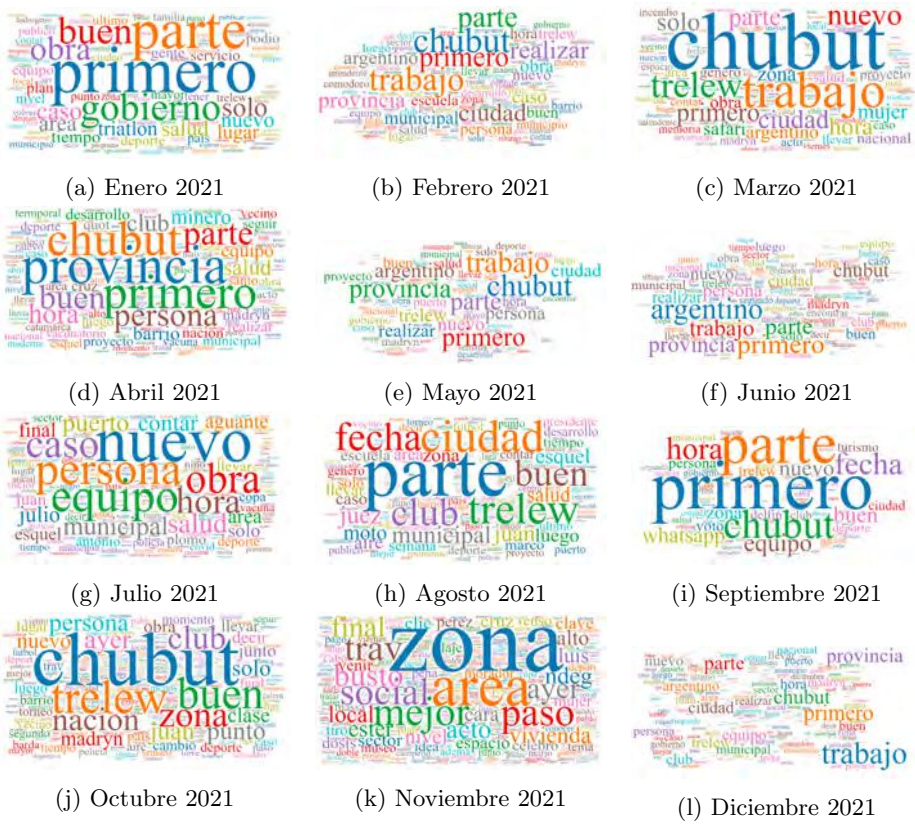
(d) Abril 2020 (e) Mayo 2020 (f) Junio 2020

(g) Julio 2020 (h) Agosto 2020 (i) Septiembre 2020

(j) Octubre 2020 (k) Noviembre 2020 (l) Diciembre 2020

Figura 5: Nubes de palabras año 2020.

8. Perdomo, L., Ordinez, L.: From global to local in the sneakers universe: A data science approach. In: XXVII Congreso Argentino de Ciencias de la Computación (CACIC)(Modalidad virtual, 4 al 8 de octubre de 2021) (2021)
9. Shafique, U., Qaiser, H.: A comparative study of data mining process models (kdd, crisp-dm and semma). International Journal of Innovation and Scientific Research 12(1), 217–222 (2014)
10. Tommasel, A., Diaz-Pace, A., Rodriguez, J.M., Godoy, D.: Capturing social media expressions during the covid-19 pandemic in argentina and forecasting mental health and emotions. arXiv preprint arXiv:2101.04540 (2021)
11. Wirth, R., Hipp, J.: Crisp-dm: Towards a standard process model for data mining. In: Proceedings of the 4th international conference on the practical applications of knowledge discovery and data mining. vol. 1. Springer-Verlag London, UK (2000)
12. Zhang, Y., Chen, J., Liu, B., Yang, Y., Li, H., Zheng, X., Chen, X., Ren, T., Xiong, N.: Covid-19 public opinion and emotion monitoring system based on time series thermal new word mining. arXiv preprint arXiv:2005.11458 (2020)



(a) Enero 2021 (b) Febrero 2021 (c) Marzo 2021

(d) Abril 2021 (e) Mayo 2021 (f) Junio 2021

(g) Julio 2021 (h) Agosto 2021 (i) Septiembre 2021

(j) Octubre 2021 (k) Noviembre 2021 (l) Diciembre 2021

Figura 6: Nubes de palabras año 2021.