UNIVERSIDAD NACIONAL DE ROSARIO (UNR).

FACULTAD DE CIENCIAS EXACTAS, INGENIERÍA Y AGRIMENSURA (FCEIA).

TECNICATURA UNIVERSITARIA EN INTELIGENCIA ARTIFICIAL (TUIA).

PASANTÍA, TUIA.

Informe de la pasantía

Estudiante:

Francisco J. Alomar

AÑO 2025, Rosario.

INTRODUCCIÓN

El siguiente escrito informa la pasantía como asignatura obligatoria de la TUIA, FCEIA, UNR, Argentina. Dada nuestra formación en Antropología[1], optamos por sumarnos al Programa en Investigación y Desarrollo (PID): "Discursos sobre lo público-común plataformizados: caracterización interdisciplinaria de las estrategias enunciativas empleadas en plataformas mediáticas contemporáneas y sus flujos de sentido", ejecutado desde la Secretaría de Ciencia y Tecnología de la UNR, acreditado por el Programa para la Investigación Universitaria Argentina (PRINUAR), bajo la dirección de la Dr. Natalia Raimondo Anselmino y la codirección de la Dr. Irene Gindin, con radicación en el Centro de Investigaciones en Mediatizaciones (CIM) de la Facultad de Ciencia Política y RRII (FCPOLIT) de la UNR. Consideramos que la participación en el PID es de utilidad dado que se constituye en un aporte a la interdisciplina.

El proyecto se focaliza en el estudio de las transformaciones en los discursos sobre lo público-común plataformizados en las sociedades contemporáneas (van Dijck, Poell y de Waal, 2018). Para examinar las estrategias enunciativas (Raimondo Anselmino, 2011) y los flujos de sentido que emergen, se toma un caso: los discursos difundidos en portales de noticias, redes sociales y aplicaciones de mensajería, sobre una serie de hechos violentos ocurridos en Rosario, Argentina, entre marzo y mayo de 2024. Los eventos relacionados con el narcoterrorismo implicaron el uso de la fuerza por parte de organizaciones criminales para presionar al poder político provincial, generando un clima de alarma social que paralizó numerosas actividades e instituciones, y que requirió la intervención de las autoridades nacionales y fuerzas federales. El análisis se basa en un conjunto de publicaciones, compuesto por: a) artículos publicados en seis portales de noticias, tanto locales como nacionales (lacapital, rosario3, elciudadanoweb; lanacion, infobae y clarin), junto con los comentarios de lectores cuando están disponibles; b) publicaciones relacionadas realizadas en las cuentas oficiales de *Facebook*, *Instagram* y *X* de estos portales, incluyendo las interacciones de los usuarios; c) publicaciones en esas mismas plataformas de funcionarios públicos de distintos niveles de gobierno; y d) mensajes (texto, imágenes, audio y video) compartidos por la población a través de *WhatsApp*.

Del universo de los distintos puntos, abordamos una arista. El objetivo principal es analizar textos periodísticos y su repercusión en la red social *Facebook*, a través de scripts escritos en *Python* e implementados en *Google Colaboratory*. Esto se logra mediante la

[1] Licenciatura en Antropología, título otorgado por Escuela de Antropología, Facultad de Humanidades y Artes (FHyA), UNR.

extracción automatizada de artículos de diversos portales nacionales y locales y de los comentarios en *Facebook* asociados. Para ello, es necesario procesar y estructurar los datos en una base organizada. Posteriormente, transformamos los textos en vectores usando técnicas de *embeddings* y, luego, reducimos su dimensionalidad. Finalmente, se aplica un algoritmo de *clustering* para identificar patrones que agrupen los datos.

Pues, podemos establecer objetivos específicos:

a) Construir una base de datos de artículos periodísticos relacionados con la temática a partir de distintos portales de noticias nacionales y locales.
b) Extraer automáticamente comentarios de *Facebook* sobre artículos de portales locales y agregarlos a la base de datos.
c) Transformar los textos en representaciones numéricas mediante *embeddings*.
d) Reducir la dimensionalidad de los vectores para facilitar su procesamiento, visualización y análisis.
e) Aplicar el algoritmo de *k-means* para identificar patrones en los datos[2].

Para alcanzar los objetivos, definimos actividades concretas:

a) Implementar técnicas de *scraping* y recopilar artículos de los portales lacapital, rosario3, elciudadano, lanacion, infobae y clarin.
b) Procesar los datos extraídos y organizar un *dataset* estructurado con los atributos: medio, url, fecha, titulo, bajada, cuerpo_texto, comentarios_fb.
c) Implementar técnicas de *scraping* para recopilar en la base de datos los comentarios de *Facebook* sobre artículos de medios locales (lacapital, rosario3, elciudadano).
d) Convertir los textos en vectores de 512 dimensiones mediante la biblioteca *Sentence Transformer.*
e) Aplicar *Uniform Manifold Approximation and Projection* (UMAP) para reducir la dimensionalidad de los *embeddings* a 3 y 2 dimensiones.
f) Ejecutar sobre el nuevo conjunto de datos numéricos el algoritmo de aprendizaje no supervisado *k-means* que permite determinar *clusters* a partir justamente de centroides (*k*) y la distancia media aritmética (*means*) de los datos que comprenden.

Para guiarnos en el análisis, planteamos tres hipótesis:

[2] Hay un artículo previo en el que se aplica la metodología utilizada por nosotros (algoritmo *k-means* y métricas asociadas), aunque con otra implementación, hipótesis y objetivos. Aquel escrito aplica técnicas de *clustering* para el estudio de géneros periodísticos (Raimondo, Rostagno y Cardoso, 2021).

a) Dado que los artículos tratan una misma temática es esperable que los *embeddings* se distribuyan en un espacio relativamente compacto. Esto implicaría que los artículos estén próximos dentro del espacio de representación vectorial, pudiendo reflejar similitudes en el contenido y estructura discursiva.
b) Los clusters identificados no estarían necesariamente delimitados por el medio de origen, sino que agruparán artículos de distintos medios (locales y nacionales) y de diferentes fechas. Los agrupamientos reflejarían patrones discursivos compartidos, tales como estrategias enunciativas similares, subtemáticas recurrentes o el uso frecuente de determinados sintagmas que permitan identificar isotopías en la construcción de sentido.
c) Los comentarios de *Facebook* formarían un agrupamiento distinto de los artículos periodísticos. Aunque estén vinculados a los textos de los que emergen, se proyectarán en un espacio de representación propio, alejados. Esto permitiría no solo corroborar la hipótesis (b) en relación con el agrupamiento de comentarios, sino también analizar la distancia geométrica entre la producción periodística y la respuesta de los lectores, identificando posibles transformaciones discursivas en la transición del discurso periodístico al discurso en la red social *Facebook*.

El informe se estructura en dos apartados. En el primero exponemos la construcción del *dataset*. En el segundo, explicamos el código generado para aplicar la minería de datos, gráficos y métricas obtenidas. En cada apartado se establecen hipervínculos hacia un repositorio de *Github* en el que se encuentran los archivos del trabajo realizado. Luego, exponemos los resultados. Hacia el final deducimos conclusiones en relación a los resultados y las hipótesis, planteando, además, nuevas líneas de trabajo que podrían robustecer el análisis realizado como parte del estudio mayor en el que se incluye.

1. Construcción de *dataset*

Durante el transcurso de la carrera, en cada materia específica del campo de la IA (Minería de datos, Aprendizaje Automático I y II, etc.), se nos repetía casi a modo de mantra que el 80% del trabajo en las disciplinas relacionadas a la ciencia y análisis de datos es, justamente, sobre los datos: su depuración, integridad, imputación, obtención, etc.; y, que la aplicación de algoritmos particulares que generan modelos, es sólo el 20%. Aunque el porcentaje exacto es debatible, la idea general ha sido confirmada en la práctica. Comenzamos la construcción del *dataset* en marzo de 2024 y nos llevó unos 8 meses hasta obtener una presentación aceptable para el procesamiento.

En esta sección no ahondamos en los códigos relacionados al *scraping* dado que su aporte a futuros trabajos es limitado. Los algoritmos fueron generados de modo *ad hoc*, o sea, particular y artesanalmente para cada portal de noticias. Incluso, en el lapso de tiempo mencionado tuvimos que variar algunos de ellos por modificaciones de los sitios *web* que "escrapeábamos". Por el contrario, expondremos una generalidad de cómo fue el proceso, principales bibliotecas utilizadas, dificultades, y, por último, el *dataset*.

1.1. Dificultades en la construcción del *dataset*

El principal desafío fue diseñar algoritmos para cada portal. Estudiamos cada *web* en particular, su diseño html y carga dinámica, los códigos java, incrustaciones *iframes*[3] y demás particularidades que varían entre los sitios. En la sección siguiente comentamos las dos principales bibliotecas utilizadas en la implementación de los códigos de *scraping*.

Por otra parte, los portales, en general, tienen un número limitado de noticias a las que acceder en determinado lapso de tiempo. Es decir, desde un portal periodístico sólo se puede acceder a las noticias durante un determinado período de tiempo; en cambio, si tuviéramos su *url*, probablemente podríamos acceder en otros tiempos. Es lógico que los sitios *web* administren los recursos de este modo dado que el periodismo construye textualidades sobre el presente en sentido lato y, aunque haya interés por cuestiones históricas, las noticias después de cierto tiempo pierden valor de actualidad. Esto dificulta ir "para atrás" en el *scraping* de noticias después de determinado tiempo ya que no están disponibles en los portales pasado cierto período.

---

[3] Los *inline frame* de una página web son elementos *HTML* que permiten insertar otro documento *HTML* dentro de la misma página.

Al comienzo de la construcción del *dataset* contamos con una herramienta diseñada para extraer noticias de portales *web*[4]. El problema fue que debido a cuestiones técnicas, no se recolectaron los artículos del primer mes del recorte temporal de la temática (marzo 2024). Pues, diseñamos otras estrategias para extraer los datos necesarios, como recurrir al repositorio *Wayback Machine*[5] o búsquedas automatizadas en *Google*.

1.2. Construcción del *dataset*

1.2.1. Bibliotecas utilizadas

Para la construcción de la base de datos utilizamos principalmente dos bibliotecas: *Selenium* y *Newspaper3k*[6]. La primera permite automatizar la navegación *web* mediante una ejecución en segundo plano y, a su vez, simular la interacción con las páginas como usuario humano, por ejemplo, hacer *clicks* sobre botones, desplazarse por ellas, eludir publicidades, establecer tiempos de espera un tanto aleatorios, etc. *Selenium* resulta de gran utilidad para el *scraping* dado que su implementación resuelve la carga dinámica de los sitios: hay información a la se accede sólo si uno interactúa con la página. Este tipo de diseño *web* se relaciona con la economía de recursos mostrando sólo la información que un usuario deseara ver, pero dificulta el *scraping* dado que los datos no están disponibles y se cargan a medida que se navega. Con *Selenium* obtuvimos las *urls* de los artículos y los comentarios de *Facebook* vinculados que estaban disponibles mediante incrustaciones *iframes*.

Una vez conseguidas las *urls* de cada artículo, implementamos algoritmos con *Newspaper3k*. La biblioteca facilita la extracción de información relevante como el título, bajada y cuerpo del texto de los artículos. Decimos que "facilita", porque, en pocas líneas de código se obtienen, además, la fecha, palabras claves y resumen de los artículos, mediante Procesamiento de Lenguaje Natural (PLN).

1.2.2. Estrategias de selección de artículos y decisiones de recolección de datos

La selección fue otro de los problemas porque la temática que engloba al conjunto de artículos es demasiado específica. Si la búsqueda a desarrollar hubiera sido, por ejemplo, sobre secciones temáticas periodísticas (política, deporte, economía, etc.), nos habría

[4] La herramienta informática fue desarrollada por el Centro de Altos Estudios en Tecnología Informática (CAETI) de la Universidad Abierta Interamericana, en su sede Rosario.
[5] Archivo digital que permite recuperar versiones anteriores de sitios web: https://web.archive.org/.
[6] La documentación de las bibliotecas se encuentra disponible en: https://www.selenium.dev/documentation/?utm_source=chatgpt.com y https://newspaper.readthedocs.io/en/latest/?utm_source=chatgpt.com, respectivamente.

resultado más sencillo diseñar un modelo de IA y PLN (supervisado) para seleccionar los artículos, ya que es factible encontrar numerosos y extensos *set* de datos etiquetados para entrenar y testear un modelo tal. Pero este no fue el caso. La temática como se describe en la introducción es particular, y, además, su recorte temporal resulta acotado. Tal es así, que hay noticias que se vinculan al narcotráfico en la ciudad de Rosario durante este período, involucran asesinatos, amenazas, y, dados los criterios que diseñó el equipo, no resultan pertinentes. Además, el narcoterrorismo que se vivenció en la ciudad de Rosario fue de algún modo transversal, por lo que una noticia que sirviera al estudio del caso podría hallarse en la sección económica, policial o política de un portal.

Dado este escenario, diseñamos con el equipo la siguiente estrategia[7]:

a) Definimos sintagmas nominales que fueran probables de ser mencionados en los en los textos[8].

b) Si ninguno de los sintagmas aparecía en el texto (título, bajada, cuerpo del texto), ni tampoco en los resúmenes automáticos generados por *Newspaper3k* de cada artículo, el texto sería descartado.

c) En el caso de los diarios nacionales, ponderamos la aparición de la palabra "rosario" para la selección del artículo, criterio no aplicable para los diarios locales, que aluden a la ciudad de distintos modos y no siempre la llaman por su nombre.

Es evidente que planteamos un *threshold* para rechazar bien los artículos, pero no así aceptarlos. En este punto, el filtrado y la selección se hicieron de forma manual sin aplicar procesamiento automático, y se realizó por los integrantes del equipo del PID a quienes agradecemos su colaboración[9].

Una vez obtenidos los artículos pertinentes para el caso, recolectamos los comentarios de *Facebook*. En primer lugar decidimos trabajar con los comentarios de portales locales de la ciudad de Rosario. En segundo lugar, recolectamos los comentarios de manera desatendida,

---

[7] Desconocemos los criterios de selección de la herramienta automática, aunque sí sabemos que se utilizaron los sintagmas_clasificadores. La estrategia que describimos sólo fue aplicada a los artículos que nosotros recolectamos.

[8] Estos sitntagmas los intodujimos en la variable sintagmas_clasificadores = ["amenaza", "armadas", "armado", "armados", "asesinada", "asesinadas", "asesinado", "asesinados", "asesinato", "asesinatos", "ataque", "ataques", "baleada", "baleadas", "baleado", "baleados", "bullrich", "ciudadano", "civil", "cococcioni", "colectivero", "colectiveros", "colectivo", "colectivos", "conmoción", "crimen", "crímenes", "cárcel", "cárceles", "ejército", "estaciones", "estación", "fuerzas", "gendarmería", "javkin", "mafia", "mafioso", "milei", "narco", "narcos", "narcoterrorismo", "narcotráfico", "ola", "paquete", "penitenciario", "petri", "playero", "playeros", "policial", "policía", "policías", "pullaro", "rosario", "sangre", "seguridad", "servicio", "sicario", "sicarios", "tachero", "tacheros", "taxi", "taxis", "taxista", "taxistas", "terror", "terrorismo", "violencia"].

[9] Natalia Raimondo Anselmino, Irene Gindin, Emanuel A. Pérez Zamora y Daniela Sánchez. fueron quienes se dedicaron a la selección. Además, identificaron artículos que no fueron captados por la herramienta o los códigos diseñados por nosotros, que luego procesamos y agregamos al *set* de datos.

es decir, se cargaron en el a *dataset* bajo la pauta de estar relacionados al artículo. En el proceso se perdió información relevante según el tipo de análisis que se planifique, como la estructura dialogal de los comentarios[10] o metadatos relacionados (*likes, replies* y tiempo transcurrido desde el posteo).

1.2.3. Depuración del d*ataset.*

Al generar la recolección de artículos y comentarios obtuvimos información no deseada. En los comentarios de *Facebook*, "levantamos" publicidades (mayormente en inglés), o, desde el cuerpo del texto de los artículos y de modo variable, comentarios de la plataforma *X*, publicidades e hipervínculos a otras noticias. Para limpiar el *set* de datos detectamos primero las estructuras repetidas a extraer del texto, explorando manualmente el *dataset*. Una vez identificadas, diseñamos algoritmos que mediante expresiones regulares las sustrajeran del corpus. Por ejemplo, para el caso del portal La Capital, en el cuerpo de texto aparecen a menudo conexiones a otras noticias que comienzan con ">> Leer más:", expresión seguida de un hipervínculo y que se encuentra entre saltos de línea ("\n"). Entonces, la lógica a seguir para el sitio *web* fue quitar del corpus todo lo que comience con ">> Leer más:", situado entre saltos de líneas. En las publicidades de *Facebook* se utilizó la biblioteca *Langdetect*, con la cual se identificaron y eliminaron las publicidades en inglés.

1.2.4. Presentación del *set* de datos.

Debido a las particularidades de los portales desarrollamos el trabajo descrito para cada uno de ellos. Sin embargo, dada la heterogeneidad de la fuente de los datos (distintos portales) y los distintos *approaches* adoptados en su recolección (códigos generados por nosotros, herramienta automática de *scraping*), debimos normalizarlos, garantizar su integridad (por ejemplo, quitar registros repetidos) y concatenarlos en un solo *dataset*. Este proceso de generación del *set* de datos final está implementado en código generacion_dataset.ipynb, el cual toma los archivos individuales de cada medio también disponibles en el repositorio, dentro de la carpeta csv_por_medio para, finalmente, generar el archivo dataset_medios_narcoterrorimo_rosario.csv.

El *dataset* contiene 599 registros y 7 columnas, cada una con información sobre noticias relacionadas con el narcoterrorismo en Rosario. Las columnas son:

---

[10] Por ejemplo, suponiendo que para una noticia participaron dos personas de la cuales una de ellas "posteó" diez comentarios seguidos, y la otra, le contestó sólo en uno. Para este hipotético caso, la recolección contaría con once comentarios, pero sin distinguir que los diez primeros fueron de una persona, y el último, que enunció en respuesta hacia aquella, de otra.

- medio: contiene el nombre del medio de comunicación (clarin, lacapital, rosario3, infobae, lanacion, elciudadano).
- url: contiene el enlace a la noticia original.
- fecha: contiene la fecha de publicación en formato YYYY-MM-DD.
- titulo: contiene el título original de la noticia.
- bajada: contiene el resumen introductorio del artículo.
- cuerpo_ texto: contiene el cuerpo de texto del artículo.
- comentarios_fb: contiene en una lista los comentarios extraídos de la sección de *Facebook* de cada artículo. Sólo se recopilaron de medios locales (lacapital, rosario3 y elciudadano). En algunos registros esta columna contiene valores nulos (NaN).

2. Análisis sobre el conjunto de datos

En este apartado convertimos a *embeddings* el texto de los artículos y comentarios, reducimos su dimensionalidad para luego aplicar el algoritmo no supervisado *k-means*.

2.1. Transformación del texto a *embeddings*

La vectorización del texto está implementada en el código transformacion_ebeddings_dataset.ipynb. La biblioteca utilizada es *Sentence Transformers*. Básicamente, para cada registro del corpus, tomamos las columnas titulo, bajada, cuerpo_texto y comentarios_fb, y convertimos su contenido a una representación vectorial de 512 dimensiones. Además, normalizamos los vectores mediante *Min-Max Scaling* definiendo las componentes en el rango [-1, 1]. La estandarización es necesaria para que al aplicar *k-means* cada dimensión de los vectores contribuya de manera equitativa al proceso de *clustering*, ya que el algoritmo se basa en distancias euclidianas y es sensible a la escala de los datos. El código genera el archivo dataset_embedding_normalizado.parquet; las columnas indicadas contienen *embedding* respectivos a los textos de *dataset* original.

2.2. Reducción de dimensionalidad de los *embeddings*

La reducción dimensionalidad es necesaria por dos motivos. El primero es la "maldición de la dimensionalidad", expresión referida en la ciencia de datos a los problemas que surgen al trabajar en espacios de alta dimensión (en nuestro caso, *embeddings* de 512 componentes). A medida que aumentan, la distancia media entre los datos se incrementa, pero lo más crítico es que la variabilidad entre las distancias disminuye exponencialmente, haciendo que casi todas las instancias queden igualmente lejos unas de otras. Esto provoca

que los datos se vuelvan escasos en el espacio, y no se distingan vectores cercanos de lejanos. En el caso de *k-means*, que se basa en distancias para agrupar datos, este fenómeno es especialmente perjudicial. Cuando la diferencia relativa entre distancias se vuelve mínima, el algoritmo tiene dificultades para identificar *clusters* significativos ya que los centroides se vuelven menos representativos y se asignan de manera poco precisa a los agrupamientos.

La segunda razón es obtener una representación más comprensible de los datos. Aunque en espacios de alta dimensión se conservan propiedades matemáticas importantes (como operaciones que involucran el coseno entre vectores), representar un espacio como R512 resulta inherentemente complejo y difícil de visualizar. Reducir los vectores a 2 y 3 dimensiones permite aproximarnos a la estructura del conjunto de datos, facilitando su análisis e interpretación.

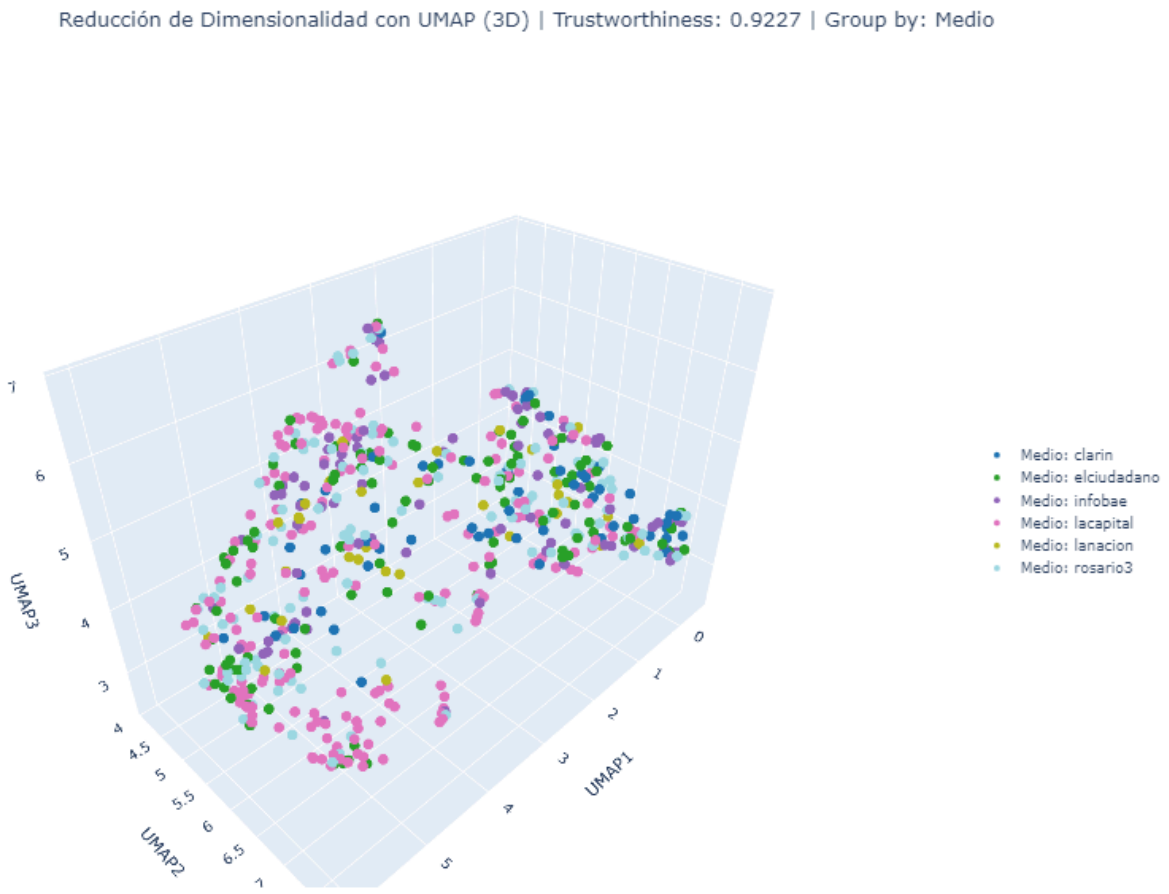
Existen diversas técnicas de reducción de dimensionalidad. Hemos optado por UMAP, algoritmo que preserva las estructuras locales y globales de los datos y es capaz de desenrollar[11] la estructura del espacio de alta dimensión, facilitando la visualización y exploración de relaciones intrincadas. Su objetivo es construir una representación en baja dimensión que conserve la integridad del espacio original (Géron, 2023). Además, utilizamos la métrica *trustworthiness* la cual indica qué tan bien se preserva la estructura local de los datos tras la reducción de dimensionalidad. La métrica se evalúa en un rango de 0 a 1, donde valores cercanos a 1 muestran que la vecindad original de cada punto se mantiene en la representación de baja dimensión. En pruebas sucesivas, con diferentes técnicas[12], la métrica demostró que UMAP ofrecía los mejores resultados.

Es importante aclarar que los *embeddings* de los artículos, que incluyen el vector del título, la bajada y cuerpo del texto, los promediamos en un solo vector . Hicimos esto para que cada artículo se represente por un sólo punto en los gráficos y, además, para poder compararlos con los comentarios de *Facebook*, los cuales también fueron vectorizados en *embeddings* de 512 dimensiones. A continuación presentamos dos gráficos que muestran al conjunto de vectores (sin los comentarios de *Facebook*) ya promediados y reducidos en su

[11] El término no debe entenderse en su acepción coloquial, sino como una noción específica dentro del análisis de datos no lineales y la reducción de dimensionalidad (*manifold learning*). Utilizamos el verbo porque proviene del concepto *Swiss Roll*, un conjunto de datos sintéticos que representa una variedad bidimensional embebida en un espacio tridimensional en forma de espiral. En este sentido, "desenrollar" hace referencia al proceso mediante el cual un algoritmo como UMAP reconstruye la estructura de los datos al mapearlos a un espacio de menor dimensión, preservando su geometría local y relaciones de proximidad (Géron, 2023).

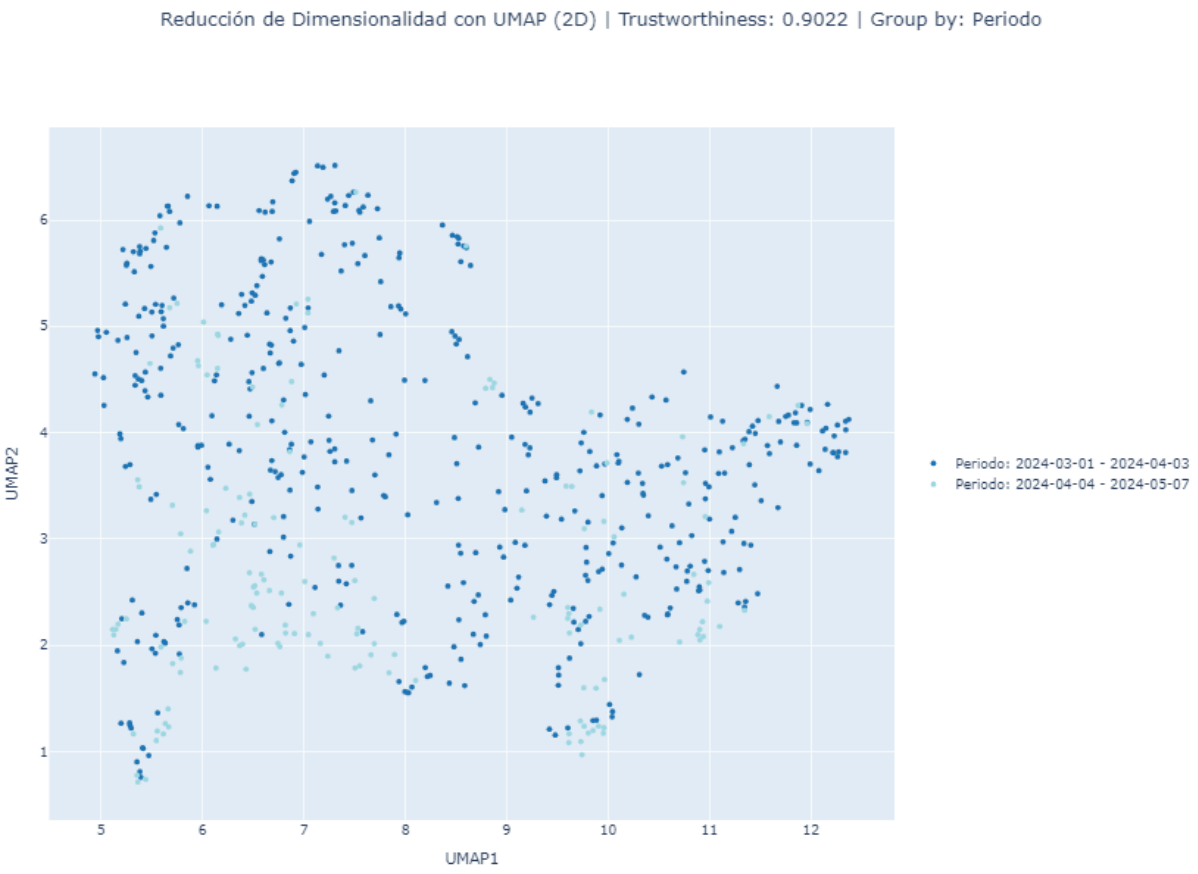
[12] Evaluamos diversas técnicas de reducción, en particular *Principal Component Analysis* (PCA)*, t-Distributed Stochastic Neighbor Embedding* (t-SNE) e *Isometric Mapping (Isomap).*

dimensionalidad a 3D y 2D, donde cada gráfico muestra la métrica *trustworthiness*[13] y un etiquetado trivial de los datos:



G. 2.2.A

[13] Esta métrica se importa de la biblioteca *Scikit-Learn* a partir del módulo *Manifold* (*from sklearn.manifold import trustworthiness*).



G. 2.2.B

El "etiquetado trivial" tiene como función mostrar que la división por fechas y por medio no explican agrupamientos de los artículos, sino que, por el contrario, denotan una "mezcla" de ellos. Es aquí donde el aprendizaje no supervisado puede otorgar su mayor aporte al análisis de los datos, estableciendo relaciones no evidentes. El algoritmo *k-means* minimiza la suma de las distancias entre los puntos y sus centroides, revelando agrupamientos que emergen de la optimización de la distancia total. Posteriormente, sería necesario formular hipótesis que expliquen las posibles razones detrás de los agrupamientos y las correlaciones observadas.

2.3. Implementación de *k-means*

Tanto la reducción de dimensionalidad, la determinación del número de *cluster* como la ejecución del algoritmo *k-means* se encuentra en el código umap_kmeans_narcoterrorimos_rosario.ipynb, que toma como entrada el archivo dataset_embedding_normalizado.parquet y genera todo el procesamiento consecuente.

El algoritmo *k-means* es una técnica ampliamente utilizada para segmentar datos en grupos. Se fija un número k de *clusters* y se seleccionan aleatoriamente puntos que actúan como centroides iniciales (inicialización aleatoria). Cada dato se asigna al centroide más cercano y, tras ello, se recalculan las posiciones de los centroides en función de los datos. Este ciclo se repite hasta que las agrupaciones se estabilizan. En esencia, *k-means* resuelve un problema de optimización al minimizar la suma de las distancias cuadráticas entre cada dato y el centroide de su grupo.

La función construida en el código def aplicar_kmeans() es la encargada de implementar el algoritmo. En ella se definen 4 hiperparámetros:

- n_clusters: Establece el número de clusters a formar (este valor se pasa como parámetro a la función).
- random_state=42: Garantiza la reproducibilidad de los resultados fijando una semilla para la generación de números aleatorios.
- n_init=50: Especifica el número de veces que se ejecutará el algoritmo con diferentes inicializaciones, eligiendo la solución óptima. Para el tamaño del *set* de datos, este número resulta más que adecuado debido a que ejecutar el algoritmo 50 veces para encontrar la mejor solución no supone un costo computacional significativo y aumenta la robustez al evitar caer en óptimos locales.
- init='k-means++': Determina el método para la inicialización de los centroides, utilizando el algoritmo *k-means*++, reduciendo la posibilidad de quedar “atrapado” en un óptimo local, mejorando la convergencia y calidad final de los *clusters*.

Consideramos que la elección del número de n_clusters es de gran importancia porque *k-means* separará al conjunto de datos en 2, 3, …, o n agrupamientos, tiene consecuencias directa en la explicación que podamos extraer: no es lo mismo dividir el *set* de datos, por ejemplo, en 2, 3 o 9 *clusters*, porque, luego será menester dar cuenta de las diferenciaciones. Al ser k un hiperparámetro, el algoritmo encontrará tantas divisiones como le asignemos. Para comprender la intuición tras el argumento, planteemos los casos más extremos: uno, que considerare un sólo *cluster,* y otro, en el que haya un número de *cluster* cercano al de vectores. El primero, mostraría que los artículos se encuentran cercanos unos a otros, sólo evidenciando la correlación directa de pertenecer a una temática; el segundo, nos depara a una obviedad extrema: los vectores son todos distintos. Pues, determinar el número de *clusters* es crucial. Sobre este punto en particular versamos en la siguiente sección.

2.4. Determinación del número de *clusters*

Podemos hallar las cantidades de agrupaciones al explorar la variación del hiperparámetro y evaluar cómo el algoritmo separa los datos. No obstante, contamos con métricas desarrolladas para tal fin. En el análisis utilizamos tres.

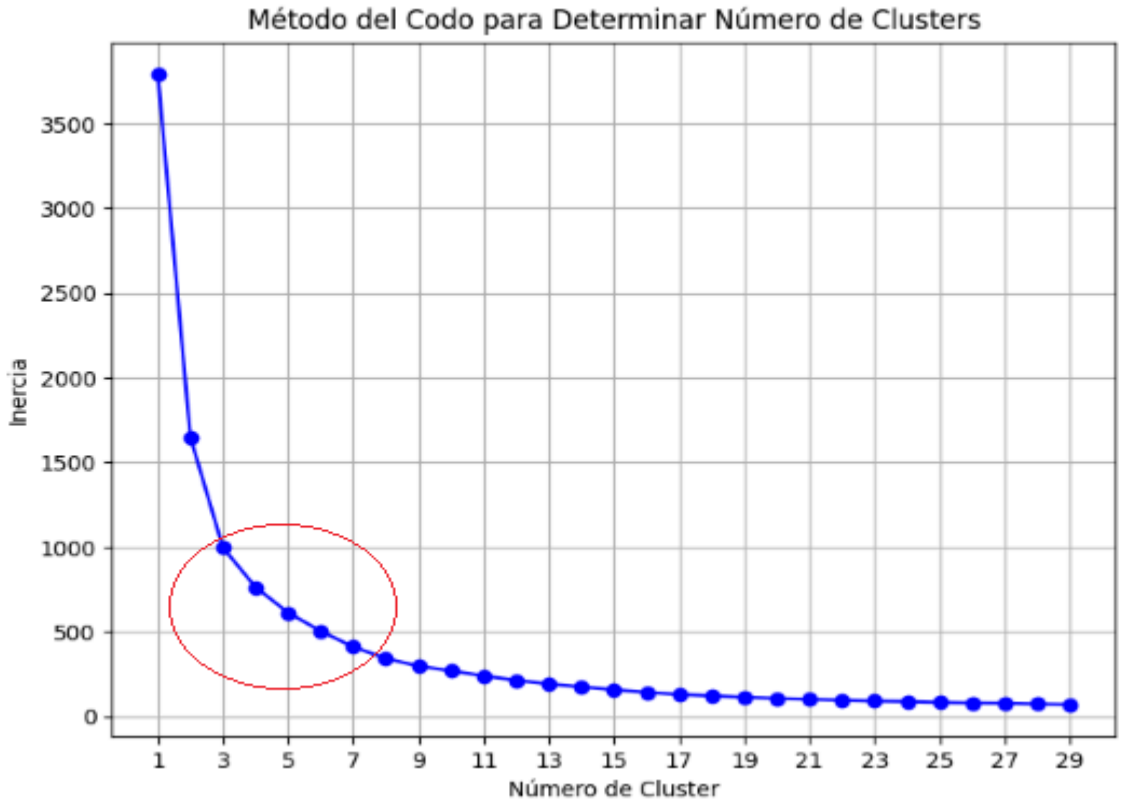
Antes de explicar las métricas quisiéramos retomar un principio que aprendimos durante la TUIA, ha guiado el trabajo de la pasantía y resultó particularmente útil en la interpretación de las métricas. Nos referimos al principio de Guillermo de Ockham: "Cuando se ofrecen dos o más explicaciones de un fenómeno, es preferible la explicación completa más simple; es decir, no deben multiplicarse las entidades sin necesidad." (Wikipedia, 2025). En términos de construcción de modelos que, por definición, son representaciones simplificadas aunque explicativas de una realidad, el principio resulta productivo. Además de figurar cierta estética desde la economía del pensamiento, nos alienta a diseñar generalidades simples, lo cual a menudo resulta difícil. El principio dice que no debemos, innecesariamente, introducir una pluralidad y persuade que "menos es más". Se comprenderá mejor la potencialidad de la Navaja de Ockham al interpretar las métricas, y, en consecuencia, al obtener la cantidad de *clusters* (k).

La primera métrica es *Gap Statistic* que determina el número de *clusters* al comparar la dispersión de datos reales con la esperada en un conjunto aleatorio (Zaki y Meir, 2020). Para cada k se ejecuta con *Optimalk*[14] el algoritmo *k-means* sobre ambos conjuntos y se mide la diferencia en la estructura resultante. El *gap* se calcula como la diferencia entre el logaritmo de la dispersión de los datos reales y aleatorios, eligiendo k cuando el valor deja de aumentar significativamente. El problema que hallamos es que el número de *clusters* arrojado por la métrica, en general, oscila alrededor de 12, en un rango de [2, 15]. La cantidad es poco adecuada para dar cuenta de una posterior clasificación de artículos debido que, justamente, se introduce demasiada pluralidad en un conjunto relativamente pequeño (599 vectores promediados como representación de los artículos, 335 comentarios de *Facebook*). *Gap Statistic* compara la dispersión de los *clusters* con una referencia aleatoria. Si la variación entre los agrupamientos es leve, el método tendería a aumentar k hasta encontrar una diferencia significativa, y puede llevar a sobreestimar el número de agrupamientos. Variando el hiperparámetro n_iter a números elevados (7000), que determina cuántas veces se generan *dataset* aleatorios para comparar la dispersión de los *clusters*, llegamos a resultados óptimos de la elección de k. El problema es que el tiempo de ejecución aumenta considerablemente.

[14] *OptimalK* es una clase dentro de *gap_statistic*, biblioteca que implementa *Gap Statistic* (*from gap_statistic import OptimalK*). La documentación está disponible en https://www.kaggle.com/code/mallikarjunaj/gap-statistics.

Al aplicar el principio de Ockham (no se debe hacer con más lo que puede lograrse con menos) consideramos otras métricas y técnicas para obtener de modo más eficiente el número de k.

La dispersión *Gap Statistic* se mide a través de la inercia[15], que representa la compactación de los *clusters*. Ahora bien, la métrica también se evalúa, directamente, acumulando sus valores para distintos k. Graficamos la inercia en función de k para aplicar el método del codo y así identificar el número óptimo de *clusters*. El método busca puntos de inflexión en la curva (Géron, 2023)[16], donde la reducción de la inercia deja de ser significativa, sugiriendo números de k adecuados. Hay una zona de la gráfica en que la pendiente disminuye, o dicho de otro modo, la derivada de la curva resultante se reduce, lo que indica que agregar más *clusters* no mejora la compactación de los datos. El siguiente gráfico muestra la evolución de la inercia a medida que aumenta k y destacamos con rojo la zona que comprende los valores de *clusters* a elegir según el principio del codo:



[15] En la documentación de *Scikit Learn* para los módulos de *clustering* se explica el concepto: https://scikit-learn.org/stable/modules/clustering.html

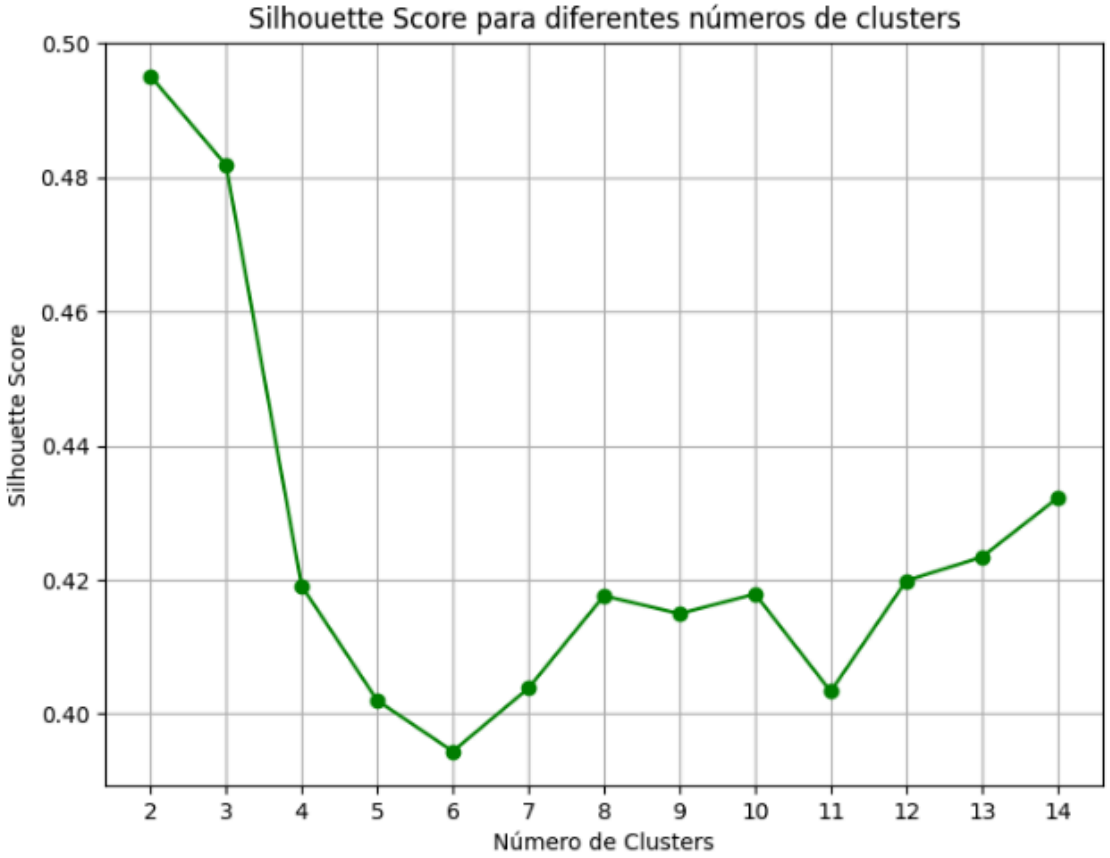
[16] En el libro citado se explica el método del codo (inflexión de la curva) para la elección de números de componentes principales en el mecanismo lineal de reducción de dimensionalidad *Principal component analysis* (PCA). El método es aplicable a la inercia de la elección de *clusters* en *k-means*. También se utiliza en la implementación de *Gap Statistic,* documentación disponible en https://www.kaggle.com/code/mallikarjunaj/gap-statistics

G. 2.4.A

Siguiendo a Ockham ¿cuál de los valores resulta más adecuado? En nuestra opinión 3 conserva relativa simplicidad y otorgaría cierta explicabilidad en la división de los datos. El problema con la inercia es que depende de modo particular del número de *clusters*: aumentar k siempre la reduce (sobreajuste). Por otra parte, la inercia no mide la separación entre *clusters*, sólo evalúa la compactación. Es por ello que utilizamos otra métrica para sortear esta limitación.

Nos referimos a la métrica *silhouette*[17] que además de evaluar la cohesión dentro de un *cluster* tiene en cuenta la separación respecto de otros. Es decir, que se maximiza cuando los agrupamientos están bien definidos y separados. En términos intuitivos, a diferencia la inercia, *silhouette* no siempre disminuye cuando a aumenta k. Su valor oscila en el rango [-1, 1]: si el valor es cercano a 1, significa que los grupos están bien separados unos de otros, claramente distinguidos; si ronda en las inmediaciones de 0, da cuenta que los *clusters* son indiferentes y su distancia es insignificante; si adopta valores cercanos a -1 indica que los grupos están asignados de forma incorrecta (Zaki y Meir, 2020). A continuación, graficamos la métrica en función de k:

[17] La métrica *silhouette_score* se importa del módulo *metrics* de la biblioteca *Scikit-learn* (*from sklearn.metrics import silhouette_score*). Documentación disponible en https://scikit-learn.org/stable/modules/generated/sklearn.metrics.silhouette_score.html



G.2.4.B

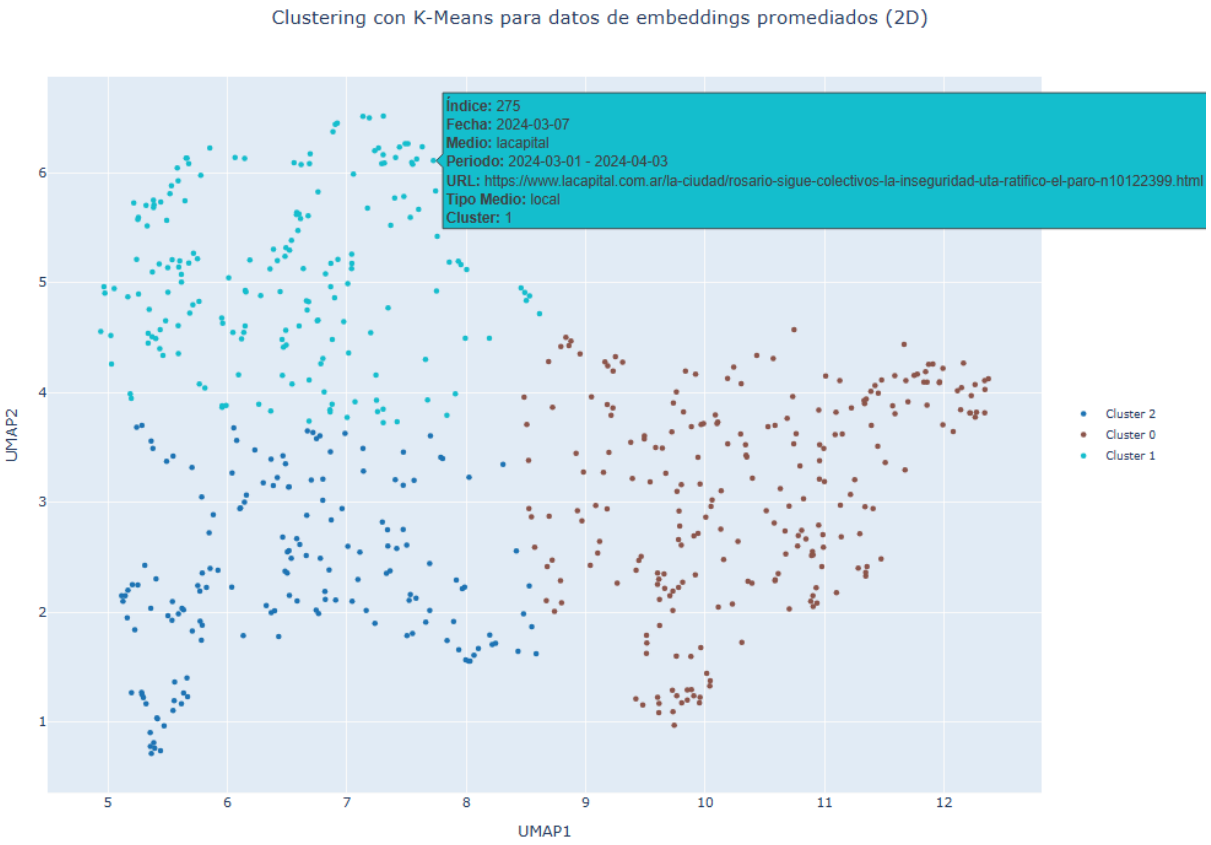
La métrica otorga el mayor valor para k igual a 2. Sin embargo, cuando es 3, la diferencia entre ambas cantidades es menor a 0.02. Consideramos que la métrica, a su vez, refuerza la elección de 3 *clusters* dado que en el gráfico de la inercia y mediante la técnica del codo se podría haber optado para valores de k entre 3 y 7. Sin embargo, en el gráfico G.2.4.B se evidencia cómo disminuye el valor de la métrica *silhouette* cuando k aumenta en ese rango.

3. Resultados

En esta sección presentamos los resultados. Todos los gráficos (2D y 3D) e implementación algorítmica del trabajo se encuentran en el archivo umap_kmeans_narcoterrorimos_rosario.ipynb. Además, explicamos el uso de una sencilla interfaz del código que permite explorar los datos.

3.1. Representación de *clusters*

Con una elección de 3 *clusters* como hiperparámentro, *k-means* devuelve los siguientes agrupamientos de artículos:



G.3.1.A

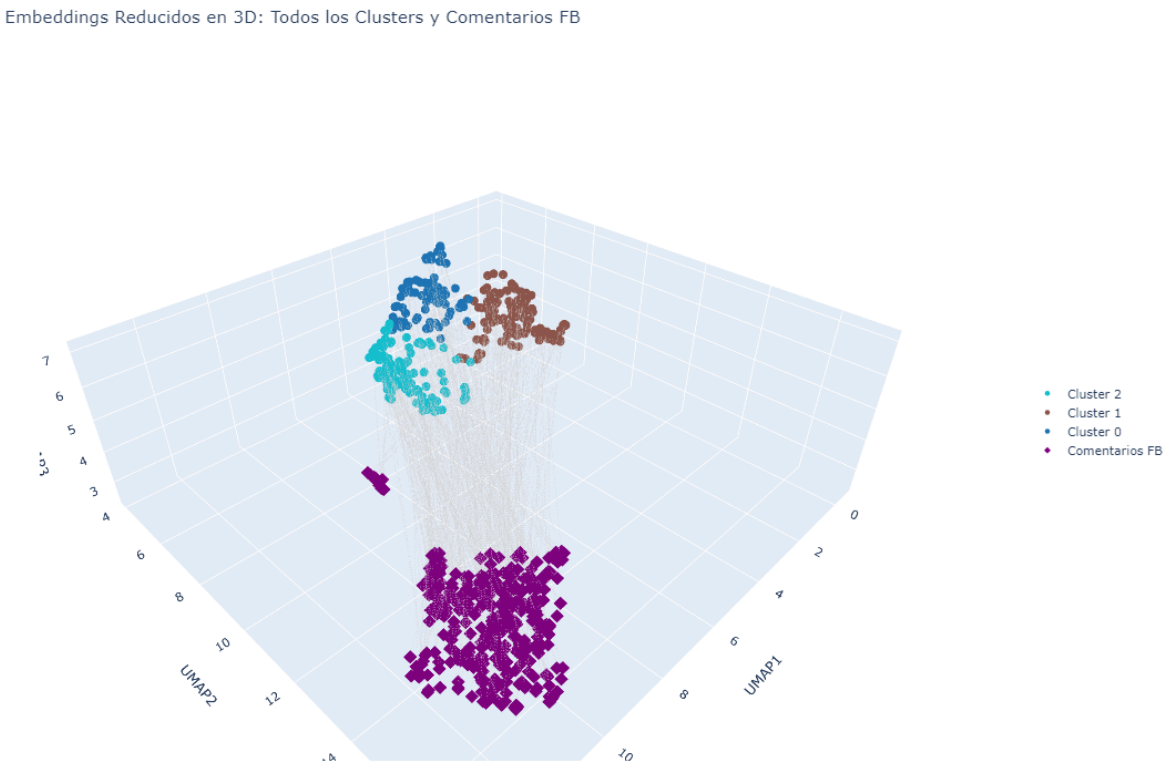
Al comparar el gráfico con el G.2.2.A y G.2.2.B, se observa que los *clusters* contienen artículos de todos los medios, cuya fecha de publicación pueden bien pertenecer al primer o segundo período que definimos de modo trivial. Por otra parte, hay una división no mencionada que separa al *set* de datos en medios locales y nacionales, accesible mediante el enlace del código. En el *plot* de los datos "taggeados" de este modo, también se aprecia una mezcla de ellos: los *cluster* que se muestran en G.3.1.A contienen puntos que pertenecen o bien a un conjunto, u a otro de la dicotomía nacional-local. Pues, la hipótesis (a) y la primera parte de la (b) se corroboran, dado que el conjunto total de los datos se encuentra en un espacio de representación relativamente reducido (a) y, (b) los agrupamiento conformados contienen artículos de distintos medios (locales y naciones), no separables por fecha (al menos en base a la división trivial de tiempo).

Si bien hay tres *clusters* diferenciados, no estamos seguros de los límites entre ellos: ¿Qué significa la separación? Se observa que están próximos unos a otros puntos pertenecientes a distintos *clusters*. Es decir, que en las inmediaciones de los bordes, que un

artículo pertenezca o no a un grupo, puede relacionarse más a la optimización del algoritmo *k-means* que a una correlación con criterios lingüísticos que funcionen en los textos.

Este es el motivo por el cual desarrollamos un herramienta exploratoria de los resultados. Después de ejecutar el código, los gráficos generados son interactivos. Si nos posicionamos con el cursor sobre cualquier punto se abre una pestaña con metadatos del artículo (G.3.1.A). El recuadro tiene una doble función. En primer lugar, ayuda a identificar la fecha, medio, título (mediante la *url*), etc., que rápidamente otorgan una representación del punto (vector). En segundo lugar, se muestra el índice del *set* de datos para una indagación más profunda. Si se ejecuta la sección de código para exploración de datos, al introducir el índice y presionar >>*enter*, se imprime por pantalla el contenido de la pestaña y el texto del artículo, separado en título, bajada, cuerpo del texto y lista de comentarios de *Facebook*. También, la *url* que se muestra es hipervínculo hacia el artículo en los portales de noticias.

A continuación mostramos el gráfico 3D de artículos y comentario para mostrar la utilidad de la interfaz exploratoria:



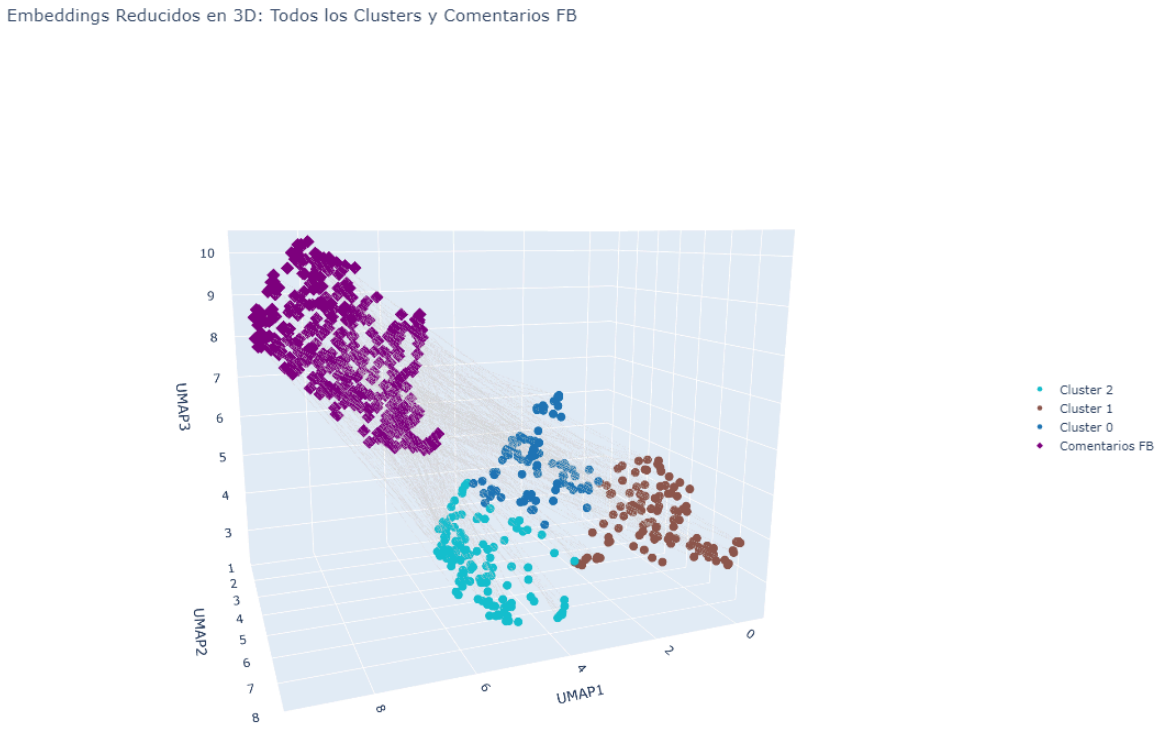
G.3.1.B

Entre los dos conjuntos macro (artículos y comentarios) hay un grupo de comentarios "a medio camino". Nos interesamos particularmente por el pequeño subconjunto. Mediante la indagación de la pestaña emergente en el gráfico, notamos que todos los comentarios

pertenecían al portal elciudadano. Luego, con el dato de índice de los artículos utilizamos la interfaz para explorar en profundidad los comentarios. El factor determinante para conformar el pequeño conjunto fue que había comentarios publicitarios repetidos en cada uno de ellos y en inglés. Pues, cometimos la omisión de sustraer estos comentarios, por lo que volvimos a procesar los datos y quitarlos del *set* de datos.

3.2. Representación de *clusters* con los comentarios de *Facebook*

La hipótesis (c) anticipa una separación geométrica entre el conjunto de artículos y comentarios. En el siguiente gráfico presentamos sólo aquellos artículos que poseen comentarios, justamente, para establecer la relación:

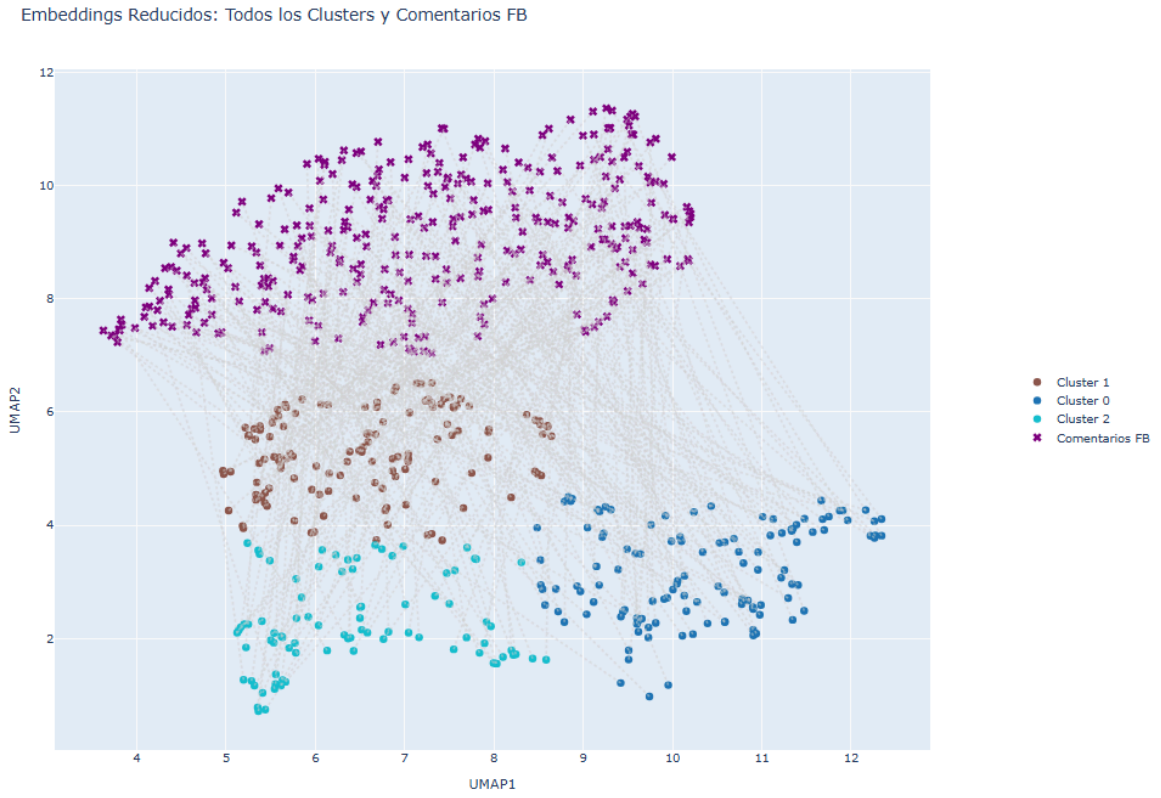


G.3.2.A

Las distancias permiten figurar que los artículos, por un lado, y los comentarios, por otro, son más cercanos entre sí, dentro de sus respectivos conjuntos, que en relación con el vínculo que los conecta. En correlación, pareciera haber atributos determinantes que distinguen las textualidades, donde el modo de escritura de los comentarios resulta más relevante para la conformación del conjunto que el referente del texto (supuestamente, es sobre el artículo que se habla en los comentarios). En todo caso, resulta evidente que la distancia geométrica entre los datos permite inferir criterios de inclusión y exclusión. Ahora

bien, ¿cuáles son? Consideramos que uno de ellos es el género discursivo de pertenencia: el de comentarios en redes sociales (en este caso *Facebook*) y el de noticias *web*.

No es novedad que los géneros separen textualidades, sino más bien lo contrario. Sin embargo, ha llamado nuestra atención la madeja de líneas que unen a cada artículo con sus comentarios. Esto se aprecia más en el gráfico 2D:



G.3.2.B

Tal vez, este aparente enredo refleja una correlación entre los mecanismos de transformación que operan en el tránsito de un género a otro, que deberemos objetivar en nuevas hipótesis mediante precisiones provisorias que los expliquen.

CONCLUSIÓN

El informe da cuenta del proceso de trabajo realizado durante la pasantía. Intentamos ponderar la claridad en la exposición. En este sentido, la profundidad teórica fue restringida exclusivamente para contextualizar y fundamentar el análisis. Consideramos que el ejercicio nos prepara para distintos escenarios debido a que las implementaciones y métricas serían aplicables a contextos diversos. Es susceptible adoptar un enfoque similar en agronomía y medicina, al vectorizar atributos de guisantes o patologías. No debemos perder de vista que

los resultados técnicos siempre deberán contrastarse con la visión disciplinar, la cual, será referencia ineludible para evaluar los resultados. El *feedback* nos obligará a profundizar en algunos aspectos, o en otros, reformular los planteos del problema.

Los objetivos y actividades de un informe son fundamentales ya que constituyen la estructura sobre la cual se organiza el texto. A través de las actividades logramos cumplir los objetivos. Sin embargo, consideramos fundamental el papel de las hipótesis. Si bien en principio un informe no requiere necesariamente la formulación explícita de hipótesis, deben elaborarse. Todo escrito de estas características parte, aunque sea de manera implícita, de una hipótesis que orienta su estructura y análisis[18]. Por esta razón, resulta adecuado esbozar hipótesis porque no solo refuerzan la argumentación, sino que también permiten interpretar el alcance de los resultados.

En relación con las hipótesis, corroboramos (a) y, en parte, (b) y (c). Decimos "en parte" porque, por un lado, en (b) separamos el conjunto de artículos en 3 *clusters*, lo que evidencia que el agrupamiento no coincide plenamente con el etiquetado trivial. Sin embargo, no identificamos estrategias enunciativas, subtemáticas recurrentes o isotopías en la construcción de sentido. Por otro lado, en (c), observamos una separación geométrica entre el agrupamiento de artículos y sus comentarios, que sugiere una correlación entre la distancia y las posibles transformaciones del discurso periodístico al de redes sociales. No obstante, aún no especificamos los mecanismos concretos de transformación, ni cómo se producen estos "saltos" entre géneros discursivos que abordan una misma temática. Cabe destacar que no avanzar en la confirmación o el rechazo de la totalidad de las hipótesis fue una decisión deliberada. La elección se fundamenta en tres aspectos. En primer lugar, profundizar hubiera excedido los límites del perfil técnico y el estado del informe preliminar (no olvidemos que esta es la primera presentación de análisis sistemático sobre el conjunto de datos). En segundo lugar, consideramos que la construcción de conocimiento no debe ser precipitada, y es preferible otorgar cierta temporalidad, aunque sea lógica antes que mediata, para revisitar los resultados con mayor perspectiva. En tercer lugar, el trabajo realizado hasta aquí se enriquecerá con las devoluciones de equipo del PID, que habilitarán un diálogo interdisciplinario para perfeccionar las hipótesis antes que apresurar su corroboración/rechazo. Por estas razones, proponemos una aproximación exploratoria a los datos mediante la interfaz generada en el código.

[18] En un escrito anterior, trabajamos la idea de hipótesis velada en la formulación de la situación problema de una investigación, la cual podría considerarse una proto-hipótesis entramada en dicha escena textual (Alomar, 2023).

Ahora bien, resulta necesario reflexionar de manera crítica sobre los resultados, particularmente, para diseñar nuevas estrategias de abordaje. Sabemos que una correlación no implica siempre causalidad, por lo que, puede que la división del *set* de datos en 3 *cluster* provenga de la optimización de distancia entre vectores hacia los centroides de *k-means*, pero que carezca de fundamentos que expliquen los conjuntos mediante hipótesis formuladas desde el análisis del discurso. Bien podrían ser una consecuencia de la estructura de los datos: por ejemplo, los comentarios de *Facebook* fueron *input* del algoritmo que los convierte a *embeddings* como un objeto lista, donde por artículo (registro del *dataset*) dos corchetes los encierran, separados por coma y, muchos de ellos, conteniendo emojis. Esto no ocurre con los vectores que representan los artículos surgidos de promediar el título, la bajada y el cuerpo del texto. Sí descartamos que la normalización haya compactado de modo desmesurado los datos porque, a pesar del planteo de la diferencia en la estructura de los datos, la distancia entre los comentarios y artículos persiste.

Pues, podemos sopesar nuevamente los resultados con el fin de desestimar la crítica planteada mediante un nuevo *approach* dentro del campo del PLN, la IA y la minería de datos. Nos referimos a aplicar sobre el mismo *dataset* otro algoritmo de aprendizaje no supervisado, *Latent Dirichlet Allocation* (LDA)[19]. A diferencia de *k-means*, que agrupa los datos optimizando distancias a centroides, LDA identifica tópicos en un corpus basándose en la frecuencia de palabras y probabilidades condicionales. Si la distribución de tópicos obtenida con LDA resultara congruente con los clusters de *k-means*, podríamos establecer una correspondencia en la construcción de conjuntos que justifique los agrupamientos. No obstante, este análisis plantea un problema metodológico nuevo, que exige un abordaje específico y quedará para un estudio futuro.

[19] En un artículo previo utilizamos LDA para el análisis de entrevistas relacionadas a una controversia ambiental (Preiti y Alomar, 2024). Salvando las diferencias con la temática actual, metodológicamente, la implementación de LDA no requeriría de mayores modificaciones para ser aplicada al *set* de datos construido para el caso de narcoterrorismo en la ciudad de Rosario.

BIBLIOGRAFÍA

Alomar, F. J. (2023). La hipótesis abductiva y el nivel supraunitario de la matriz de datos en la escena textual de la situación problemática . *Papeles De Trabajo. Centro De Estudios Interdisciplinarios En Etnolingüística Y Antropología Socio-Cultural*, (45). Recuperado de: https://papelesdetrabajo.unr.edu.ar/index.php/revista/article/view/227/191 (27/02/2025).

Géron, A. (2023). *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow* (3rd ed.). O'Reilly Media. Recuperado de: https://www.clc.hcmus.edu.vn/wp-content/uploads/2017/11/Hands_On_Machine_Learning_with_Scikit_Learn_and_TensorFlow.pdf (27/02/2025).

Mohammed J. Zaki, Wagner Meira (2020). *Data Mining and Machine Learning: Fundamental Concepts and Algorithms*. Jr. Cambridge University Press. 2nd Edition. Recuperado de: https://pzs.dstu.dp.ua/DataMining/bibl/mohammed_j_zaki_wagner_meira_jr_data_mining_and_analysis_fun.pdf (27/02/2025).

Preiti, F. J., & Alomar, F. J. (2024). Modelado de tópicos (LDA) aplicado a una controversia ambiental: Isotopías del discurso ambiental sobre el fuego en islas del Paraná, Argentina. *La Trama De La Comunicación*, *28*(01), 080–117. Recuperado de: https://latrama.unr.edu.ar/index.php/trama/article/view/846/561 (27/02/2025).

Raimondo Anselmino, N. (2011) O ocaso do modelo intencional: a noção de "estratégia discursiva" sob o olhar sócio-semiótico. *Semeiosis*, 1(2). Recuperado de: https://semeiosis.com.br/issues?issue=YEv5nsst2sPcL29M5KWG&article=EaGHosEF8L5 DVptfXaWC (23/08/2024).

Raimondo, F., Rostagno, L., & Cardoso, G. (2021). *Aplicación de técnicas de clustering para el estudio sociosemiótico sobre géneros periodísticos en fanpages de Clarín y La Nación*. Del prudente saber y el máximo posible de sabor, 23(14), 77-103. Recuperado de: https://pcient.uner.edu.ar/index.php/dps/article/view/1137/1294 (27/02/2025).

Wikipedia (2025). *Navaja de Ockham*. En Wikipedia, La enciclopedia libre. Recuperado de: https://es.wikipedia.org/wiki/Navaja_de_Ockham (27/02/2025).