Credibilidad como Dimensión de Calidad de Datos en Redes Sociales

Informe de Proyecto de Grado presentado al Tribunal

Evaluador como requisito de graduación de la carrera Ingeniería en Computación.

Montevideo - Facultad de Ingeniería

2019

Estudiantes:

- Luciana Martínez
- Claudia Iracce

Tutores:

- Adriana Marotta
- Sebastián García

Agradecimientos

Se agradece a Paula Scotti, Federico García, Daniel Mazzone, Carlos Álvarez y a IDatha, por el tiempo que dedicaron y toda la información brindada, fue de mucha utilidad.

Resumen

La masividad de información que circula hoy en día a través de Internet ha propiciado su recolección y su posterior utilización en diversas ramas. La extracción de información de valor desde medios sociales (Facebook, Twitter, TripAdvisor, Yelp, etc) se ha convertido en un desafío necesario para organizaciones públicas y privadas alrededor del mundo. La calidad de datos de fuentes internas, tradicionalmente se ha evaluado utilizando diferentes dimensiones de calidad. El crecimiento de nuevas fuentes de datos, externas a las organizaciones, ha generado la necesidad de investigación en la dimensión *credibilidad* para estas fuentes.

En este proyecto de grado se realiza un estado del arte vinculado a la dimensión de calidad *credibilidad* aplicado a datos provenientes de plataformas de social media. Se define un conjunto de dimensiones, factores y métricas, basados en la dimensión de credibilidad, en base a un relevamiento de bibliografía científica y entrevistas realizadas a usuarios expertos. Como prueba de concepto se desarrolla un prototipo de una herramienta web utilizando el lenguaje Python, que recibe como entrada fuentes de datos proporcionados por la red social Twitter y devuelve una evaluación de la calidad de estos. Para acotar el alcance se optó por utilizar tweets geolocalizados próximos al territorio Uruguayo y que estuvieran relacionados con temas relativos al Covid 19.

La herramienta se compone de dos interfaces web, denominadas *FakeBusters Server* y *FakeBusters Web.* La primera permite obtener usuarios y tweets directamente de Twitter, por medio de la API pública. Estos datos son guardados en dos bases de datos no relacionales generadas con tal fin, una que brinda la capacidad de búsquedas rápidas por texto (MongoDB), y una segunda orientada a grafos que posibilita mapear las relaciones entre

usuarios y tweets (Neo4j). La segunda interfaz implementa las métricas definidas distinguiendo entre métricas de usuario y métricas de clip. Utilizando los datos persistidos en *FakeBuster Server,* permite calcular el valor de las métricas específicas para un tweet definido; así como también calcular un valor de credibilidad global para el mismo tomando en cuenta varias de las métricas y asignándoles diferentes pesos. *FakeBuster Web* también permite realizar una búsqueda manual de usuarios y tweets con la finalidad de calcular sus métricas, a la vez que guarda un historial de los cálculos realizados.

Los resultados obtenidos en la prueba de concepto permiten afirmar que el enfoque aplicado es un enfoque válido, que posibilita representar las características más importantes a tener en cuenta para determinar la credibilidad de una publicación proveniente de redes sociales. Quedando para trabajos futuros la implementación de las métricas restantes, pudiendo anexarse otras redes sociales que aporten información complementaria, como ser LinkedIn. A su vez, resulta de suma importancia contar con un conjunto de datos lo más cercano a la realidad posible.

Palabras Claves:

Calidad de Datos, Dimensión Credibilidad, Métricas Credibilidad, Credibilidad de usuario Twitter, Credibilidad en Redes Sociales.

Índice

Agradecimientos ***3***

Resumen ***3***

Índice ***5***

Capítulo 1 Introducción ***9***

Capítulo 2 Conceptos Preliminares ***13***

1- Calidad de Datos *13*

1.1- Dimensiones y Métricas *13*

1.1.1- Dimensiones y Factores *14*

1.1.2- Métrica y Método de Medición *17*

1.1.3- Big Data y Dimensiones de Credibilidad *17*

2- Social Media *19*

2.1- Redes Sociales *20*

2.2- Herramientas Relacionadas con Social Media *23*

Capítulo 3 Estado del Arte ***27***

1- Perspectiva Académica *27*

1.1- Aplicaciones de Uso de los Datos de las Redes Sociales *27*

1.2- Dimensión de Calidad de Datos Credibilidad *29*

1.3- Métricas de Dimensiones de Credibilidad *35*

2- Perspectiva de la Industria *42*

2.1- Entrevistas a Usuarios de Datos de Redes Sociales *42*

Capítulo 4 Dimensiones de Calidad de Datos para Evaluar la Credibilidad ***47***

1- Definición de Dimensiones de Calidad *47*

1.1- Dimensiones Asociadas al Usuario *47*

1.2- Dimensiones Asociadas al Clip *50*

2- Definición de Métricas *52*

2.1- Constantes *52*

2.2- Metadatos *54*

2.3- Métricas Asociadas al Usuario *58*

2.4- Métricas Asociadas al Mensaje *65*

Capítulo 5 Implementación ***69***
1- Tecnología Utilizada *69*
1.1- Docker *69*
1.2- Python *69*
1.2.1- Flask *69*
1.3- Bases de Datos *69*
1.3.1- Base de Datos Relacionales vs Base de Datos Orientadas a Grafos *70*
1.3.2- Neo4j *71*
1.3.3- MongoDB *77*
2- Arquitectura *79*
3- Datos *80*
3.1- Obtención de Datos *81*
3.1.1- Tweets Streaming *81*
3.1.2- Tweets Asociados a un Usuario *81*
3.1.3- Tweets de Seguidores de un Usuario *81*
3.2- Persistencia de Datos *82*
4- Cálculo de Valores de Normalización *85*
5- Cálculo de Métricas *86*
6- Funcionalidades *94*
6.1- FakeBusters Server *94*
6.2- FakeBusters Web *94*

Capítulo 6 Caso de estudio ***97***
1- Obtención de Datos *97*
2- Definición de Parámetros y Pesos *98*
3- Elección de Datos de Estudio *99*
3.1- Usuario1 *100*
3.2- Usuario2 *102*
3.3- Usuario3 *105*
3.4- Usuario4 *107*
3.5- Usuario5 *109*
3.6- Usuario6 *111*
3.7- Usuario7 *113*
4- Valores de Normalización de las Distintas Métricas *116*

5- Cálculo de Métricas *116*
5.1- Followers_quantity *117*
5.2- Tweets_retweets_pub *118*
5.3- Favourite *119*
5.4- Antiquity *120*
5.5- Verified *122*
5.6- Commented *123*
5.7- Quote_count *124*
5.8- Retweets_count *126*
5.9- Favourite_count *127*
5.10- Followers_credibility *128*
5.11- Similar_words *130*
5.12- Same_location *131*
5.13- Reputation_path *132*
5.14- General_credibility *133*
6- Análisis de Resultados *135*

Capítulo 7 Conclusiones ***137***
1- Resultados Obtenidos *137*
2- Problemas Enfrentados *138*
3- Limitaciones *139*
4- Trabajo a Futuro *139*

Referencias Bibliográficas ***141***

Tabla de Términos ***149***

Anexo ***155***
1- Redes Sociales de Uso General *155*
2- Redes Sociales de Uso Específico *155*
2.1- Mensajería *155*
2.2- Medicina *157*
2.3- Recursos Humanos *158*
2.4- Microblogging *158*
2.5- Geolocalización *160*
2.6- Video *160*

2.7- Imágenes *160*
2.8- Música *161*
2.9- Presentaciones *161*
3- Herramientas *162*
3.1- ElasticSearch *162*
3.2- Recuperación de Información *164*
3.3- Herramientas para Manejo de BigData *165*
3.4- Herramientas para la Analítica de Datos Masivos *165*
4- Funcionalidades detalladas del prototipo *166*
4.1- FakeBusters Server *166*
4.2- FakeBusters Web *167*
4.2.1- Parámetros *167*
4.2.2- Historial *168*
4.2.3- Obtener Información Tweet *170*
4.2.4- Métrica de Usuarios y Clips *171*
4.2.5- Credibilidad General *176*

Capítulo 1 Introducción

Los volúmenes de datos generados por los usuarios en medios sociales se han incrementado drásticamente en los últimos años. Las organizaciones están utilizando estos datos con múltiples objetivos; desde analizar comentarios sobre productos en sitios de comercio electrónico, para diseñar nuevos productos a partir de las preferencias de los usuarios; hasta poder geolocalizar focos de infección de ciertas enfermedades a partir de publicaciones de usuarios en redes sociales. La extracción de información de valor desde medios sociales (Facebook, Twitter, TripAdvisor, Yelp, etc.) se ha convertido en un desafío necesario para organizaciones públicas y privadas alrededor del mundo. Debido a la característica abierta de las plataformas de medios sociales, a diferencia de las fuentes de datos internas, se hace fundamental comprender la relevancia y credibilidad de los datos. La calidad de datos de fuentes internas, tradicionalmente, se ha evaluado utilizando diferentes dimensiones de calidad.

El crecimiento de nuevas fuentes de datos, externas a las organizaciones, ha generado la necesidad de investigación en la dimensión *credibilidad*, y al mismo tiempo la industria está requiriendo soluciones sólidas al respecto. Buscan responder preguntas como: ¿Qué tanta credibilidad tiene una publicación de un usuario en redes sociales?, haciendo referencia a determinada empresa, producto o persona.

Este proyecto de grado persigue dos objetivos principales: (i) comprender el estado del arte vinculado a la dimensión de calidad *credibilidad* aplicado a datos provenientes de plataformas de social media, y (ii) definir un conjunto de factores, métricas y dimensiones basados en la dimensión de *credibilidad*. Así mismo, se busca desarrollar un prototipo de una herramienta para la evaluación de las dimensiones de calidad propuestas, que reciba como entrada fuentes de datos de medios sociales y devuelva la evaluación de la calidad de estas.

Las dimensiones y métricas obtenidas en el proyecto están basadas en el relevamiento de bibliografía científica y en entrevistas realizadas a usuarios expertos, lo cual le da una amplitud de visiones consideradas para las definiciones propuestas. El abordaje a la problemática planteada se realiza

desde el área de investigación calidad de datos. La propuesta está basada en dimensiones de calidad de datos con sus respectivos factores y métricas, basados en diferentes publicaciones científicas. Además, esta propuesta se desarrolla en forma coordinada, con el desarrollo de la tesis de maestría, en curso de uno de los tutores, Sebastián García, la cual está enfocada en la aplicación de calidad de datos en la credibilidad de publicaciones de redes sociales.

Se decidió evidenciar los resultados obtenidos llevándolo a la práctica por medio de un prototipo web basándose en datos de la red social Twitter, implementando las métricas viables y comparando resultados esperados con los recabados. Para acotar el alcance se optó por utilizar tweets geolocalizados próximos al territorio Uruguayo y que estuvieran relacionados con temas relativos al Covid 19. Los resultados obtenidos en la prueba de concepto permiten afirmar que el enfoque aplicado es un enfoque válido, que posibilita representar las características más importantes a tener en cuenta para determinar la credibilidad de una publicación proveniente de redes sociales.

Queda para trabajos futuros la implementación de las métricas restantes, pudiendo anexarse otras redes sociales que aporten información complementaria, como ser LinkedIn. A su vez, resulta de suma importancia contar con un conjunto de datos lo más cercano a la realidad posible.

En los siguientes capítulos se explica con detenimiento cada una de las etapas; comenzando el capítulo 2 con los conceptos preliminares, como ser cuál es la importancia de la calidad de datos, qué es una dimensión, sus distintos componentes: métricas, factores; los distintos métodos de medición, así como también se expresa qué se entiende por Big Data y se introduce las dimensiones de credibilidad. Le continua una sección que aborda la temática social media. En el capítulo 3 se realiza el estado del arte, distinguiendo entre la perspectiva académica y la perspectiva de la industria. El capítulo 4 contiene el análisis de la información, en él se fundan las definiciones de las dimensiones y métricas a utilizar en los capítulos restantes, discriminando las mismas entre usuario y clip; se toma como base la información descrita en el capítulo 3.

La implementación del prototipo se detalla en el capítulo 5, aquí se describen las diferentes tecnologías utilizadas (lenguaje, frameworks, bases de datos) junto al motivo de su elección, la arquitectura del sistema, la forma de obtener y persistir los datos, cómo se definen los valores de normalización para las distintas métricas, cómo se calcula cada una de ellas y por último las diferentes funcionalidades implementadas en el prototipo. En el capítulo 6 es donde se plantea un caso de estudio, comienza estableciendo cómo se obtienen los datos, es decir, que características deben cumplir los datos para ser agregados, los valores de parámetros y pesos de las distintas métricas, necesarios para la obtención del resultado de credibilidad. A su vez, se seleccionan y especifican los datos de estudio a utilizar, así como los valores de normalización de las métricas. El capítulo finaliza con el cálculo y posterior análisis de las métricas para cada uno de los datos de estudio seleccionados.

El capítulo 7 está dividido en 3 secciones donde se detallan las conclusiones obtenidas, los problemas enfrentados, las limitaciones y los trabajos a futuro. El siguiente capítulo contiene las referencias bibliográficas (Referencias Bibliográficas); a continuación, en el capítulo Tabla de Términos, se precisa una tabla con los términos claves utilizados a lo largo del informe. Por último, se agrega un capítulo de Anexos, en él se encuentran especificadas otras redes sociales que existen actualmente, así como otras herramientas investigadas.

Capítulo 2 Conceptos Preliminares

En esta sección se brindará una breve introducción a los conceptos necesarios para el desarrollo del trabajo, la intención es darle al lector un marco teórico para la comprensión de este.

1- Calidad de Datos

Nos encontramos en una era digital, estamos inmersos en una sociedad en la cual casi toda la información que se maneja se encuentra digitalizada, por ende, los datos físicos dejaron de ser relevantes.

Las diferentes organizaciones (gubernamentales, comerciales, ciudadanas, etc.) usan esta información para la toma de decisiones, planificación de estrategias, guía de diferentes procesos, así como también para relacionarse entre ellas; por este motivo los datos se han vuelto la materia prima más importante para las mismas. En este escenario la calidad de la información que se maneja se torna de una importancia determinante.

Dada la gran masividad de los datos se hace difícil clasificarlos, analizarlos y depurarlos para poder obtener datos de valor real; en este sentido la calidad de estos juega un papel decisivo.

Se manejan varias definiciones de calidad de datos, para este trabajo se va a usar la definida en el estándar ISO/IEC 25012:2008 [1]; el cual define a la calidad de los datos como el grado en el que las características de los datos satisfacen las necesidades declaradas e implícitas cuando son usadas bajo ciertas condiciones y proveen un modelo de calidad de datos general para persistir los datos en un formato estructurado dentro de un sistema informático.

1.1- Dimensiones y Métricas

La calidad es descrita por atributos que ayudan a calificar los datos, los cuales son denominados dimensiones, a su vez las dimensiones se pueden subclasificar en factores, los cuales son medidos con métricas; que necesitan modelos de medición para ser usadas. En la siguiente sección se ampliará sobre estos conceptos.

1.1.1- Dimensiones y Factores

Cada dimensión captura un aspecto distinto de la calidad de los datos (DQ). Las dimensiones se pueden clasificar y agrupar de diferentes maneras, en el estándar ISO/IEC 25012 [1], son agrupadas en tres grandes categorías: *Inherente*, *Dependiente* e *Inherente Dependientes*, abarcando esta última a las dimensiones que pertenecen a ambos grupos simultáneamente.

En la Tabla 1 podemos visualizar algunas dimensiones de calidad según estas categorías.

Clasificación Dimensiones Calidad		
Inherentes	**Dependientes del sistema**	
Correctness (*Exactitud*)	*Accessibility* (*Accesibilidad*)	*Availability* (*Disponibilidad*)
Completeness (*Completitud*)	*Compliance* (*Conformidad*)	*Portability* (*Portabilidad*)
Consistency (*Consistencia*)	*Confidentiality* (*Confidencialidad*)	*Recoverability* (*Recuperabilidad*)
Credibility (*Credibilidad*)	*Efficiency* (*Eficiencia*)	
Currentness (*Actualidad*)	Precision (*Precisión*)	
	Traceability (*Trazabilidad*)	
	Understandability (*Comprensibilidad*)	

Tabla 1. Clasificación dimensiones ISO

Inherente se refiere al grado con el que las características de calidad de los datos tienen el potencial esencial para satisfacer las necesidades establecidas y necesarias cuando los datos son utilizados bajo condiciones

específicas; por ejemplo, valores de dominios de datos y posibles restricciones, relaciones entre valores de los datos, metadatos.

Dependiente del Sistema se refiere al grado con el que la calidad de los datos es alcanzada y preservada a través de un sistema informático cuando los datos son utilizados bajo condiciones específicas.

A continuación (Tabla 2) se detallan las definiciones de la ISO de algunas de las dimensiones:

Característica de Dimensión de Calidad	Definición
Correctness (Exactitud)	El grado en el que los datos tienen atributos que representan correctamente el verdadero valor del atributo deseado de un concepto o evento es un contexto específico de uso.
Completeness (Completitud)	El grado en el que los datos asociados con una entidad tienen valores para todos los atributos esperados y las instancias relacionadas con la entidad en un contexto específico de uso.
Consistency (Consistencia)	El grado en que los datos tienen atributos libres de contradicción y son coherentes con otros datos en un contexto específico de uso.
Credibility (Credibilidad)	El grado en que los datos tienen atributos que los usuarios consideran verdaderos y creíbles en un contexto específico de uso.
Currentness (Actualidad)	El grado en que los datos tienen atributos que tienen la vigencia adecuada en un contexto específico de uso.
Accessibility (Accesibilidad)	El grado en que los datos pueden ser accedidos en un contexto específico de uso, particularmente por personas que necesitan tecnología de apoyo o configuraciones especiales debido a alguna discapacidad.
Compliance (Conformidad)	El grado en que los datos tienen atributos que se adhieren a las normas, convenciones o

	regulaciones vigentes y reglas similares relacionadas con los datos en un contexto específico de uso.
Confidentiality (Confidencialidad)	El grado en que los datos tienen atributos que aseguran que solo sea accesible e interpretable por los usuarios autorizados en un contexto específico de uso.
Efficiency (Eficiencia)	El grado en que los datos tienen atributos que pueden procesarse y proporcionar los niveles de rendimiento esperados al usar las cantidades y los tipos de recursos apropiados en un contexto de uso específico.
Precision (Precisión)	El grado en que los datos tienen atributos que son exactos o que proporcionan discriminación en un contexto específico de uso.
Traceability (Trazabilidad)	El grado en que los datos tienen atributos que proporcionan una pista de auditoría de acceso a los datos y de cualquier cambio realizado en los datos en un contexto específico de uso.
Understandability (Comprensibilidad)	El grado en que los datos tienen atributos que le permiten ser leídos e interpretados por los usuarios, y se expresan en los idiomas, símbolos y unidades apropiados en un contexto específico de uso.
Availability (Disponibilidad)	El grado en que los datos tienen atributos que le permiten ser leídos e interpretados por los usuarios, y se expresan en los idiomas, símbolos y unidades apropiados en un contexto específico de uso.
Portability (Portabilidad)	El grado en que los datos tienen atributos que le permiten ser instalado, reemplazado o movido de un sistema a otro preservando la calidad existente en un contexto específico de uso.
Recoverability (Recuperabilidad)	El grado en que los datos tienen atributos que le permiten mantener y preservar un nivel

	específico de operaciones y calidad, incluso en caso de falla, en un contexto específico de uso.

Tabla 2. Estándar ISO de Dimensiones de calidad de datos.

En este proyecto se analizará más a detalle la dimensión de *Credibilidad*, la cual refiere al grado en el que los datos tienen atributos que son considerados como verdaderos y creíbles para los usuarios en un contexto específico.

Cuando hablamos de un aspecto particular de una dimensión esto se denota como factor de calidad, es decir una dimensión es un conjunto de factores de calidad que tienen el mismo propósito.

1.1.2- Métrica y Método de Medición

Cuando hablamos de métrica nos referimos al instrumento utilizado para definir la forma de medir los factores de calidad; éstos pueden tener diferentes métricas asociadas.

Para determinar una métrica se debe definir su semántica; o sea, qué se está midiendo, las unidades de medición y la granularidad de la medida (pudiendo ser una celda, tupla o atributo si nos estamos refiriendo a un modelo relacional). Cada métrica debe tener uno o varios métodos de medición, los cuales están asociados a dónde se tomó la medida, qué datos están incluidos, el dispositivo de medición y la escala en la cual los resultados están reportados. Siendo un método de medición un proceso que implementa una métrica.

La implementación del método es dependiente de la aplicación y también de la estructura de la base de datos, por ejemplo, para medir el tiempo transcurrido desde la última actualización se puede usar un timestamps de la base de datos, acceder a sus logs de actualización, etc.

1.1.3- Big Data y Dimensiones de Credibilidad

Según Dumbill [2], Big Data es la información que excede la capacidad de procesamiento de bases de datos convencionales. Los datos son demasiado grandes, se mueven demasiado rápido o no se ajustan a las estructuras de

las arquitecturas de base de datos convencionales. Para obtener valor de estos datos, se debe elegir una forma alternativa de procesarlos.

El término Big Data es usado para identificar conjuntos de datos estructurados y no estructurados, que son imposibles de guardar y procesar usando las herramientas de software comunes, como base de datos relacionales.

Cada vez se gana más terreno en esta área, tanto en lo académico como en relación con lo empresarial, hoy en día el principal desafío a cumplir son las tres 3V:

- *Variety* (variedad): La cual se refiere a la heterogeneidad de la adquisición de los datos, de la representación y de su interpretación semántica.
- *Volume* (volumen): La cual refiere al tamaño de los datos, el cual aumenta a un ritmo acelerado hoy en día.
- *Velocity* (velocidad): Se refiere a la tasa de obtención de los datos y al tiempo de vida útil de los mismos.

A esto se agrega una cuarta V (4V) *Veracity*, la cual refiere al ruido y la alteración de los datos. Se debe verificar si los datos que se almacenan y se extraen están relacionados y/o son significativos al problema que se quiere analizar. *Veracity* refiere a la calidad de los datos en particular a la credibilidad de estos.

El tamaño de los datos no es el único aspecto que hacen al dato pertenecer a Big Data, si no su crecimiento continuo en el tiempo, que termina siendo el factor que aporta mayor valor al negocio.

Hay tres tipos principales de fuentes de datos que pueden ser consideradas como Big Data [1]:

- *Fuentes de información de origen humano (Twitter, Facebook, Google, etc.)*
- *Fuentes de información de procesos* (registros médicos, comercio electrónico, tarjetas de crédito, etc.)

- *Fuentes de información generadas por máquinas* (cámara web, sensores de tráfico, sensores de polución, GPS, etc).

En este proyecto nos vamos a centrar en los datos provenientes de la primera fuente de información (de origen humano). Para poder aprovechar al máximo las oportunidades que se derivan del uso de datos web es realmente importante tener una clasificación de los mismos, en esta categorización *trustworthiness* (veracidad) y *provenance* (procedencia) juegan un papel destacado [1].

T*rustworthiness* (veracidad) se puede caracterizar sobre la base de tres dimensiones: *believability* (credibilidad), *verifiability* (verificabilidad) y *reputation* (reputación), las cuales contienen diferentes métricas.

Provenance (procedencia) de un dato es un conjunto de metadatos que contienen descripciones de entidades y actividades involucradas en la producción y entrega del mismo. El uso principal de la procedencia está relacionado con entender de dónde provienen los datos, identificar la propiedad y los derechos del mismo, hacer juicios para determinar si es confiable o no, verificar que el proceso utilizado para obtener un resultado cumple con los requisitos estipulados y reproducirlo.

2- Social Media

Hoy en día las redes sociales juegan un papel muy importante como aglomerador de datos, aunque lamentablemente son un área muy vulnerable en cuanto a calidad de datos se refiere. Un claro ejemplo es lo fácil que puede ser aumentar la cantidad de seguidores en *Twitter* comprando o creando usuarios falsos, así como darle relevancia a una noticia, producto o marca de manera ficticia o hacer viral una noticia falsa, entre otros.

Por otro lado, no es nada sencillo detectar esas cuentas falsas, el propio *Twitter* en un intento de depurar su red bloqueó más de 80000 cuentas de las cuales muchas no eran falsas [3].

Los datos generados a partir de las redes sociales brindan a las organizaciones una forma eficaz de conectar con los clientes tanto actuales como futuros y de esta manera crear interés en sus productos. El interés genera *clicks*, *likes*, comentarios, opiniones y todo esto genera ingresos,

publicidad, incidencia en la opinión pública; que a su vez determinan el éxito o el fracaso de un negocio, su sostenibilidad, cambios políticos, etc. Para llevarlo a cabo de una manera satisfactoria es necesario que los datos obtenidos de las diferentes redes sociales sean creíbles, dado que luego se analizarán para aportar conocimiento y visión a la toma de decisiones. Contar con datos creíbles no sucede en la mayoría de los casos. Los datos poco fiables son tan peligrosos como los datos sucios y las consecuencias pueden ser devastadoras.

Una rama o disciplina que analiza estos datos es la *web analytics*, la cual se centra en el análisis de los datos que fluyen a través de sitios y páginas web [4]. No basta con obtener los datos de visitas, tasas de rebote y de salida, tasas de conversión, rankings de páginas, número de *me gusta* en *Facebook*, cantidad de tweets y retweets, entre otros datos; se trata de realmente extraer y dar valor cuantitativo a la información relevante para el negocio.

Una parte de la *web analytics* es la llamada *social analytics*, la misma permite integrar y analizar los datos no estructurados que se encuentran en el correo electrónico, mensajería instantánea, portales web, blogs y otros medios sociales, por medio de las herramientas de obtención de datos existentes [4].

En colaboración con la *web analytics* se usa *sentiment analysis*, lo cual pretende saber si una palabra o una opinión acerca del producto y/o servicio, de una marca es positiva o negativa para la empresa y si es capaz de incitar a la compra o generar impactos influyentes.

2.1- Redes Sociales

Existe una gran variedad de redes sociales y esto va en aumento; las mismas captan a miles de multifacéticos usuarios, abarcando diferentes edades, orígenes, intereses, etc.

Las empresas *We are Social* y *Hootsuite* en enero 2018 realizaron un reporte *Digital Global 2018* [5], donde se tomó como referencia los usuarios activos en un mes de las principales redes sociales y se elaboró un ranking mostrando las redes más usadas, diferenciando entre redes sociales y redes de mensajería. Se llegó a la conclusión de que *Facebook*, *Youtube* e

Instagram son las tres redes sociales más usadas, en tanto *Whatsapp*, *Messenger* y *Wechat* son las más usadas en mensajería (Figura 1).

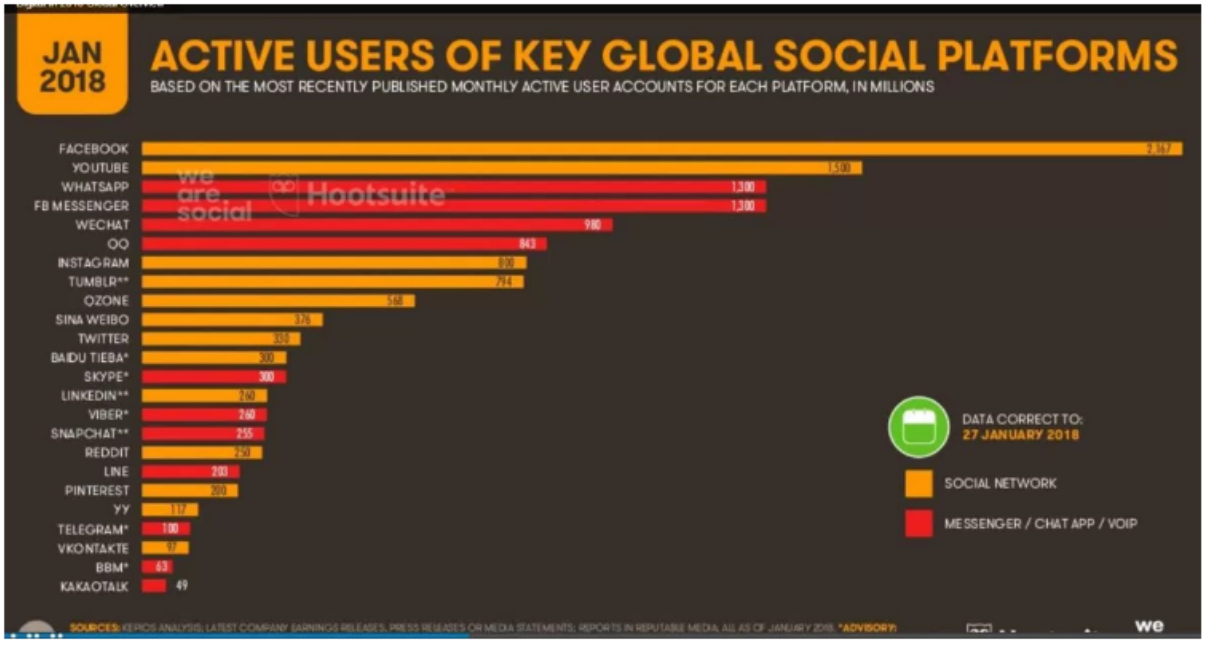

Figura 1. Reporte relevado por Digital Global 2018

Dada la dimensión social de las redes se opta por realizar una clasificación más específica, dividiéndolas en dos grandes grupos de acuerdo con su uso, de uso general y de uso específico (Figura 2).

Las de uso general no tienen una temática definida, están dirigidas a un público genérico y no tienen un propósito concreto; como función principal se puede destacar el relacionar usuarios por medio de las herramientas que cada una brinda. La mayoría cuenta con perfil de usuario y permite compartir contenidos.

Por otro lado, se encuentran las redes de uso específico, las cuales surgen para brindar un espacio de intercambio común a los usuarios. Dado que se está categorizando por redes de uso específico, su taxonomía podría ser tan diversa como los asuntos que tratan, es decir, se podría llegar a crear una subcategoría por cada tema en el que se basan; por lo que se decidió subdividir *por temática*, donde se encuentran las de mensajería, medicina,

microblogging y geolocalización y *por contenido compartido*, categorizando estas últimas en video, fotos, música y presentación.

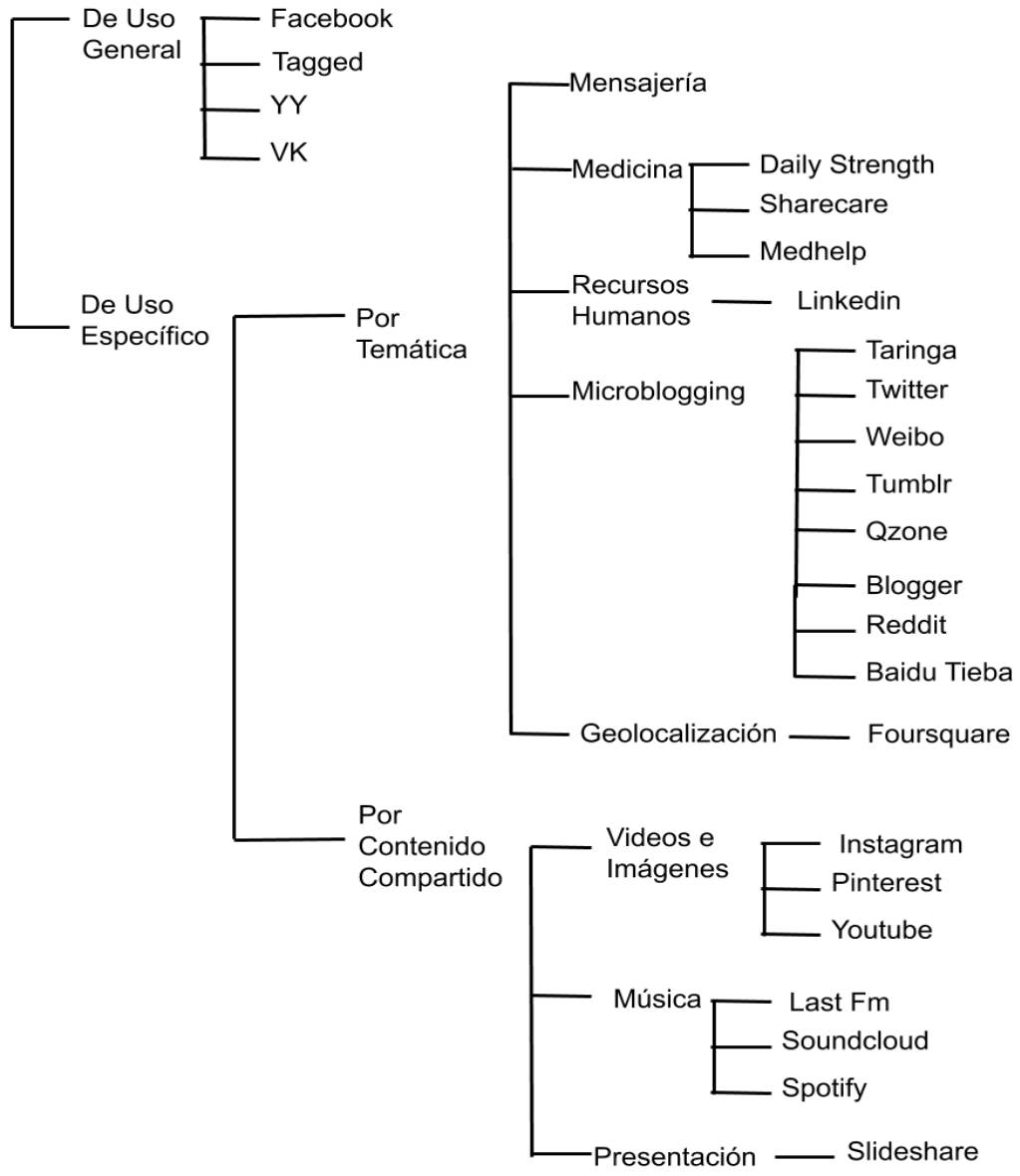

Figura 2. Taxonomía Redes Sociales

2.2- Herramientas Relacionadas con Social Media

Acompañando la gran variedad de redes sociales y usuarios que las utilizan, han aparecido diversidad de herramientas que administran, analizan y aprovechan las diferentes funcionalidades y datos que aportan las redes sociales. Al igual que para las redes sociales, se hace una clasificación de las herramientas encontradas que al parecer son interesantes (Figura 3).

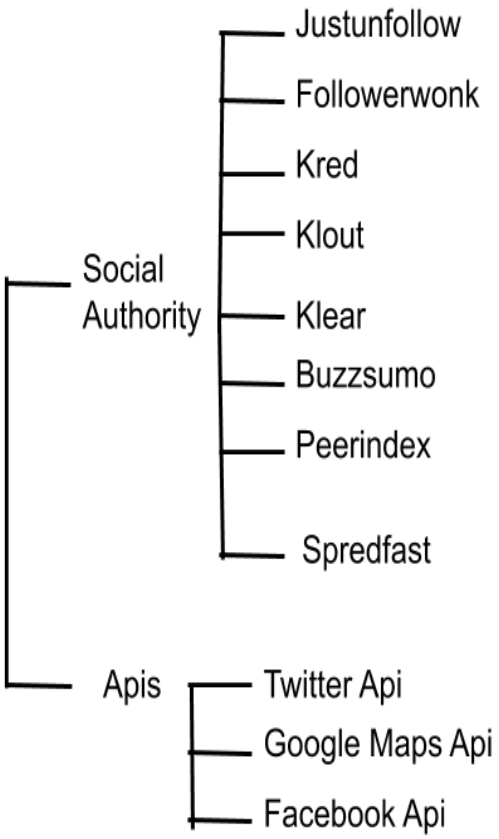

Figura 3. Taxonomía Herramientas Sociales

Podemos clasificar las herramientas en dos grupos, por un lado, se tienen las *APIS* (interfaz de programación de aplicaciones) dentro de las cuales se encuentran *Twitter*, *Facebook*, *Google Maps*; éstas permiten obtener

información valiosa de sus respectivas redes, generalmente de forma accesible, permitiendo un mejor análisis de estas.

Por otro lado, se tiene *Social Authority* (Autoridad Social), donde el valor real de las redes sociales proviene de aprovechar las oportunidades que brinda el interactuar o relacionarse con los demás. Cuanto más se aporta, más crece el capital social y más personas confían. La autoridad social se obtiene participando regularmente en la web y aportando información valiosa con un enfoque honesto y genuino.

Antes de las redes sociales, la reputación de una persona se definía por *a quién conocías* y *qué sabías*, ahora las mismas han introducido dos nuevos componentes: *quién te conoce* y *por qué te conocen*.

Existe una gran variedad de herramientas que son usadas para determinar la *Social Authority* de los usuarios. A continuación, se describen las que resultan más interesantes a nivel del proyecto, dejando en el anexo información sobre el resto de las herramientas que mencionamos:

Klout [4]

Klout es una herramienta gratuita, la cual dejó de existir en el 2018. La herramienta mide la influencia social que tiene una persona a través de las redes sociales a las cuales pertenece. Esta influencia se ve reflejada a través de un puntaje que es asignado por *Klout* y se basa en diversos factores o *señales* que son medidos por esta aplicación, entre los que se considera el número de seguidores que dicha persona tiene, la calidad de sus interacciones con otros y lo popular que llega a ser el contenido que comparte.

Para conocer este puntaje, el sitio web de *Klout* permite crear un perfil para cada uno de sus usuarios. En este perfil es posible ver el puntaje y conocer su evolución día a día. Además, es posible ver otro tipo de estadísticas.

El puntaje es asignado en base al número de seguidores en *Twitter*, los suscriptores en *Facebook* que se tiene, las recomendaciones profesionales de una persona en su perfil en *LinkedIn*, los consejos compartidos a través de *Foursquare*, entre otras características.

Permite medir la influencia de una persona o marca, a través del rastreo de los diferentes perfiles sociales. A partir de ello se obtiene *true reach* (alcance real), es decir, las personas reales a las que llega cada publicación, eliminando robots, personas con poca actividad o spam. También permite obtener *amplification* (amplificación), que mide la probabilidad que tiene un contenido de ser comentado o compartido y *network score* (puntuación en la red), que determina la influencia de las personas que responden a las publicaciones.

JustUnfollow [6]

Es una poderosa herramienta para utilizar en los perfiles de *Twitter*. Además de saber quién ha dejado de seguir una cuenta o quiénes son los usuarios inactivos, también permite que la cuenta sea promocionada para conseguir más adeptos, así como copiar los seguidores de otras cuentas, especialmente interesante si se busca encontrar personas con un perfil determinado.

Spredfast [7]

Esta herramienta está enfocada a quienes busquen una solución orientada a la analítica. Permite gestionar y evaluar los datos que se recogen tanto de *Facebook* como de *Twitter* y *Flicker*. De este modo se pueden obtener datos como a cuántas personas se ha llegado o si el público objetivo está siendo alcanzado por el contenido. Los datos se devuelven en gráficos, permitiendo también comparar otras campañas, entre otras posibilidades.

Twitter API

La *API* de *Twitter* ofrece una gran variedad de funcionalidades [8], las cuales se dividen en cinco grupos principales [9]:

- Cuentas y usuarios

  Permite a los desarrolladores administrar de forma programática el perfil y la configuración de una cuenta, silenciar o bloquear usuarios, administrar usuarios y seguidores, solicitar información sobre actividad de una cuenta, autorizar, etc.

- Tweets y respuestas

Los tweets y las respuestas públicas están a disposición de los desarrolladores, y se permite publicar tweets a través de la *API*. Se puede acceder a los tweets al buscar palabras clave específicas o solicitar una muestra de tweets de cuentas específicas.

- Mensajes directos

  Brinda acceso a las conversaciones por DM (mensaje directo) de usuarios que han otorgado permiso de forma explícita a una aplicación específica.

- Anuncios

  Ofrece un conjunto de funcionalidades para permitirles a los desarrolladores, como *Sprinklr*, ayudar a las empresas a crear y administrar automáticamente campañas de anuncios en *Twitter*. Los desarrolladores pueden usar tweets públicos para identificar temas e intereses, y así brindar a las empresas herramientas para ejecutar campañas publicitarias a fin de alcanzar las diversas audiencias en *Twitter*.

- Herramientas y SDK del editor

  Se proporcionan herramientas para desarrolladores y editores de software que permiten integrar las cronologías de *Twitter*, el botón para compartir y otros contenidos de *Twitter* en las páginas web. Estas herramientas permiten a las marcas incluir conversaciones públicas en vivo de *Twitter* en su experiencia web, de forma de que los clientes puedan compartir fácilmente información y artículos de sus sitios.

Capítulo 3 Estado del Arte

En esta sección se realiza una revisión de antecedentes relacionados con la temática de Credibilidad en Redes Sociales. Dividimos la misma en dos partes; perspectiva académica, en la cual se expone un estado del arte basado en artículos académicos, perspectiva de la industria, en la cual realizamos una revisión basada en entrevistas y tecnologías existentes.

1- Perspectiva Académica

El motivo de esta sección es generar un marco teórico y realizar una revisión de los trabajos realizados sobre la temática desde una perspectiva académica. Se compone de trabajos realizados utilizando redes sociales, calidad de datos, credibilidad y métricas de la misma.

1.1- Aplicaciones de Uso de los Datos de las Redes Sociales

Debido a la gran cantidad de usuarios, el alcance y el caudal de información que se maneja en las redes sociales, han surgido muchas aplicaciones que usan como fuente esa información. Se están explotando los datos obtenidos a través de las redes de diversas formas, dándole un sin fin de utilidades, aplicándolos a la salud, el medio ambiente, la prevención de desastres naturales, en la política, entre otros.

Twitter es una de las redes más usadas para este fin, la misma permite aprovechar todo su potencial mediante su API. Los usuarios discuten y hablan sobre los temas más diversos, incluidas sus condiciones de salud.

Una aplicación en este sentido es el trabajo *Vigilancia Activa para el Dengue* [10], en el cual se analiza cómo la epidemia del Dengue se ve reflejada en *Twitter* y cómo esa información puede ser usada para detectar brotes. Se propone un método de vigilancia activo basado en cuatro dimensiones: *volume* (volumen), *location* (locación), *time* (tiempo) y *public perception* (percepción pública). En una primera instancia se analiza la dimensión de percepción pública llevando a cabo un análisis de sentimientos, el cual permite filtrar el contenido que no es relevante para el caso planteado. Luego se verifica la alta correlación entre el número de casos reportados por estadísticas oficiales y el número de tweets posteados durante el mismo

período. Finalmente se usa un enfoque de clustering para explotar la dimensión espacio-temporal.

Dado que la transmisión del dengue es estacionaria y se incrementa durante la estación húmeda, el sistema de vigilancia debe ser capaz de detectar estos eventos para descubrir los brotes. La dimensión *volume* representa la cantidad de tweets que mencionan la palabra *dengue* y puede ser usada para aproximar la proporción/ritmo de incidencia del dengue. La dimensión *location* es la información geográfica asociada a los tweets que mencionan la palabra *dengue*. La dimensión *time* refiere a cuando estos tweets fueron posteados. La dimensión *public perception* es la percepción/sentimiento sobre la epidemia del dengue en general.

Twitter también se puede usar para inferir comportamientos sociales, por ejemplo, a partir del tráfico de logs de *Twitter* se pudo saber qué hicieron las personas en la web durante el terremoto ocurrido en Japón en el año 2011 [11].

Las redes sociales fueron usadas de manera efectiva, durante y después del mismo, ya sea para obtener información de lo que estaba ocurriendo o comunicarse con amigos y familiares. Las infraestructuras de red fueron dañadas y muchos usuarios no fueron capaces de conectarse a *Internet* en períodos cercanos al incidente; por lo cual fue posible observar que les pasó a los usuarios de la web en relación con el terremoto analizando tweets. Se usó el conjunto de tweets de usuarios japoneses como datos de entrada, en el cual se definieron usuarios frecuentes de *Twitter* como los que postean más de una cierta cantidad de tweets por día. Se analizó la cantidad de esos tweets agrupados por hora y se los normalizó para cierto período de tiempo. La frecuencia de tweets fue analizada comparándola con patrones de frecuencia en una situación normal y luego del terremoto. También se comparó por localidad, usando la ubicación de los tweets y diferenciando por medio de comunicación usado (smartphones, teléfonos, pc). Se concluyó que se mantiene el mismo número de tweets y que los mismos siguen un patrón cíclico. La mayoría de los usuarios no tuvo problema en postear durante el terremoto exceptuando las zonas altamente afectadas las cuales no recuperaron su frecuencia por varios días.

En el trabajo *Earthquake shakes Twitter users: real-time event detection by social sensors* [12], los usuarios de Twitter fueron usados como sensores para crear un mecanismo para la detección de terremotos. Se creó un modelo espacio-temporal probabilístico el cual encuentra el centro del terremoto y su trayectoria. El enfoque fue capaz de enviar alertas más rápido que las agencias meteorológicas.

Un uso cada vez más común de *Twitter* es en el área de la política. Tumasjan mostró en *Predicting elections with Twitter: What 140 characters reveal about political sentiment* [13], que las opiniones identificadas en los tweets relacionados con las elecciones federales alemanas son capaces de reflejar el sentimiento político de la gente en la realidad, esto permite usar a *Twitter* como predictor de acontecimientos políticos.

En el ámbito de la salud [14], el rápido crecimiento de información electrónica referida a la misma y la capacidad de procesar grandes volúmenes de esos datos automáticamente (usando procesamiento de lenguaje natural y algoritmos de aprendizaje automático), abrieron nuevas oportunidades a la farmacovigilancia, entendiéndose ésta como la prevención, evaluación y detección de los efectos adversos en los medicamentos. Por medio de redes sociales como ser *DailyStrength* y *MedHelp*, que tienen como objetivo principal que los usuarios se brinden apoyo mutuamente y compartan sus experiencias con los tratamientos, se logró tener un acercamiento a la detección y extracción de ADRs (reacciones adversas) a distintos medicamentos. El hecho de que es una fuente directa de experiencias personales de los usuarios lo hace a su vez una fuente lucrativa.

Las desventajas encontradas al usar contenido de las redes sociales pueden incluir cuestiones con respecto a la credibilidad, singularidades, frecuencia y prominencia de los datos.

1.2- Dimensión de Calidad de Datos *Credibilidad*

Con la evolución de la tecnología, tanto los individuos como las organizaciones obtienen y procesan datos de diferentes fuentes, por tal motivo es crítico asegurar la integridad de los datos para la toma de decisiones efectivas. La disponibilidad de datos completos hace posible obtener conocimiento más certero y exacto; y con ello colaborar a una toma

de decisiones más informada. Sin embargo, la dependencia de los datos en los procesos de toma de decisiones requiere que los mismos sean de buena calidad y confiables [15].

A partir de ahora se utilizará el término *Credibility* y *Believability* haciendo referencia al mismo concepto.

Según Wang y Strong[16] *believability* (credibilidad) es la medida en que los datos son aceptados o considerados como verdaderos, reales y creíbles.

Según C. Batini y M. Scannapieco credibility [1] también puede ser definida como la medida subjetiva de la creencia de un usuario de que el dato es *verdadero*.

Según Fogg, B.J. & Tseng *credibility* [17] es una cualidad percibida, no reside en un objeto, una persona o una pieza de información. La percepción de *credibility* resulta de la evaluación de múltiples subdimensiones simultáneamente; la gran mayoría de los investigadores identifican dos componentes clave de *credibility*: *trustworthiness* (veracidad) y *expertise* (experiencia).

Trustworthiness es definido a través de los términos *bien intencionado*, *veraz*, *imparcial* y *pronto*. *Expertise*, se define por términos tales como *experto*, *experimentado*, *competente*; por ende, la dimensión de experiencia de la credibilidad captura el conocimiento y habilidades adquiridas.

Para Y.L. Simmhan, B. Plale y D. Gannon [18] *credibility* del valor de los datos depende de su origen (fuente) y posteriormente de su historial de procesamiento. En otras palabras, depende de la procedencia del dato, definida como la información que ayuda a determinar el historial de derivación de un producto de datos, comenzando desde su fuente original.

Y. Lee, L. Pipino, J. Funk y R. Wang visualizan *credibility* [19] como la composición de tres subdimensiones: *of source* (de fuente), *compared to internal commonsense standard* (de sentido común interno), *based on temporality of data* (basada en la temporalidad del dato.)

N. Prat y S. E. Madnick también descomponen *credibility* en tres subdimensiones [20]: *trustworthiness* (veracidad), hasta qué punto el valor de un dato proviene de fuentes confiables; *reasonableness* (razonabilidad), hasta

qué punto el valor de un dato es probable; y *temporality* (temporalidad), basándose en el origen y el posterior historial de procesamiento de los datos. Presentan un modelo de procedencia y un enfoque basado en metadatos de procedencia; el mismo está estructurado en tres bloques: definición de métricas para evaluar la credibilidad de las fuentes de datos, definición de métricas para evaluar la credibilidad de los datos resultantes de un proceso ejecutado y evaluación de credibilidad basada en todas las fuentes y el historial de procesamiento de datos.

La opinión de los usuarios sobre qué elementos hacen que un sitio sea creíble es relevante para el proyecto. En el artículo *¿What Makes Web Sites Credible? A Report on a Large Quantitative Study* [21] se aborda este tema donde se considera que una persona hace una evaluación de confiabilidad y experiencia para llegar a una evaluación general de credibilidad, manejando los mismos conceptos definidos por Fogg, B.J. & Tseng para estos términos. En el mismo se concluyó que los cinco factores que aumentan la credibilidad son real-world feel (sentimiento del mundo real); pruebas de la existencia real de la organización, como ser fotos, dirección física, *ease of use* (la facilidad de uso); si el usuario se siente cómodo con la usabilidad de un sitio lo considera más creíble, *expertise* (experiencia); la trayectoria de la organización aporta credibilidad a la misma, *trustworthiness* (confiabilidad); de donde se obtiene la información, fuentes, materiales, *tailoring* (adaptabilidad); adaptar un sitio a la experiencia del usuario conduce a un aumento de las percepciones de credibilidad web.

Minjeong Kang [22] realizó un estudio sobre la medida de la credibilidad de un blog basándose en sus propios usuarios. En este estudio, los participantes indicaron atributos específicos tales como apasionado, transparente, influyente, auténtico, perspicaz, o centrado; a la vez enfatiza criterios de credibilidad comunes, tales como conocimiento, precisión, equidad o consistencia. De acuerdo con los resultados obtenidos en el estudio, authority (autoridad o influencia) y *reliability* (fiabilidad) del blogger son aspectos importantes en el factor de credibilidad del mismo, y *accuracy* (precisión) y *focus* (enfoque) (como el lugar de intereses personales y experiencia) son claves indicadores de la credibilidad del contenido del blog. El investigador también encontró que *authority* y *reliability* están fuertemente asociados entre

sí. Los resultados de este estudio indican que los atributos *authentic* (auténtico), *insightful* (perspicaz) e *informative* (informativo) son indicadores aproximados, mientras que los atributos *focused* (enfocado), *consistent* (consistente) y *fair* (justo) están fuertemente asociados a la credibilidad. Varios encuestados indicaron en sus respuestas abiertas que los aspectos estéticos de los blogs o el diseño profesional de blogs a menudo se convierten en pista útil para decidir sus primeras impresiones de blogs y la base de sus juicios de credibilidad.

Chenyun Dai, Dan Lin, Elisa Bertino y Murat Kantarcioglu [23] proponen un modelo de verdad basado en la procedencia de los datos, respondiendo a preguntas como ¿De dónde vienen los datos? ¿Cuán fidedigna es la fuente de datos original? ¿Quién maneja los datos? ¿Son los manejadores de los datos confiables? Para tratar de responder a estas preguntas propusieron un modelo que estima el nivel *trustworthiness* (veracidad) tanto de los datos como de los proveedores de los mismos asignándoles valores de verdad. Esos valores de verdad representan una información clave en la cual los usuarios de los datos pueden basarse para decidir si usar el dato y con qué propósito. Se toma en cuenta *data similarity* (similitud de los datos), *data conflict* (conflicto de datos), *path similarity* (similitud de ruta) y *data deduction* (deducción de datos). Además de *data similarity* y *data conflict*, la forma en que se recopilan los datos se considera también un factor importante a la hora de determinar la confiabilidad de los datos; por ejemplo, si varias fuentes independientes proporcionan los mismos datos, es más probable que dichos datos sean confiables. También se observó que es probable que los datos sean confiables si son proporcionados por proveedores de datos confiables, y un proveedor de datos es confiable si la mayoría de los datos que proporciona son confiables. Debido a tal interdependencia entre los proveedores de datos y los datos, desarrolló un procedimiento iterativo para calcular los puntajes de confianza. Para comenzar el cálculo, primero se le asigna a cada proveedor de datos un puntaje de confianza inicial que puede obtenerse consultando la información disponible sobre los proveedores de datos. En cada iteración, se calcula la confiabilidad de los datos en función de los efectos combinados de los cuatro aspectos mencionados anteriormente y se recalcula la confiabilidad del proveedor de datos al usar los puntajes de confianza de los datos que proporciona. Cuando una etapa estable se

alcanza, es decir, cuando los cambios en los puntajes de confianza son insignificantes, el proceso de cálculo de confianza se detiene.

C. Batini y M. Scannapieco [1] relacionan dos nuevas dimensiones con el concepto de credibilidad: *reputation* (reputación) y *verifiability* (verificabilidad).

Se describe *verifiability* como el grado y facilidad con que se puede verificar la corrección de información [78]. Se utiliza el término *verifiability* como medio que se proporciona a un consumidor, para examinar los datos para la corrección. Sin estos medios, la garantía de la exactitud de los datos provendría de la confianza del consumidor en esa fuente.

V*erifiability* es una dimensión importante cuando los datos son provistos por fuentes con baja credibilidad o reputación. Esta dimensión permite decidir si aceptar la información provista por la misma o no.

R*eputation* es un enjuiciamiento hecho por el usuario para determinar la integridad de una fuente. Puede ser asociada con el proveedor de los datos, una persona, organización, grupo de personas o puede ser una característica del conjunto de datos.

A continuación (Tabla 3) se resumen las diferentes características de *believability* descritas por las distintas fuentes:

Propuestas	Subdimensiones	Características Credibilidad
Wang y Strong [16]		- Medida en que los datos son aceptados o considerados como verdaderos, reales y creíbles.
C. Batini y M. Scannapieco [1]	*Reputation* (reputación), *Verifiability* (verificabilidad)	- Medida subjetiva de la creencia de un usuario de que el dato es *verdadero*. - Relaciona dos nuevas dimensiones con el concepto de credibilidad *reputation* (reputación) y *verifiability* (verificabilidad). Se describe *verifiability* como el *grado y facilidad con que se puede verificar la corrección de información*. *Reputation* es un enjuiciamiento hecho por el usuario para determinar la integridad de una fuente. Puede ser asociada con el

		proveedor de los datos, una persona, organización, grupo de personas o puede ser una característica del conjunto de datos.
Fogg, B.J. y Tseng [17]	*Trustworthiness (*veracidad*),* *Expertise (*experiencia*)*	- Es una cualidad percibida, no reside en un objeto, una persona o una pieza de información.
Y.L. Simmhan, B. Plale y D. Gannon [18]	*Of source (*de fuente*)*	- Depende de su origen (fuente) y posteriormente de su historial de procesamiento.
Lee, L. Pipino, J. Funk y R. Wang [19]	*Of source (*de fuente*),* *Compared to internal commonsense standard (*de sentido común interno*),* *Based on temporality of data (*basada en la temporalidad del dato*)*	- Es la composición de tres subdimensiones: *of source* (de fuente), *compared to internal commonsense standard* (de sentido común interno), *based on temporality of data* (basada en la temporalidad del dato).
N. Prat y S. E. Madnick [20]	*Trustworthiness (*veracidad*),* *Reasonableness (*razonabilidad*),* *Temporality (*temporalidad*)*	- Es la composición de tres subdimensiones: *trustworthiness* (veracidad), *reasonableness* (razonabilidad) y *temporality* (temporalidad), basándose en el origen y el posterior historial de procesamiento de los datos.
BJ Fogg, Jonathan Marshall, Othman Laraki, Alex Osipovich, Chris Varma, Nicholas Fang, Jyoti Paul, Akshay Rangnekar, John Shon, Preeti Swani, Marissa Treinen [21]	*Real-World Feel* (sentimiento del mundo real)*,* *Ease of Use* (la facilidad de uso)*,* *Expertise (*experiencia)*,* *Trustworthiness (*confiabilidad)*,* *Tailoring*	Factores que aumentan la credibilidad son -*Real-World Feel* (sentimiento del mundo real) pruebas de la existencia real de la organización, como ser fotos, dirección física, etc. -*Ease of Use* (la facilidad de uso), si el usuario se siente cómodo con la usabilidad de un sitio lo considera más creíble. -*Expertise (*experiencia), la trayectoria de la organización aporta credibilidad a la misma.

	(adaptabilidad)	-*Trustworthiness (*confiabilidad), de donde se obtiene la información, fuentes, materiales. - *Tailoring* (adaptabilidad), adaptar un sitio a la experiencia del usuario conduce a un aumento de las percepciones de credibilidad web.
Minjeong Kang [22]	*Authority (*autoridad, influencia*),* *Reliability* (*fiabilidad*), *Accuracy* (*precisión*), *Focus* (enfoque), *Authentic* (auténtico), *Insightful* (perspicaz), *Informative* (informativo), *Focused* (enfocado), *Consistent* (consistente), *Fair* (justo)	- *Authority* (autoridad o influencia) y *reliability* (fiabilidad) del blogger son aspectos importantes en el factor de credibilidad del mismo, y *accuracy* (precisión) y *focus* (enfoque) son claves indicadores de la credibilidad del contenido del blog. - Los resultados de este estudio indican que los atributos *authentic* (auténtico), *insightful* (perspicaz) e *informative* (informativo) son indicadores aproximados, mientras que los atributos *focused* (enfocado), *consistent* (consistente) y *fair* (justo) están fuertemente asociados a la credibilidad.
Chenyun Dai, Dan Lin, Elisa Bertino y Murat Kantarcioglu [23]	Trustworthiness *(veracidad)*	- La forma en que se recopilan los datos se considera también un factor importante a la hora de determinar la confiabilidad de los datos; por ejemplo, si varias fuentes independientes proporcionan los mismos datos, es más probable que dichos datos sean confiables. - Es probable que los datos sean confiables si son proporcionados por proveedores de datos confiables, y un proveedor de datos es confiable si la mayoría de los datos que proporciona son confiables.

Tabla 3. Características de Believability (Credibilidad).

1.3- Métricas de Dimensiones de Credibilidad

Como mencionamos anteriormente C. Batini y M. Scannapieco [1], definen *believability* como la medida subjetiva de la creencia de que los datos son

verdaderos. Las siguientes son distintas estrategias posibles para medir la credibilidad:

- Calcula la veracidad de las declaraciones RDF (Resource Description Framework) basadas en la procedencia de la información y en la opinión de otros consumidores y le aplica una función de verdad que asigna un valor que se encuentra en el intervalo [-1, 1], donde 1 es la verdad absoluta, -1 es la absoluta incredulidad y 0 la falta de ambas. Las funciones de verdad están basadas en clasificaciones de los usuarios y en métodos basados en la procedencia o la opinión.
- Calcula la veracidad de una entidad, de un objeto o de un recurso: un tercero confiable brinda una medida de verdad objetiva para cada entidad, a través de información como ser cantidad de citas o reputación global. Una vez que cada entidad tiene su valor de verdad asignado, es posible realizar inferencias de verdad en las nuevas entidades que aparezcan.
- Se calcula la verdad entre dos entidades usando una combinación de un algoritmo de propagación que utiliza técnicas estadísticas sobre dos entidades a través de una trayectoria y un algoritmo de agregación basado en un mecanismo de ponderación para calcular el valor agregado sobre todas las trayectorias.
- Se obtiene contenido confiable a través de los usuarios: basado en asociaciones que transfieren verdad desde las entidades a los recursos.
- Se detecta la veracidad, confiabilidad y credibilidad de una fuente de datos: usando anotaciones de verdad hechas por varios individuos.
- Se asigna valores de verdad a las fuentes de datos usando ontologías de verdad que asignan valores de verdad basados en contenido o metadata, que pueden ser transferidos desde datos conocidos a datos no conocidos.
- Se determinan los valores de verdad de los datos usando anotaciones para datos como blacklisting, autoritatividad, ranking basado en enlaces.

- Se chequea si el proveedor/contribuidor está contenido en una lista de proveedores confiables, se obtiene meta-información sobre la identidad del proveedor de la información.
- Existe una interdependencia entre el proveedor de los datos y el dato en sí; los datos por lo general son aceptados como verdaderos si provienen de un proveedor verídico y el proveedor de datos es considerado verídico si provee datos verdaderos.

Según Prat y Madnick el valor de un dato puede ser atómico o complejo (tablas, archivos, xml, etc.); en el artículo *Measuring Data Believability: A Provenance Approach* [20], se basan en valores atómicos y numéricos. El valor de un dato lo definen como la instancia del mismo. Se dividen en dos tipos de valores de datos, pudiendo ser una fuente o un resultado (es decir la salida de la ejecución de un proceso). Se presentan diferentes métricas de *believability* para cada tipo de valor.

Se define *transaction time* (tiempo de transacción) para el valor del dato como el tiempo de ejecución del proceso que generó ese dato en el caso de ser un valor de tipo resultado; si es de fuente ese valor viene adjunto al dato.

Se tiene en cuenta también la noción de *valid time (*tiempo válido), definida como el tiempo válido de un hecho, es el tiempo donde el hecho es verdadero en la realidad modelada [24], depende de la semántica de los datos. A su vez utilizan el concepto *possibility* (posibilidad) [20], definida como hasta qué punto el valor de un dato es posible; una distribución de posibilidad está asociada a los datos. Las distribuciones de la posibilidad se pueden adquirir de expertos, toman sus valores entre 0 (imposible) y 1 (totalmente posible) y se pueden definir en intervalos.

Es necesario tener en cuenta *trustworthiness* (veracidad) de los agentes (organización, grupo o persona) que proveen los datos para los diferentes dominios de conocimiento, medida como el valor *trustworthiness* que va en un rango de 0 a 1.

Trustworthiness del valor de un dato fuente es equivalente al valor *trustworthiness* del agente que proveyó el dato.

Para calcular *reasonableness* del valor del dato fuente es necesario definir métricas para *possibility*, *consistency over sources*, *consistency over time*.

Possibility se obtiene directamente del modelo de procedencia, usando la distribución de posibilidad del dato correspondiente.

Para calcular *consistency over sources* se consideran otros valores para ese mismo dato, obtenidos a partir de otras fuentes; calculando la distancia entre esos valores y el dato. Se transforman las distancias en semejanzas por medio del complemento a 1, y luego se calcula el promedio de todas ellas.

Para calcular *consistency over time* la idea general es que los valores del mismo dato no deberían variar demasiado en el tiempo, de otra manera serían menos creíbles.

Para calcular *temporal believability* (credibilidad temporal) se consideraron dos aspectos: credibilidad basada en la transacción y la cercanía a tiempos válidos y credibilidad basada en tiempos válidos superpuestos. La idea general es que el valor de un dato estimado (calculado de antemano) es más confiable a medida que se acerca al tiempo válido, sobre todo al tiempo válido final.

La credibilidad basada en la superposición de tiempos válidos mide la medida en que un valor de datos resultante de un proceso se deriva de valores de datos con *coherencia*, es decir, tiempos válidos superpuestos.

Para calcular *believability* del valor de un dato, se necesita considerar la procedencia o linaje del mismo; algunos aspectos de *believability* se transmiten a través de los diferentes valores que va tomando el dato en el transcurso de los diferentes procesos, por ejemplo, la *trustworthiness*. Sin embargo, algunos otros aspectos, como ser *possibility* o *temporality,* pueden no transmitirse junto al dato cuando éste sufre modificaciones a través de los distintos procesos; por lo que es necesario tener en cuenta el linaje completo del valor del dato.

En el momento de calcular el valor global de cualquiera de las tres subdimensiones de credibilidad, es necesario utilizar dos tipos de pesos (ponderaciones). El primero es el factor de descuento, el cual refleja la intuición de que la influencia de un vértice en otro decrece con el tamaño de

la ruta que separa uno de otro; el segundo se basa en derivadas parciales, que intentan medir la influencia del valor del dato en cierto proceso.

Estos pesos reflejan el hecho de que, para un proceso dado, los valores de los datos de entrada no contribuyen de igual manera en el proceso, y consecuentemente, en el resultado.

Luego que se calcularon los valores globales para cada una de las tres subdimensiones, el valor de *believability* global se calcula multiplicando estos tres valores.

Una dimensión relevante para nuestro proyecto es *reputation* (reputación), la misma puede ser clasificada en enfoques humanos o semi-automatizados. El enfoque basado en humanos es a través de una encuesta en una comunidad o mediante cuestionarios a otros miembros que puedan ayudar a determinar la reputación de una fuente o de una persona que publicó un conjunto de datos. El enfoque (semi)automatizado puede ser ejecutado por el uso de enlaces externos o rangos de páginas.

R*eputation* es una noción social de verdad. La verdad es usualmente representada en una red de verdad, donde los nodos son entidades y las aristas son los valores de verdad basados en una métrica, la cual refleja la reputación que una entidad le asigna a otra. Basados en la información presentada a un usuario, el mismo forma una opinión y juzga sobre la reputación del conjunto de datos o del publicador (editor).

En cuanto a la dimensión de *Verifiability* (verificabilidad) una forma de verificación en datos vinculados (enlazados) es proveer información básica de procedencia junto con el conjunto de datos. Otro mecanismo es el uso de firmas digitales, donde una fuente puede firmar un documento que contiene una serialización RDF o un grafo RDF, asegurando así al usuario que los datos que recibe son, de hecho, los datos que la fuente ha avalado.

Hay usuarios que se coordinan con el fin de mostrar información errónea, estos usuarios coordinados han correlacionado su comportamiento (por ejemplo, sus tweets son muy similares o sus votos tienen patrones similares), en el artículo *Measuring User Credibility in Social Media* se desarrolla un algoritmo denominado *CredRank*, el cual encuentra usuarios con comportamientos similares y los agrupa con la intención de darle un valor

menor a su opinión por ser poco creíble. Utiliza un método de agrupación jerárquica para agrupar usuarios similares, para esto mide la similitud entre un usuario base y sus vecinos, usando una función que mide esto en función del tiempo. Para cada dominio se usa una función específica para medir la similitud, por ejemplo, para medir la similitud de los usuarios de Twitter, calculamos la similitud de sus tweets. Se pueden utilizar varios métodos como *edit-distance*, *tf-idf*, *coeficiente de Jaccard*.

Luego se agrupa a los usuarios si su similitud excede un umbral τ. El valor de τ varía para diferentes dominios, y utilizando una fórmula, se asignan los pesos de los grupos.

A continuación, se muestra una tabla (Tabla 4) donde se detalla resumidamente para cada dimensión sus métricas:

Dimensión	Referencia	Métricas
Credibilidad	C. Batini y M. Scannapieco [1]	Calcula la veracidad de las declaraciones RDF (Resource Description Framework) basadas en la procedencia de la información y en la opinión de otros consumidores y le aplica una función de verdad. Las funciones de verdad están basadas en clasificaciones de los usuarios y en métodos basados en la procedencia o la opinión.
		Calcula la veracidad de una entidad, de un objeto o de un recurso: un tercero confiable brinda una medida de verdad objetiva para cada entidad, a través de información como ser cantidad de citas o reputación global. Una vez que cada entidad tiene su valor de verdad asignado, es posible realizar inferencias de verdad en las nuevas entidades que aparezcan.
		Se calcula la verdad entre dos entidades usando una combinación de un algoritmo de propagación que utiliza técnicas estadísticas sobre dos entidades a través de una trayectoria y un algoritmo de agregación basado en un mecanismo de ponderación para calcular el valor agregado

		sobre todas las trayectorias.
		Se obtiene contenido confiable a través de los usuarios: basado en asociaciones que transfieren verdad desde las entidades a los recursos.
		Se detecta la veracidad, confiabilidad y credibilidad de una fuente de datos: usando anotaciones de verdad hechas por varios individuos.
Credibilidad	C. Batini y M. Scannapieco [1]	Se asigna valores de verdad a las fuentes de datos usando ontologías de verdad que asignan valores de verdad basados en contenido o metadata, que pueden ser transferidos desde datos conocidos a datos no conocidos.
		Se determinan los valores de verdad de los datos usando anotaciones para datos como blacklisting, autoritatividad, ranking basado en enlaces.
		Se chequea si el proveedor/contribuidor está contenido en una lista de proveedores confiables, se obtiene meta-información sobre la identidad del proveedor de la información. Existe una interdependencia entre el proveedor de los datos y el dato en sí; los datos por lo general son aceptados como verdaderos si provienen de un proveedor verídico y el proveedor de datos es considerado verídico si provee datos verdaderos.
Consistencia a través del tiempo	Christian S. Jensen James Clifford Ramez Elmasri [24]	Para calcularla la idea general es que los valores del mismo dato no deberían variar demasiado en el tiempo, de otra manera serían menos creíbles.

Credibilidad temporal	Christian S. Jensen James Clifford Ramez Elmasri [24]	Se consideraron dos aspectos: credibilidad basada en la transacción y la cercanía a tiempos válidos y credibilidad basada en tiempos válidos superpuestos. La idea general es que el valor de un dato estimado (calculado de antemano) es más confiable a medida que se acerca al tiempo válido, sobre todo al tiempo válido final.
Reputación	C. Batini y M. Scannapieco[1]	Pueden ser clasificadas en enfoques humanos o semi-automatizados. El enfoque basado en humanos es a través de una encuesta en una comunidad o mediante cuestionarios a otros miembros que puedan ayudar a determinar la reputación de una fuente o de una persona que publicó un conjunto de datos. El enfoque (semi)automatizado puede ser ejecutado por el uso de enlaces externos o rangos de páginas.
Verificabilidad	C. Batini y M. Scannapieco[1]	Una forma de verificación en datos vinculados (enlazados) es proveer información básica de procedencia junto con el conjunto de datos. Otro mecanismo es el uso de firmas digitales, donde una fuente puede firmar un documento que contiene una serialización RDF o un grafo RDF, asegurando así al usuario que los datos que recibe son, de hecho, los datos que la fuente ha avalado.

Tabla 4. Dimensiones y sus respectivas métricas.

2- Perspectiva de la Industria

En esta sección se realiza una revisión desde la perspectiva de la industria, la cual se compone de entrevistas realizadas a usuarios de datos de las redes sociales y una investigación de las tecnologías existentes sobre la temática.

2.1- Entrevistas a Usuarios de Datos de Redes Sociales

Con el objetivo de enriquecer lo que se espera de una herramienta que determine el índice de credibilidad de una fuente se decidió consultar a

referentes de diferentes áreas, los cuales utilizan día a día los datos de las redes sociales, para su trabajo, toma de decisiones, informar, etc.

El siguiente es un resumen de entrevistas realizadas a Paula Scotti y Federico García (Antel), Carlos Álvarez (analista de redes sociales en IDatha) y Daniel Mazzone (periodista y docente), a los cuales se le agradece su tiempo y disposición.

Estas entrevistas resultaron de gran utilidad para el proyecto, dado que brindan una visión desde la experiencia y del conocimiento teórico y práctico del manejo de datos provenientes de redes sociales.

Según Paula Scotti y Federico García, Antel posee una fuerte política del cuidado del cliente, por ende, es muy importante la atención de reclamos y verificación de la credibilidad de los mismos. Reciben reclamos sobre los servicios, la empresa, así como también políticos. Las redes sociales facilitan la cercanía con el usuario, aunque por otro lado también habilita a mensajes malintencionados o reclamos no reales.

Para verificar la credibilidad de un cliente utilizan un método manual que consiste en:

- Investigar el perfil de la persona que realiza el reclamo, verificando la cantidad de seguidores, cantidad de tweets, la temática de los mismos (si es muy monotemático o si son solo peleas con diferentes cuentas, se lo considera sospechoso), las interacciones con sus seguidores, historial de reclamos realizado, tendencia política del usuario (esto es importante dado que la empresa es estatal), estética de la redacción del reclamo.
- Al contar la empresa con una base de datos fiable se solicita al usuario los datos personales para poder comparar con dicha base; en caso de que el usuario no proporcione ningún dato se lo considera falso.
- Para determinar la credibilidad en estos casos no importa el contexto, se determina la credibilidad de la persona en general y no en un contexto definido.

Carlos Álvarez, coincide en varios puntos sobre lo que aporta credibilidad a una fuente, ya que para él quién mandó el mensaje, el contenido y la

congruencia del mismo (en la historia, los mensajes publicados siguen una misma línea y no se contradicen en el tiempo), la masividad (cantidad de persona que tweetean, reenvían un mensaje, *linkean*), son factores determinantes a la hora de verificar la credibilidad de una fuente. Por otro lado, también agrega otros factores, como ser la antigüedad de un usuario en la red (lo cual ayuda a descartar que el usuario no sea un *boot*), la confianza de quién envió el mensaje, la cual es muy relevante, aunque ésta sea definida exclusivamente por el receptor de este. A su vez, al considerar los *repost* de un mensaje, el hecho de que una fuente fidedigna, como un periodista, lo use, lo vuelve más creíble.

Por el contrario, en lo referente a este último punto, Daniel Mazzone opina que el periodismo no toma nunca como fuente los datos obtenidos de las redes sociales. Él cree que el periodismo está ajeno a la crisis actual donde las noticias fluyen como verdades por las redes y se hace difícil discernir lo que es verdad de lo que es mentira, dejando de existir la verdad absoluta y pasando a ser la verdad una construcción colectiva; surgiendo así la verdad civil, verdad que no pertenece a nadie, pero en la que cree la mayoría (deja entrever esto en su libro *La mentira de la post verdad*).

Por otro lado, también expone que hoy en día el emisor no tiene compromiso, no tiene problema de dañar. Algunos medios están empezando a dejar el fact-checking del lado del lector, haciendo así que los usuarios se responsabilicen de la veracidad de lo que leen. ¿Por qué entonces dice que el periodista no es parte de esta crisis? Porque para él, el periodista debe verificar con varios medios, los cuales está seguro de que son verídicos (para dar credibilidad debe aparecer nombre y apellido) y no hacerse eco de las noticias que circulan.

Refiriéndonos a la confianza del emisor, Carlos Álvarez opina que la trayectoria, la cantidad de seguidores y su interacción, el hecho de que sea una figura pública, los conocimientos o estudios que tenga la persona con respecto al tema del mensaje (una persona puede ser creíble en ciertos temas; por tener estudios de una rama específica, pero no en todos), son los factores que ayudan a dar confianza en una persona.

Las redes sociales que utilizan en Antel para verificar la credibilidad son (en orden de importancia):

- Instagram (se usa para el vivo, historias, cobertura de espectáculo y eventos; y es usado para contemplar al público más joven.)
- Facebook
- Twitter (se usa mucho, es una red libertaria; se publica más contenido con un perfil más tecnológico.)
- Snapchat
- Youtube
- Spotify

A nivel técnico en IData por ejemplo las más usadas son:

- Twitter (porque es todo público, se pueden obtener muchos datos, aunque para conseguir historias antiguas se precisa del pago).
- YouTube
- Amazon, Trivago
- Análisis de búsquedas en google (te permite ver la cantidad de búsquedas por palabras en los últimos días).

Para el análisis de la información se ayudan también de las siguientes herramientas:

- R
- Python
- Panda (Mezcla de R y Python)
- Notebooks (Python)
- Herramientas para el análisis de Redes de contactos: Interacciones entre los diferentes contactos, análisis de grafos con respecto a los seguidores; se forman clusters de contactos. Luego se mide el nivel de confianza con respecto a cada cluster que se forme en la red.

Muchas veces se viralizan los mensajes para conseguir la masividad y darle así credibilidad a una noticia, Carlos Álvarez expone que esto se hace a través de *influencers*. Coincide con Antel en que los mismos tienen un perfil marcado, levantan noticias de marcas, hablan con una orientación política estipulada, generalmente se unen a otros usuarios que hacen lo mismo. Por lo tanto, para determinar si un *influencer* es una persona creíble, hay que observar sus *engagements,* así como también su *Social Authority* (índice que toma en consideración, la cantidad de mensajes y tweets que tiene; tiene en cuenta los retweets, la cantidad de seguidores).

Desde Antel exponen que una herramienta necesaria sería un CRM que unifique un cliente en las diferentes plataformas. También esperarían de una aplicación realizada para determinar la credibilidad de una fuente, que permita distinguir seguidores reales y manejar una comunidad de estos, que advierta casos de cuentas spammers u otro tipo de cuentas falsas (divididas en dos grupos; *Trolls* creados con un propósito específico y *Bots* reales) y que pueda brindar el resumen del comportamiento de una cuenta.

Capítulo 4 Dimensiones de Calidad de Datos para Evaluar la Credibilidad

En este capítulo se realiza un análisis de toda la información recabada en las secciones anteriores y se diseña la solución del problema planteado. Se encuentra dividido en dos subsecciones; definiciones de dimensiones de calidad, la cual incluye la especificación de los factores, definición de métricas asociadas a los mismos.

1- Definición de Dimensiones de Calidad

La definición de dimensiones se basó en la información recabada en la sección anterior. Se intenta utilizar los diferentes aportes de los artículos y personas entrevistadas para lograr una puesta en común, y así obtener un conjunto de dimensiones y factores completo y adecuado. Al definir las dimensiones se distinguió entre las dimensiones asociadas al *usuario*, pudiendo corresponder a una organización o a una persona; y las dimensiones asociadas al *clip*, el cual engloba al mensaje.

En las tablas 5 y 6 se presenta un resumen de las dimensiones y factores definidos, divididos en dos grupos asociados al *usuario* y asociados al *clip*, respectivamente.

1.1- Dimensiones Asociadas al Usuario

En esta sección se describen las dimensiones asociadas al usuario.

- ***Authenticity* (Autenticidad)**

Authenticity (Autenticidad), entendida como el grado en que un usuario es auténtico o real.

Se desprende de las entrevistas realizadas, sobre todo la información proporcionada por Antel y Carlos Álvarez; la cual detalla el proceso de verificación de un usuario. En este proceso, según Antel, se investiga el perfil de la persona que realiza el reclamo, verificando la cantidad de seguidores, cantidad de tweets, lo que se asume refiere al factor *Popularity (Popularidad)*.

También se destaca la importancia de las interacciones del usuario con sus seguidores lo cual refiere al factor *Activity (Actividad)*.

Por otro lado, se hace hincapié en la relevancia de la temática de los clips, ya que si es muy monotemático se lo considera sospechoso. Carlos Álvarez acompaña este pensamiento, dado que, para él, el contenido del clip y la congruencia de éste son un factor determinante en la verificación del usuario. En base a ambas opiniones se decidió determinar un factor *Topic (Temática)* el cual refiere al grado de heterogeneidad de la temática de los clips que realiza un usuario, es decir si es monotemático o politemático.

La antigüedad del usuario en la red es importante dado que ayuda por ejemplo a descartar que el usuario sea un bot; se representa como el factor *Antiquity (Antigüedad)*, el cual refiere a la antigüedad del usuario en la red, tomando en cuenta la fecha de creación del mismo.

Otro aspecto relevante para la autenticidad se encuentra en C. Batini y M. Scannapieco [1], donde se describe la dimensión *Verifiability (Verificabilidad)* como el *grado y facilidad con que se puede verificar la corrección de información*. En este trabajo se va a tomar la idea de esta dimensión para definir un factor que representa una parte más específica de la misma, la cual está relacionada con la autenticidad del usuario. Por ende, se define *Checkable (Verificable)*, que refiere a si el usuario puede catalogarse como validado por la red.

- ***Expertise* (Experticia)**

Expertise se define por términos tales como experto, experimentado, competente [15]; por ende, la dimensión de experiencia de la credibilidad captura el conocimiento percibido y habilidad de la fuente.

Para Carlos Álvarez los conocimientos o estudios que tenga la persona con respecto al tema del clip son muy importantes para calcular la credibilidad de este (una persona puede ser creíble en ciertos temas; por tener estudios de una rama específica, pero no en todos). Por ende, se toma *Expertise* como una dimensión de calidad que abarca todo el conocimiento que tiene un usuario sobre un tema.

Pensándolo de esa manera se pueden distinguir diferentes fuentes de conocimiento y por ende diferentes factores. *Certification (Certificación)*, se refiere al grado de conocimiento certificable que tiene un usuario, por ejemplo, título académico, certificaciones realizadas, reconocimientos obtenidos dentro del área. *Experience (Experiencia)*, se refiere a la cantidad de años que la persona u organización ha dedicado a formarse (nivel académico y laboral) sobre un tema. *Production (Producción),* se refiere a las investigaciones o publicaciones realizadas por el usuario de un tema específico.

- ***Reputation (Reputación)***

Según C. Batini y M. Scannapieco la reputación se relaciona directamente con la credibilidad y es un enjuiciamiento hecho por el usuario para determinar la integridad de una fuente. Puede ser asociada con el proveedor de los datos, una persona, organización, grupo de personas; o puede ser una característica del conjunto de datos.

Según Prat y Madnick la reputación es una noción social de verdad. La verdad es usualmente representada en una red de verdad, donde los nodos son entidades y las aristas son los valores de verdad basados en una métrica, la cual refleja la reputación que una entidad le asigna a otra.

Se define *Reputation* como el grado en que un usuario es considerado fiable, respetable y se lo relaciona con *Confidence (Confianza)* y *Engagement (Compromiso).*

Chenyun Dai, Dan Lin, Elisa Bertino, y Murat Kantarcioglu [19] relacionaron la confianza con la credibilidad; Carlos Álvarez le da un papel muy importante a la misma, expresando que no solo es importante la cantidad de gente que confía en algo sino también el prestigio de quien avala a ese algo o alguien, ejemplo un periodista tiene más peso que un usuario normal. Por ende, se determina que la confianza se refiere a la cantidad y calidad de usuarios que avalan al emisor del clip.

Carlos Álvarez también expresó que la cantidad de persona que twittean, reenvían un mensaje, o linkean el mismo es un factor que se relaciona directamente con la credibilidad de una persona o clip, se define como

Engagement (la cantidad de veces que un clip de un usuario es compartido o mencionado por otros usuarios).

Dimensiones Usuario	**Factores**
***Authenticity**: Es el grado en que un usuario es auténtico o real.*	**Popularity**: Se refiere a la cantidad y calidad de seguidores que tiene un usuario.
	Activity: Se refiere a la cantidad de interacciones que tiene un usuario.
	Topic: Se refiere al grado de heterogeneidad de la temática de los clips que realiza un usuario, es decir si es monotemático o politemático.
	Account Age: Se refiere a la antigüedad del usuario en la red, tomando en cuenta la fecha de creación de este.
	Checkable: Se refiere a si el usuario puede catalogarse como validado por la red.
***Expertise**: Es el grado en que el usuario es experimentado o competente en un tema*	**Certification**: Se refiere al grado de conocimiento certificable que tiene un usuario, por ejemplo, título académico, certificaciones realizadas, reconocimientos obtenidos dentro del tema.
	Production: Se refiere a las investigaciones o publicaciones realizadas por el usuario de un tema específico.
	Influence: Se refiere a la cantidad de usuarios relacionados con el tema que lo avalan.
	Experience: Se refiere a la cantidad de años que el usuario ha dedicado a formarse (nivel académico y laboral) sobre el tema.
***Reputation**: Es el grado en que un usuario es considerado fiable, respetable.*	**Engagement**: Se refiere a la cantidad de veces que un clip de un usuario es compartido o mencionado por otros usuarios.
	Confidence: Se refiere a la cantidad y calidad de usuarios que avalan al mismo.

Tabla 5 Dimensiones y Factores asociados al Usuario

1.2- Dimensiones Asociadas al Clip

En esta sección se describen las dimensiones asociadas al Clip.

- ***Temporality*** **(Temporalidad)**

En el artículo *Measuring Data Believability: A Provenance Approach* [16], Prat y Madnick consideran el concepto de *Temporal Believability* donde el valor de un dato es más confiable a medida que se acerca al tiempo válido, siendo éste el tiempo donde el hecho es verdadero en la realidad modelada. Por lo tanto, se definió la dimensión *Temporality* (Temporalidad) como el grado en que un clip es creíble basándose en un tiempo válido; asignándole el factor *Timeliness*, que refiere a la vejez del clip vinculada con tiempo válido y con la oportunidad de este, entendiendo oportunidad como uso del dato para la tarea en cuestión.

- ***Provenance*** **(Procedencia)**

Tomando en cuenta la conclusión arribada por Y.L. Simmhan, B. Plale y D. Gannon mencionada en el artículo [14], donde se establece que *believability* del valor de los datos depende también del origen de este, se definió, en este sentido, la dimensión *Provenance (Procedencia)*, la cual refiere a la cuenta y al lugar donde se originó el clip.

Los autores distinguen que la procedencia del dato depende tanto de la fuente y de su posterior historial de procesamiento; en el ámbito de este proyecto, se van a representar como dos factores, *Location (Ubicación)* que toma como referencia la ubicación geográfica del clip con respecto a la cuenta que originó el mismo; y *Record (Historial)*, que toma en consideración la reputación de la cuenta origen del clip combinada con el camino de este.

Dimensiones Clip	**Factores**
Temporality*: El grado en que un clip es creíble basándose en un tiempo válido.*	**Timeliness**: Se refiere a la vejez del clip vinculada con tiempo válido y con la oportunidad de este.
Provenance*: Se refiere a la cuenta y al lugar donde se originó el clip.*	**Location**: El grado en el cual la ubicación geográfica es válida para el clip (deben coincidir la ubicación del clip y su ubicación válida).
	Reputation Path: Se refiere a la reputación de la cuenta origen del clip, combinada con el camino de este.

Tabla 6 Dimensiones y Factores asociados al Clip

2- Definición de Métricas

En esta sección se definen las métricas asociadas a los factores especificados anteriormente, basándose en la información recabada en el capítulo anterior. Para esto, previamente definimos un conjunto de constantes y metadatos, que luego se van a usar en las métricas.

2.1- Constantes

Se define una tabla (Tabla 7) de constantes para tomar como referencia en los cálculos de normalización de las distintas métricas. La misma es dependiente del conjunto de datos de Twitter obtenidos hasta el momento del cálculo; se construye tomando el máximo de ciertos valores de interés o haciendo diferentes cálculos con los datos obtenidos.

Nombre	Descripción
followers_avg_set	Valor promedio de seguidores que puede tener un usuario basándose en los datos recolectados hasta el momento.
tweets_retweets_pub_avg_set	Valor promedio de tweets y re-tweets realizados por el usuario teniendo en cuenta la variable statuses_count del usuario (dato procedente del json indicado en Figura 4), basándose en los datos recolectados hasta el momento.
statuses_avg_set	Valor promedio de los tweets de un usuario que son respuesta a otros tweets, basándose en los datos obtenidos hasta el momento.
avg_favourite	Valor promedio de me gustas que tienen los usuarios basándose en los datos recolectados hasta el momento.
statuses_max_ref	Valor máximo de tweets y re-tweets realizados por el usuario basándose en valores de referencia.
tweets_total_set	Valor total de tweets recolectados hasta el momento.

Twitter_foundation_date	Fecha de fundación de Twitter, 21 de marzo de 2006.
avg_quote_count	Valor promedio de citas que tiene un tweet, basándose en los datos recolectados hasta el momento.
avg_favourite_count	Valor promedio de me gustas que tiene un tweet, se calcula teniendo en cuenta la suma total de me gustas te tweets de un usuario y luego se divide entre la cantidad total de usuarios; basándose en los datos recolectados hasta el momento.
short_time_frame	Período de tiempo determinado para calcular el exceso de actividad de los usuarios.
max_excess	Valor máximo de cantidad de actividades que no puede superar un usuario.
max_title_count	Valor máximo de títulos basándose en los datos recolectados hasta el momento.
max_exper_years_count	Valor máximo de experiencia de un usuario, basándose en los datos recolectados hasta el momento.
max_pub_count	Valor máximo de publicaciones realizadas por un usuario, basándose en los datos recolectados hasta el momento.
max_user_influ_count	Valor máximo de usuarios que avalan a otro usuario en un tema en particular, basándose en los datos recolectados hasta el momento.
avg_retweets_count	Valor promedio de retweets que tiene un tweet, basándose en los datos recolectados hasta el momento.
credibility_range	Valor utilizado para indicar si un usuario es creíble o no. Si la métrica de credibilidad general supera este valor el usuario se

	considera creíble y no creíble en caso contrario.

Tabla 7 Valores máximos para normalización.

2.2- Metadatos

Los datos obtenidos a partir de la API de Twitter siguen el siguiente formato (se muestran solamente los tags utilizados) (Figura 4):

{'created_at': 'Wed Dec 19 22:31:16 +0000 2018',
'id': 1075518958286106624,
'text': 'Mejorando el dia',
'source': 'Twitter for Android',
'truncated': False,
'in_reply_to_status_id': None,
'in_reply_to_user_id': None,
'in_reply_to_screen_name': None,
'user': { 'id': 194288401,
'name': '####',
'screen_name': ####,
'location': 'Corrientes, Argentina',
'url': None,
'description': 'Productora de @### \n ###@hotmail.com ✉',
'protected': False,
'verified': False,
'followers_count': 244,
'friends_count': 221,
'listed_count': 0,
'favourites_count': 1644,
'statuses_count': 291,
'created_at': 'Thu Sep 23 20:38:32 +0000 2010',

```
                                },
 'coordinates': None,
 'place': None,
 'quoted_status_id': 1075514880931872769,
 'quoted_status': {... },
 'is_quote_status': True,
 'quote_count': 0,
 'reply_count': 0,
 'retweet_count': 0,
 'favorite_count': 0,
 'entities': {
             'hashtags': [],
             'urls': [],
             'user_mentions': [],
            'symbols': []
  },
}
```

Figura 4. Formato Json de un Tweet.

Se definieron tres tipos de nodos: usuario (*TwitterUser*), tweet (*Tweet*) y hashtag (*Hashtag*).

El nodo *TwitterUser* contiene los datos relacionados con el usuario que escribió el tweet, utiliza los siguientes metadatos (Tabla 8):

User	Descripción de tags
id	Identificador único del usuario en Twitter.
name	Nombre definido por el usuario.
screen_name	Alias o nombre de pantalla con el que el usuario se identifica.
verified	Cuando es verdadero indica que es una cuenta verificada.

location	Lugar geográfico definido por el usuario al cual la cuenta corresponde.
followers_count	El número de seguidores que tiene la cuenta.
friends_count	El número de usuarios que la cuenta sigue.
created_at	Fecha de creación de la cuenta.
url	Url asociada a la cuenta, definida por el usuario.
description	Descripción de la cuenta realizada por el usuario.
listed_count	El número de listas públicas de las que el usuario es miembro.
favourites_count	El número de tweets que el usuario ha dado me gusta.
statuses_count	El número de tweets, incluidos retweets, que el usuario ha publicado.
protected	Si es verdadero indica que el usuario decidió proteger sus tweets.

Tabla 8 Metadatos para el nodo TwitterUser.

El nodo *Tweet* contiene los datos relacionados con el tweet en sí a partir de los metadatos que se listan a continuación (Tabla 9):

Tweet	Descripción de tags
id	Identificador único del tweet.
created_at	Fecha en la cual el tweet fue creado.
coordinates	Representa el punto geográfico donde fue creado el tweet reportado por la aplicación del usuario.
place	En caso de existir indica que el tweet está asociado a un lugar específico.
truncated	Indica si el valor del campo text fue truncado por ser demasiado largo.
text	El texto del tweet.
full_text	En caso de del tweet haber sido truncado el texto completo se encuentra en este parámetro.
quote_count	Indica aproximadamente cuántas veces el tweet fue citado.
reply_count	Indica aproximadamente cuántas respuestas tuvo el tweet.

retweet_count	Indica aproximadamente cuántas veces fue retuiteado
favorite_count	Indica aproximadamente a cuántos usuarios le gustó el tweet.
hashtags	Una lista con todos los hashtags que se encuentran en el texto del tweet.
in_reply_to_status_id	En caso de ser un comentario a un tweet, tiene el valor del id del tweet original. En otro caso es nulo.
in_reply_to_user_id	En caso de ser un comentario a un tweet, tiene el valor del id del usuario creador del tweet original. En otro caso es nulo.
user_mentions	Muestra los usuarios mencionados en el texto del tweet.

Tabla 9 Metadatos para Nodo Tweet.

El nodo *hashtag* guarda los distintos hashtags utilizados con el fin de luego asociarlos a los diferentes usuarios y tweets, utiliza el metadato text del hashtag del tweet (Tabla 10):

Hashtag	Descripción de tags
text	Nombre del hashtag sin #

Tabla 10 Metadatos para nodo Hashtag.

Al mismo tiempo se usaron relaciones para vincular tanto usuarios como tweets (Tabla 11):

Relationships	Descripción
PostedBy	Relaciona el tweet con el usuario que lo escribió.
HashtagFromTweet	Relaciona el hashtag con los tweets que lo usan.
FollowedBy	Relaciona al usuario con sus seguidores.
MentionToUser	Relaciona el tweet con los usuarios que se mencionan en él.
QuotedFromTweet	Relaciona al tweet con el tweet citado, en caso de corresponder a una cita.
QuotedFromUser	Relaciona el usuario del tweet con el usuario del tweet citado, en caso de corresponder a una cita.
RetweetedFromTweet	Relaciona al tweet con el retweet original, en caso de corresponder a un retweet.

RetweetedFromUser	Relaciona el usuario del tweet con el usuario del retweet original, en caso de corresponder a una retweet.

Tabla 11 Relaciones entre nodos Tweets, TwitterUsers y Hashtags.

2.3- Métricas Asociadas al Usuario

En esta sección se muestran las métricas asociadas al Usuario, representadas en la Tabla 12, agrupadas por Dimensión y Factor.

Dimensión	Factor	Métrica
Authenticity	**Popularity**	Followers_quantity
		Followers_quality
		Tweets_retweets_published
		Commented_tweets
		Favorite
		Excess_activity
	Topic	Similar_words
	Account Age	Antiquity
	Checkable	Verified
Expertise	**Certification**	Degree_of_education
	Experience	Level_of_education
	Production	Amount_of_publications
	Influence	Degree_of_influence
	Confidence	Followers_count

Reputation	**Engagement**	Quote_count
		Retweets_count_user
		Favourite_count_user

Tabla 12 Dimensiones, Factores y Métricas asociadas al usuario.

❖ Dimensión *Authenticity*

En la Tabla 13 se detallan las métricas asociadas a cada factor de la dimensión *Authenticity*.

Factores	Métricas
Popularity	***followers_quantity*** **Descripción:** Se refiere a la cantidad de seguidores que tiene un usuario. **Unidad:** [0, 1]. **Granularidad**: usuario **Método:** Se calcula a partir del tag ***followers_count*** del usuario y se normaliza usando ***followers_avg_set***. ***followers_quantity = followers_count /followers_avg_set*** Si ***followers_quantity*** > 1 entonces ***followers_quantity*** = 1.
Popularity	***followers_quality*** **Descripción:** Se refiere al promedio de calidad de los seguidores que tiene un usuario. **Unidad:** [0, 1]. **Granularidad**: usuario **Método:** Se obtienen todos los usuarios que siguen a un usuario dado (GET followers/list API Twitter). ***followers_quality = (Σuser_quality*(i)) / *followers_count*** Con **i:1..*followers_count*** y ***user_quality*(i)** la medida de calidad del usuario **i**. ***user_quality* = (p1*followers_quantity + p2*tweets_retweets_published + p3*commented_tweets + p4*favorite + p5*verified + p6* (1-similar_words) +**

	p7antiquity* + p8**followers_count* + p9**quote_count* + p10**retweets_count_user* + p11**favorite_count_user) / (p1+p2+p3+p4+p5+p6+p7+p8+p9+p10+p11)*** Siendo p1 a p11 el peso de cada una de las métricas, es decir, cada métrica va a influir de manera diferente en el valor de la métrica de calidad del usuario. Se utiliza el promedio ponderado para la normalización, es decir, se divide la métrica ***user_quality*** entre la suma de los pesos de las métricas que la componen.
Popularity	***tweets_retweets_published*** **Descripción:** Cantidad de tweets publicados por el usuario. **Unidad:** [0, 1]. **Granularidad**: Usuario **Método:** Se calcula a partir del valor del tag ***user_statuses_count*** del usuario y se normaliza usando ***tweets_retweets_pub_avg_set***. ***tweets_retweets_published = user_statuses_count / tweets_retweets_pub_avg_set*** Si ***tweets_retweets_published*** >1 entonces ***tweets_retweets_published*** =1.
Popularity	***commented_tweets*** **Descripción:** Cantidad de comentarios realizados por el usuario. **Unidad:** [0, 1]. **Granularidad:** Usuario **Método:** Se calcula a partir de la cantidad de veces que aparece el tag ***in_reply_to_status_id*** en tweets correspondientes al usuario y se normaliza dividiendo entre ***statuses_avg_set***. Si ***commented_tweets***> 1 entonces ***commented_tweets***= 1.
Popularity	***favorite*** **Descripción:** Cantidad de me gusta realizados por el usuario. **Unidad:** [0, 1]. **Granularidad:** Usuario **Método:** Se calcula a partir de tag ***favourites_count*** del usuario y se normaliza con ***avg_favourite***

	favorite = *favourites_count / avg_favourite* si ***favorite*** > 1 entonces ***favorite*** = 1.
Popularity	***excess_activity*** **Descripción**: Valor que refleja los excesos de actividad de un usuario en un periodo corto de tiempo, abarcando la actividad me gustas, comentarios, seguidores, tweet y retweets de un usuario. **Unidad:** Boolean **Granularidad:** Usuario **Método**: Se toman las métricas ***followers_quantity***, **favorite**, ***tweets_retweets_published***, ***commented_tweets***, al inicio del periodo **short_time_frame** y al final del mismo, y se restan los valores obtenidos respectivamente. Si algún valor > ***max_excess***, **excess_activity** = true en caso contrario **excess_activity** = false.
Topic	***similar_words*** **Descripción**: Cantidad de palabras similares entre los diferentes tweets publicados o retwitteados por el usuario. **Unidad:** [0, 1] **Granularidad:** Usuario **Método:** Tomando los tweets publicados o retwitteados por cada usuario, se genera un vector de palabras con el conjunto de publicaciones del usuario; formado por el par (palabras existentes en el mensaje, número de veces que apareció). Para generar un vector de palabras previamente hay que realizar un preprocesamiento de cada publicación, el cual está formado por los siguientes pasos: ● Se pasan todas las palabras del mensaje a minúsculas. ● Se eliminan todos los signos de puntuación excepto los guiones. ● Se eliminan todos los artículos y preposiciones. ● Se eliminan links y emoticones. ● Se desconjugan los verbos. Luego los vectores de palabras de cada usuario son comparados.

Account Age	***Antiquity*** **Descripción:** Diferencia entre la fecha de creación del perfil del usuario en Twitter y la fecha actual. **Unidad:** [0, 1]. **Granularidad:** Usuario **Método:** Se calcula a partir del valor del tag ***created_at*** del usuario, la **f_actual** (fecha actual) y ***twitter_foundation_date***. ***antiquity* = (f_actual - *created_at*) / (f_actual - *twitter_foundation_date*)**
Checkable	***Verified*** **Descripción:** Categorización de usuario verificado realizada por Twitter. **Unidad:** Boolean. **Granularidad:** Usuario **Método:** Se obtiene a partir del tag booleano ***verified*** del usuario.

Tabla 13 Dimensión Authenticity, Factores y Métricas asociadas.

- Dimensión *Expertise*

Dado que se decidió basarse en datos obtenidos a partir de la red Twitter, y ésta no cuenta con demasiada información sobre nivel académico o profesión de un usuario; siendo a su vez, un problema aparte automatizar la integración con otras redes que proporcionen estos datos; se resolvió no implementar los métodos para estas métricas. De todas maneras, se podría obtener la información a partir de otras fuentes externas como LinkedIn[25], ResearchGate[51] o alguna otra red o fuente de datos que contenga información académica de los usuarios. Igualmente se describen las métricas y su implementación de manera superficial, suponiendo que se cuenta con la información necesaria proporcionada por alguna fuente de datos.

La Tabla 14 detalla las métricas asociadas a cada factor de la dimensión *Expertise*.

Factores	Métricas
Certification	***Degree_of_education*** **Descripción:** Cantidad de títulos, certificados o cursos que tiene un usuario. **Unidad:** [0, 1] **Granularidad:** Usuario **Método**: Se obtienen la cantidad de títulos o certificados del usuario, y si divide la misma entre ***max_title_count***, para normalizar.
Experience	***Level_of_education*** **Descripción:** Cantidad de años que la persona ha dedicado a formarse sobre algún tema. **Unidad:** [0, 1] **Granularidad:** Usuario **Método**: Se obtiene la fecha en la cual la persona se empezó a formar en su especialidad y se la resta a la fecha actual para obtener la cantidad de años. Ese valor se divide entre ***max_exper_years_count***.
Production	***Amount_of_publications*** **Descripción:** Cantidad de investigaciones o publicaciones realizadas por el usuario de un tema específico. **Unidad:** [0, 1] **Granularidad:** Usuario **Método**: Se obtiene la cantidad de investigaciones o publicaciones realizadas por el usuario y se divide entre ***max_pub_count*** para normalizar.
Influence	***Degree_of_influence*** **Descripción:** Cantidad de usuarios relacionados con el tema en el cual el usuario se especializa que lo avalan. **Unidad:** [0, 1] **Granularidad:** Usuario

	Método: Se obtiene la cantidad de usuarios especializados en cierto tema que avalan al usuario y se divide entre ***max_cant_influ_usu*** para normalizar.

Tabla 14 Dimensión Expertise, Factores y Métricas asociadas.

❖ Dimensión *Reputation*

En la Tabla 15 se detallan las métricas asociadas a cada factor de la dimensión *Reputation*.

Factores	Métricas
Confidence	***followers_count*** **Descripción:** Se refiere a la cantidad de seguidores que tiene un usuario. **Unidad:** [0, 1]. **Granularidad**: usuario **Método:** Se calcula a partir del tag ***followers_count*** del usuario y se normaliza usando ***followers_avg_set***. ***followers_count = followers_count /followers_avg_set.*** Si ***followers_count*** > 1 entonces ***followers_count*** = 1.
Engagement	***quote_count*** **Descripción:** Cantidad de citas que tienen todos los tweets de un usuario. **Unidad:** [0, 1] **Granularidad:** Usuario **Método:** Se calcula a partir del tag ***quote_count*** de los tweets de un usuario. Se suman todos los ***quote_count*** de cada tweet y para normalizar se usa ***avg_quote_count***. ***quote_count = (Σquote_count (i)) /avg_quote_count,*** tomando i entre 1 y la cantidad de tweet del usuario. Si ***quote_count*** > 1 entonces ***quote_count*** = 1.
Engagement	***retweets_count_user*** **Descripción:** Cantidad de veces que fueron retwitteados los tweets de un usuario.

	Unidad: [0, 1] **Granularidad:** Usuario **Método:** Se calcula a partir del tag ***retweet_count*** de los tweets de un usuario. Se suman todos los ***retweet_count*** de cada tweet y para normalizar se usa ***avg_retweets_count***. ***retweets_count_user= (*** Σ***retweet_count (i))/avg_retweets_count,*** tomando i entre 1 y la cantidad de tweet del usuario. Si ***retweets_count_user***> 1 entonces ***retweets_count_user***= 1.
Engagement	***favourite_count_user*** **Descripción:** Cantidad de me gustas que tienen todos los tweets de un usuario. **Unidad:** [0, 1] **Granularidad:** Usuario **Método:** Se calcula a partir del tag ***favorite_count*** de los tweets de un usuario. Se suman todos los ***favorite_count*** de cada tweet y para normalizar se usa ***avg_favourite_count.*** ***favorite_count _user = (*** Σ***favorite_count(i)) /avg_favourite_count,*** tomando i entre 1 y la cantidad de tweet del usuario. Si ***favorite_count _user***> 1 entonces ***favorite_count _user***= 1.

Tabla 15 Dimensión Reputation, Factores y Métricas asociadas.

2.4- Métricas Asociadas al Mensaje

En esta sección se muestran las métricas asociadas al Mensaje, representadas en la Tabla 16, agrupadas por Dimensión y Factor.

Dimensión	Factor	Métrica
Temporality	**Timeliness**	Timeliness

Provenance	**Location**	Same_location
	Reputation Path	Message_invariability

Tabla 16 Dimensiones, Factores y Métricas asociadas al mensaje.

- Dimensión de *Temporality*

En la Tabla 17 se detallan las métricas asociadas a cada factor de la dimensión *Temporality*.

Factores	Métricas
Timeliness	***Timeliness*** **Descripción:** Grado en el que un mensaje no es caduco o es oportuno. **Unidad:** [0, 1] **Granularidad:** Tweet **Método:** Se verifica que el contenido del mensaje corresponde a la fecha en la que fue publicado.

Tabla 17 Dimensión Temporality, Factores y Métricas asociadas.

- Dimensión de Provenance:

La siguiente tabla, Tabla 18, muestra en detalle las métricas asociadas a cada factor de la dimensión *Provenance*.

Factores	Métricas
Location	***same_location*** **Descripción:** Coincidencia entre la ubicación del mensaje y la ubicación válida del mismo. **Unidad:** Boolean **Granularidad:** Tweet

	Método: Se comparan el campo coordinates y place del tweet si estos coinciden el valor es true, en caso contrario es false.
Reputation Path	***message_invariability*** **Descripción:** El grado en el mensaje no varía desde el origen al destino. **Unidad:** [0, 1] **Granularidad:** Tweet **Método**: Se toma el mensaje de origen y el mensaje de destino y se arma un vector de palabras por cada uno. Para generar un vector de palabras previamente hay que realizar un preprocesamiento de cada mensaje, el cual está formado por los siguientes pasos: ● Se pasan todas las palabras del mensaje a minúsculas. ● Se eliminan todos los signos de puntuación excepto los guiones. ● Se eliminan todos los artículos y preposiciones. ● Se eliminan links y emoticones. ● Se desconjugan los verbos. Luego se comparan los dos vectores.

Tabla 18 Dimensión Provenance, Factores y Métricas asociadas.

Capítulo 5 Implementación

En esta sección se especifica todo lo que a la implementación del prototipo refiere, lo cual incluye decisiones tecnológicas tomadas, la arquitectura, el proceso de implementación y problemas encontrados.

1- Tecnología Utilizada

Se utiliza como lenguaje de programación Python. Para la persistencia de los datos se usan dos bases de datos no relacionales: MongoDB (1.3.3) y Neo4j (1.3.2), siendo esta última una base de datos orientada a grafos. Para mejorar la portabilidad y escalabilidad se separó la persistencia de la lógica en contenedores Docker.

A continuación, se detallan las distintas tecnologías utilizadas para la implementación del prototipo.

1.1- Docker

Docker crea herramientas simples y un enfoque de empaque universal que agrupa todas las dependencias de la aplicación dentro de un contenedor que luego se ejecuta en Docker Engine. Docker Engine permite que las aplicaciones en contenedores se ejecuten en cualquier infraestructura, resolviendo el *infierno de dependencia* para los desarrolladores [26] [27].

1.2- Python

Lenguaje de programación multiparadigma, soporta orientación a objetos, programación imperativa y programación funcional. Cuenta con varias bibliotecas para el manejo de datos, así como drivers para la conexión con las bases de datos no relacionales como MongoDB y Neo4j [28].

1.2.1- Flask

Flask es un framework para aplicaciones web basado en WSGI (Web Server Gateway Interface) [29] [30].

1.3- Bases de Datos

En esta sección se detallan las distintas bases de datos utilizadas.

1.3.1- Base de Datos Relacionales vs Base de Datos Orientadas a Grafos

Las bases de datos relacionales (RDBMS) mantienen tablas que están definidas por conjuntos de filas y columnas. Una fila puede percibirse como un objeto, mientras que las columnas serían atributos / propiedades de esos objetos. Un modelo relacional presenta muchas debilidades para trabajar con información compleja e interconectada, tiene una capacidad limitada para capturar explícitamente la semántica de los requisitos. Los modelos de datos basados en esquemas, por su propia definición, establecen límites sobre cómo se almacenarán los datos, es necesario un proceso manual para rediseñar el esquema en caso de que se quiera adaptar a nuevos datos. Mientras que una base RDBMS está optimizada para datos agregados, las bases de datos orientadas a grafos están optimizadas para datos altamente conectados.

Las bases de datos orientada a grafos son muy trascendentales cuando se trabaja en áreas donde la información sobre interconectividad de datos o topología es importante. Se han desarrollado implementaciones internas para hacer frente a la necesidad de los sistemas de bases de datos orientadas a grafos. Algunos ejemplos serían el Open Graph de Facebook, Knowledge Graph de Google, FlockDB de Twitter.

Un grafo es un objeto que contiene nodos y relaciones. Los nodos tienen propiedades y están organizados por relaciones que también tienen propiedades. Este modelo de datos altamente dinámico en el que todos los nodos están conectados por relaciones permite recorridos rápidos a lo largo de las aristas entre vértices. Un beneficio particular es el hecho de que los recorridos están localizados y no tienen que tener en cuenta conjuntos de datos no relacionados.

A pesar de las limitaciones involucradas en el uso de RDBMS, hay muchos beneficios; estas bases de datos están bien respaldadas por sus respectivos proveedores, están probadas, sus fortalezas y limitaciones son bien conocidas, tienen un lenguaje de consulta común (SQL) y estas plataformas están extremadamente bien documentadas. En contraste, las bases de datos orientadas a grafos enfrentan un conjunto particular de desafíos inherentes a su diseño.

La recuperación efectiva de información de un grafo requiere lo que se conoce como recorrido transversal. Un recorrido de grafo implica *caminar* a lo largo de sus elementos. El recorrido es una operación fundamental para la recuperación de datos. Una diferencia importante entre una consulta transversal y una consulta SQL es que las transversales son operaciones localizadas. No hay un índice de adyacencia global en su lugar cada vértice y borde del grafo almacenan un *mini-índice* de los objetos conectados a él. Esto significa que el tamaño del grafo no tiene impacto en el rendimiento en un recorrido y que las costosas operaciones de agrupación realizadas en las sentencias JOIN SQL no son necesarias. Es importante tener en cuenta que los índices globales existen en Neo4J, pero solo se usan cuando se intenta encontrar el punto de inicio de un recorrido. Los índices son necesarios para recuperar rápidamente los vértices en función de sus valores.

El mundo de las bases de datos de grafos aún no se ha estandarizado en un lenguaje para el desplazamiento y la inserción. Esta falta de estandarización ha conducido a implementaciones y marcos muy diferentes para la interacción de datos. Las implementaciones disponibles van desde la API de Java de Neo4j, la interfaz REST, un lenguaje DSL llamado Gremlin y otro llamado Cypher. Gremlin implementa recorridos a través de un sistema llamado tubería; cada parte de la consulta es un paso a la siguiente de una manera progresiva. Sin embargo, Cypher intenta usar más de un sistema de palabras clave que intenta ser más como SQL. [31]. En 2019 se propone crear un nuevo estándar para los lenguajes de base de datos de grafos llamado Graph Query Language (GQL) tomando a Cypher como base.

1.3.2- Neo4j

Neo4j usa grafos para representar datos y las relaciones entre ellos. Un grafo se define como cualquier representación gráfica formada por vértices (se ilustran mediante círculos) y aristas (se muestran mediante líneas de intersección). Dentro de estas representaciones gráficas, se tienen varios tipos de grafos:

- Grafos no dirigidos: los nodos y las relaciones son intercambiables, su relación se puede interpretar en cualquier sentido.
- Grafos dirigidos: los nodos y las relaciones no son bidireccionales por

defecto.
- Grafos con peso: en este tipo de gráficas las relaciones entre nodos tienen algún tipo de valoración numérica. Eso permite luego hacer operaciones.
- Grafos con etiquetas: estos grafos llevan incorporadas etiquetas que pueden definir los distintos vértices y también las relaciones entre ellos.
- Grafos de propiedad: es un grafo con peso, con etiquetas y donde podemos asignar propiedades tanto a nodos como relaciones (por ejemplo, cuestiones como nombre, edad, país de residencia, nacimiento). Es el más complejo [32].

Características de Neo4j

1.- Rendimiento: Las bases de datos orientadas a grafos como Neo4j tienen mejor rendimiento que las relacionales (SQL) y las no relacionales (NoSQL). La clave es que, aunque las consultas de datos aumenten exponencialmente, el rendimiento de Neo4j no desciende. Se realizan las consultas actualizando el nodo y sus relaciones y no todo el grafo completo.

2.- Agilidad: Neo4J tiene muchas ventajas, pero una es su agilidad en la gestión de datos. Si se quisiera llevar al límite sus capacidades, se tendría que superar un volumen total de 34.000 millones de nodos (datos), 34.000 millones de relaciones entre esos datos, 68.000 millones de propiedades y 32.000 tipos de relaciones.

3.- Flexibilidad y escalabilidad: Cuando los desarrolladores de una empresa trabajan con grandes volúmenes de datos, buscan flexibilidad y escalabilidad. Las bases de datos orientadas a grafos aportan mucho en este sentido porque cuando aumentan las necesidades, las posibilidades de añadir más nodos y relaciones a un grafo ya existente son enormes.

4.- Detección del fraude: Neo4j ya trabaja con varias corporaciones en la detección de fraude en sectores como la banca, los seguros o el comercio electrónico. Esta base de datos puede descubrir patrones que con otro tipo de base de datos sería difícil de detectar.

5.- Redes sociales: Neo4j permite conectar de forma eficaz a las personas con productos y servicios, en función de la información personal, sus perfiles

en redes sociales y su actividad online reciente. En este sentido, las bases de datos orientadas a grafos son interesantes porque son capaces de conectar personas e intereses.

6.- Gestión de centros de datos: Las bases de datos gráficas son perfectas ante el crecimiento desbordante de los datos. La gran cantidad de información, dispositivos y usuarios hacen que las tecnologías tradicionales no puedan gestionar tantos datos. La flexibilidad, rendimiento y escalabilidad de Neo4j permite gestionar, monitorizar y optimizar todo tipo de redes físicas y virtuales pese a la gran cantidad de datos [33]**.**

Cypher

Cypher es un lenguaje declarativo que permite una consulta y actualización expresiva y eficiente de un grafo. Está diseñado para ser adecuado tanto para desarrolladores como para profesionales de operaciones. Cypher está diseñado para ser simple, pero poderoso; las consultas de base de datos muy complicadas se pueden expresar fácilmente, lo que le permite concentrarse en su dominio, en lugar de perderse en el acceso a la base de datos.

Cypher se inspira en una serie de enfoques diferentes y se basa en prácticas establecidas para consultas expresivas. Muchas de las palabras clave, como WHERE y ORDER BY, están inspiradas en SQL. Toma prestados enfoques de expresiones de SPARQL. Algunas de las semánticas de la lista están tomadas de lenguajes como Haskell y Python [34] [35].

Librería Graph Algorithms

La librería Graph Algorithms de Neo4j provee algoritmos de grafos expuestos como métodos en Cypher [36]**.**

Los algoritmos de grafos son usados para computar métricas tanto para los grafos, sus nodos o sus relaciones; permiten obtener información sobre entidades relevantes en el grafo (*centrality algorithms*) así como también descubrir estructuras inherentes dentro del grafo, como ser comunidades (*community detection algorithms*).

Los algoritmos de centralidad (*centrality algorithms*) permiten determinar la importancia de los distintos nodos dentro de una red; un ejemplo es el algoritmo PageRank.

Los algoritmos de detección de comunidades (*community detection algorithms*) permiten evaluar cómo un grupo está clusterizado o particionados, así como su tendencia a unirse o separarse. Ejemplos de este tipo de algoritmos son *Louvain, Label Propagation, Weakly Connected Components*.

A continuación, se explican en detalle los dos algoritmos utilizados en el proyecto: *PageRank* y *Label Propagation*.

PageRank

PageRank es un algoritmo que mide la influencia transitiva o conectividad de los nodos. Se puede ejecutar distribuyendo de manera iterativa el grado de un nodo sobre sus nodos vecinos o atravesando de manera aleatoria el grafo contando la frecuencia con la que se alcanza cada nodo durante este recorrido [37].

El algoritmo *PageRank* surge de su utilización en Google para rankear las páginas webs en las búsquedas. Cuenta el número y la calidad de los links a una página los cuáles muestran una estimación de cuán importante es. Asume que las páginas de mayor importancia comúnmente tienen un volumen mayor de links desde otras páginas.

Algunos de los usos que se le da a este algoritmo son por ejemplo en Twitter para recomendar a un usuario otros usuarios que puede estar interesado en seguir; también para catalogar calles y espacios públicos, prediciendo el flujo del tránsito y los movimientos humanos en esas áreas (teniendo en cuenta la tendencia de las personas a estacionar en ciertos lugares, así como los horarios).

- Ejemplo de sintaxis:

```
CALL algo.pageRank(label:String, relationship:String,
    {direction: 'OUTGOING', iterations: 20, dampingFactor: 0.85, write: true, writeProperty: 'pagerank', concurrency: 4})
```

YIELD nodes, iterations, loadMillis, computeMillis, writeMillis, dampingFactor, write, writeProperty

- Parámetros:
    - Label: La etiqueta que se va a cargar del grafo, si es nula carga todos los nodos.
    - Relationship: La relación que se va a cargar del grafo, si es nula se cargan todas las relaciones.
    - Config: Configuraciones adicionales. A continuación, se detallan las utilizadas en el proyecto.
        - *iterations*: Indica cuántas iteraciones ejecutar del algoritmo.
        - *dampingFactor*: Indica el factor de amortiguación del cálculo del algoritmo.
        - *sourceNodes*: Indica los nodos origen desde los cuales realizar los cálculos.
        - *write*: Booleano que al ser seteado a True indica que el resultado se debe escribir como una propiedad del nodo.
        - *writeProperty*: Indica el nombre de la propiedad a la cual se le va a setear el resultado.
        - *weightProperty*: El nombre de la propiedad que contiene el peso, en caso de ser nula toma el grafo como no ponderado.

Label Propagation

Este algoritmo se utiliza para encontrar comunidades dentro de un grafo. Para detectar estas comunidades usa la estructura de la red como guía, sin la necesidad de tener una función objetivo predefinida ni información previa de la comunidad [38].

El algoritmo funciona propagando etiquetas a través de la red y formando comunidades basadas en este proceso de propagación. La idea detrás de este algoritmo es que una única propiedad puede tornarse dominante fácilmente en un grupo de nodos densamente conectados, pero va a tener

problemas en propagarse a través de una región escasamente conectada. Cuando el algoritmo finaliza, los nodos que terminan con la misma etiqueta van a ser considerados parte de la misma comunidad.

El algoritmo funciona de la siguiente manera:

- Cada nodo se inicializa con una única etiqueta que lo identifica.
- Estas etiquetas se propagan a través de la red.
- En cada iteración, cada nodo actualiza esta etiqueta con el valor de la etiqueta a la cual pertenece el mayor número de vecinos.
- El algoritmo termina si converge o si se alcanza el valor máximo de iteraciones definido por el usuario.

Uno de los usos que se le da a este algoritmo es para asignar polaridad a los tweets como parte de un análisis semántico, utilizando etiquetas de un clasificador entrenado para detectar emoticons positivos y negativos.

- Ejemplo de sintaxis:

  CALL algo.labelPropagation(label:String, relationship:String, {iterations:10,

  weightProperty:'weight', writeProperty:'community', write:true, concurrency:4})

  YIELD nodes, iterations, didConverge, loadMillis, computeMillis, writeMillis, write, weightProperty, writeProperty

- Parámetros:
  - Label: La etiqueta que se va a cargar del grafo, si es nula carga todos los nodos.
  - Relationship: La relación que se va a cargar del grafo, si es nula se cargan todas las relaciones.
  - Config: Configuraciones adicionales. A continuación, se detallan las utilizadas en el proyecto.
    - *iterations*: Indica el número máximo de iteraciones que debe ejecutar el algoritmo.

- *write*: Booleano que al ser seteado a True indica que el resultado se debe escribir como una propiedad del nodo.
- *writeProperty*: Indica el nombre de la propiedad a la cual se le va a setear el resultado.

Neovis.js

Neovis.js es una biblioteca que permite embeber en una aplicación web la representación de un grafo a través de javascript. Utiliza el driver javascript de Neo4j (neo4j-js-driver) [39] para conectarse y obtener los datos directo de la base Neo4j. Permite colorear y customizar el grafo basándose en nodos, etiquetas, relaciones, así como utilizando los valores obtenidos de diferentes algoritmos propios de la base de datos, como ser PageRank y Label Propagation.

Para la visualización del grafo utiliza la biblioteca javascript vis.js [40] [41].

1.3.3- MongoDB

MongoDB es un sistema de base de datos NoSQL multiplataforma de licencia libre. Está orientado a documentos de esquema libre, lo que implica que cada registro puede tener un esquema de datos distinto. Es una solución pensada para mejorar la escalabilidad horizontal de la capa de datos, con un desarrollo más sencillo y la posibilidad de almacenar datos con órdenes de magnitud mayores. El modelo de documento de datos (JSON/BSON) pretende ser fácil de programar, fácil de manejar y ofrece alto rendimiento mediante la agrupación de los datos relevantes entre sí. En MongoDB un conjunto de Campos formará un Documento, que en caso de asociarse con otros formará una Colección. Las bases de datos estarán formadas por Colecciones, y a su vez, cada servidor puede tener tantas bases de datos como el equipo lo permita [42].

Para el intercambio de datos para almacenamiento y transferencia de documentos en MongoDB se usa el formato BSON, (Binary JavaScript Object Notation). Se trata de una representación binaria de estructuras de datos y mapas, diseñada para ser más ligera y eficiente que JSON (JavaScript Object Notation).

MongoDB cuenta con una serie de herramientas que permiten trabajar con la base de datos desde diferentes perspectivas y tratar con ella para diferentes propósitos:

- Mongod: Servidor de bases de datos de MongoDB.
- Mongo: Cliente para la interacción con la base de datos MongoDB.
- Mongofiles: Herramienta para trabajar con ficheros directamente sobre la base de datos MongoDB.

MongoDB utiliza el Sharding [43] (escalado horizontal) de forma automática como método para dividir los datos a lo largo de los múltiples servidores. Una vez se ha indicado a MongoDB que se va a distribuir la base de datos, añadiendo un nuevo nodo al clúster, éste se encarga de balancear los datos, (distribuir los datos de forma eficiente entre todos los equipos), entre los servidores de forma automática. Los objetivos del Sharding en MongoDB pasan por:

1. Un clúster transparente: Para satisfacer esto MongoDB hace uso de un proceso de enrutamiento, al que se denomina mongos. Los mongos se colocan frente al clúster, y de cara a cualquier aplicación, como si estuviera delante de un servidor individual MongoDB. Mongos reenvía la consulta al servidor o servidores correctos, y devuelve al cliente la respuesta que éste ha proporcionado.

2. Disponibilidad para lecturas y escrituras: Siempre que no existan causas de fuerza mayor, (como caída masiva de la luz), debemos asegurar la disponibilidad de escritura y lecturas en nuestra base de datos. Antes de que se degrade excesivamente la funcionalidad del sistema, el clúster debe permitir fallar tantos nodos como sean necesarios. MongoDB hace que algunos de los procesos del clúster tengan redundancia en otros, de forma que, si cae un nodo, otro puede continuar con estos procesos.

3. Crecimiento ágil: El clúster debe añadir o reducir capacidad cuando lo necesite.

MongoDB, es más flexible que las bases de datos relacionales, y por ello menos restrictivo, lo que puede presentar en ocasiones problemas de volatilidad. Para esto existe la funcionalidad de réplica, MongoDB manda los

documentos escritos a un servidor maestro, que sincronizado a otro u otros servidores mandará esta misma información replicada, a estos *esclavos*.

MapReduce es una utilidad para el procesamiento de datos por lotes y operaciones de agregación. Está escrita en JavaScript y se ejecuta en el servidor. Permite utilizar funciones de agregación de datos, que de forma original sería complicado de tratar debido a la ausencia de esquemas típicos de estas soluciones. El proceso consta de dos partes, por un lado, el mapeo, y por otro la reducción. El mapeo transforma los documentos y emite un par (clave/valor). La función reduce obtiene una clave y el array de valores emitidos para esa clave y produce el resultado final.

2- Arquitectura

Para modelar el sistema se definieron cuatro componentes, cada uno en un contenedor Docker separado:

- Base de datos de Neo4j (1.3.2), donde se persisten los tweets, usuarios de Twitter y sus relaciones.
- Base de datos de MongoDB (1.3.3), donde se persiste el json del tweet, datos de usuario final (login, historial de métricas, pesos).
- Componente *FakeBusters Server*, encargado de realizar la comunicación con la API de Twitter, obtener los tweets (en formato json) y guardarlos en las diferentes bases de datos, MongoDB y Neo4j. A través de una interfaz web, implementada con el framework Flask, permite al usuario final obtener tweets y seguidores de un usuario específico, así como también obtener tweets por streaming.
- Componente *FakeBusters Web*, hace de interfaz con el usuario final. Por medio del framework Flask se comunica con ambas bases de datos para obtener datos de tweets y de usuarios de Twitter y calcular métricas. También permite guardar y obtener datos propios del usuario final, como ser historial de métricas o pesos. Por medio de la biblioteca Neovis.js se comunica con la base de datos de Neo4j para dibujar los grafos asociados a las métricas.

A continuación, en la Figura 5, se muestra un diagrama de la arquitectura del sistema:

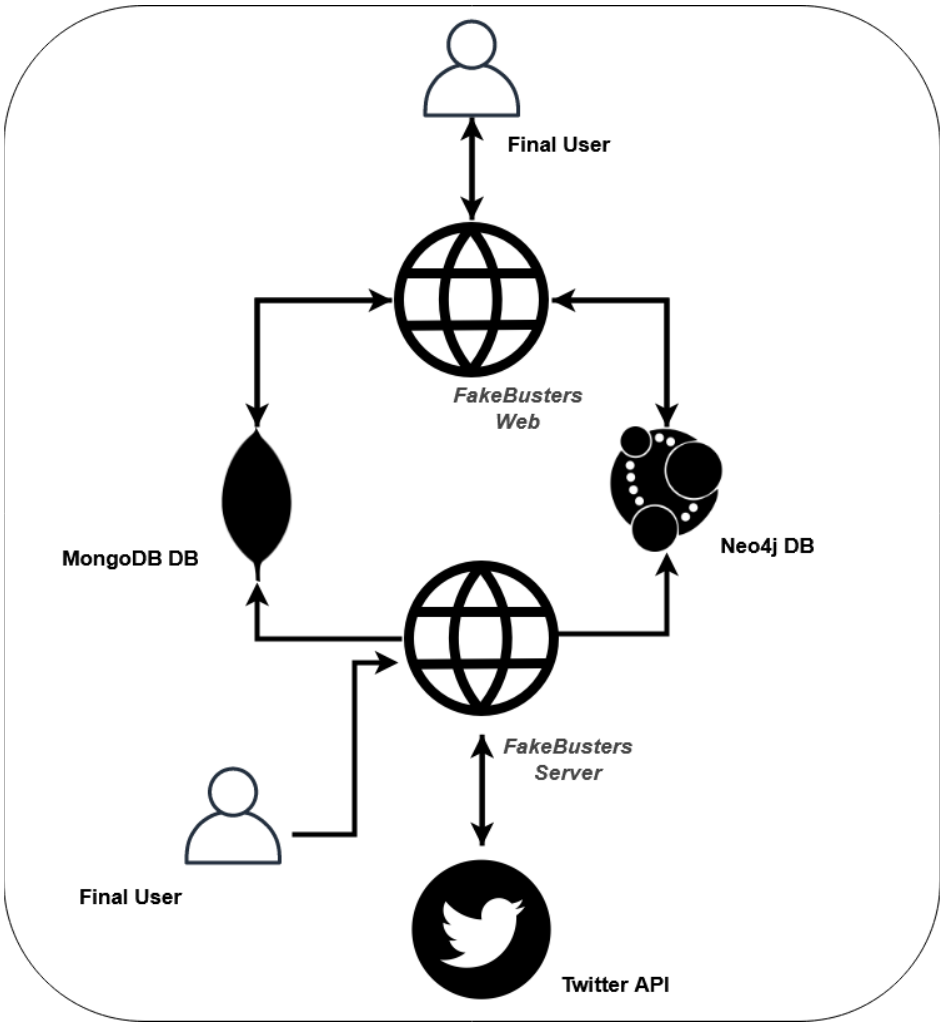

Figura 5. Arquitectura del sistema

3- Datos

Los datos de entrada para el cálculo de las métricas se obtienen haciendo uso del componente *FakeBusters Server*. Los mismos son obtenidos de Twitter y persistidos en las dos bases de datos no relacionales, para su posterior uso. Podemos obtener tweets en general, tweets asociados a un usuario y los tweets de los seguidores de un usuario.

3.1- Obtención de Datos

En esta sección presentaremos las diferentes formas de obtención de datos implementadas.

3.1.1- Tweets Streaming

Para la obtención de datos se decidió utilizar la búsqueda por streaming que brinda la API de Twitter. La misma permite obtener tweets en tiempo real que cumplan con cierto filtro de palabras definido previamente [44].

Se utilizó la biblioteca Twython que provee una manera fácil de acceder a los datos de Twitter utilizando Python [45].

Al momento de obtener un tweet se realiza la búsqueda de seguidores y demás tweets publicados por el usuario autor de este; para de esta manera ampliar la base de datos y obtener métricas más cercanas a la realidad.

3.1.2- Tweets Asociados a un Usuario

Hay casos en los que es muy útil obtener los tweets de un usuario en específico, para esto implementamos la función *load_tweets_user*.

Esta función devuelve los tweets de un usuario por medio de su screen_name (nombre de usuario) y no por streaming; para ello se utiliza la función *get_user_timeline* de Twython.

3.1.3- Tweets de Seguidores de un Usuario

Podemos obtener los tweets de los seguidores de un usuario usando la función *load_followers_tweets_user*. La misma permite a partir del screen_name obtener los datos de sus seguidores (tweets, followers, hashtags, etc) con un cierto nivel de profundidad que indica si se desea obtener también los datos de los seguidores de estos últimos y así sucesivamente. Por ejemplo, para el caso de un nivel 0 se obtienen los datos de los seguidores del usuario indicado; en el caso de un nivel igual a 1 se obtienen los datos de los seguidores del usuario y para cada seguidor de éste se obtienen los datos de sus seguidores. Para la implementación de esta funcionalidad se utiliza también la función *get_user_timeline* de Twython.

3.2- Persistencia de Datos

Para la persistencia se utilizan dos bases de datos no relacionales: MongoDB y Neo4j, explicadas en detalle en las secciones 1.3.3 y 1.3.2 respectivamente, con la intención de comparar las mismas y ver cómo se comportan en el cálculo de las diferentes métricas.

Se elige utilizar la base de datos Neo4j porque es intrínseco a las redes sociales, es una representación directa de las redes sociales y de las métricas. Por otro lado, MongoDB es mucho más sencilla de mapear y manejar, permite realizar consultas complejas de forma más sencilla, con todas las ventajas de performance de una base de datos no relacional.

A medida que se obtienen los datos de los tweets se persiste en una base en MongoDB el json obtenido, únicamente con los tags previamente descritos. Al mismo tiempo se persiste en una base en Neo4j siguiendo la estructura de nodo Usuario, nodo Tweet, nodo Hashtag, así como las demás relaciones. De esta manera se obtiene una representación visual de los datos, así como también permite calcular propiedades entre relaciones de usuarios, como ser distancias, distinguir cuán cercanos o intercomunicados están los usuarios.

Para mantener la consistencia de los datos, los nodos se generan solo si no existen previamente, de lo contrario se actualizan los datos de estos.

En síntesis, se obtienen los tweets utilizando la API de Twython *Streaming* [46], que consume el rest *statuses/filter* [47] de la API de Twitter. Para cada tweet obtenido mediante streaming, se genera:

- Un json con la metadata relativa al tweet: Usuario que lo publicó, texto del tweet, fecha, localización, cantidad de retweets, cantidad de me gusta, hashtags agregados, usuarios mencionados. En caso de ser un retweet o una cita, contiene también los datos relativos al usuario original y al tweet original.
- El json se guarda en MongoDB.
- Se genera el nodo del tweet con sus datos, el nodo Usuario con sus datos, los diferentes nodos Hashtags; y se los agrega a la base de Neo4j.

- Si es un retweet o una cita se agrega el nodo Tweet con los datos del tweet original, el nodo Usuario con los datos del usuario que generó el tweet original; las relaciones: *RetweetedFromTweet*, *RetweetedFromUser*, si es un retweet, que relaciona ambos tweets y ambos usuarios respectivamente; o las relaciones *QuotedFromTweet* y *QuotedFromUser* que relacionan ambos tweets y ambos usuarios respectivamente.
- Se agrega la relación *PostedBy* que relaciona el tweet con el Usuario que lo creó.
- Si el tweet tiene hashtags se agrega la relación *HashtagFromTweet* que relaciona cada hashtag con el tweet.
- Si el tweet menciona a otros usuarios se agrega la relación *MentionToUser* que relaciona el tweet con los usuarios mencionados.
- Para el usuario que posteó el tweet se obtienen sus seguidores, por medio de la función *get_followers_list* de Twython. Esta función consume el servicio rest *followers/list* [48] de la API de Twitter, la cual retorna en grupos de 20 usuarios por vez los seguidores del usuario por *Id* de usuario. Se crea el nodo usuario para cada uno de ellos y se crea la relación *FollowedBy* que relaciona ambos.
- Se obtienen demás tweets posteados por el usuario, por medio de la función *get_user_timeline* de Twython, la cual consume el rest *statuses/user_timeline* [49] de la API de Twitter. Retorna como máximo los 3.200 tweets posteados por el usuario, siempre que el mismo no sea un usuario protegido. Se crea un nodo tweet para cada uno de ellos y se los relaciona con el usuario por medio de la relación *PostedBy*.

A continuación (Figura 6) se detalla el procedimiento de la obtención y posterior persistencia de los datos obtenidos de Twitter.

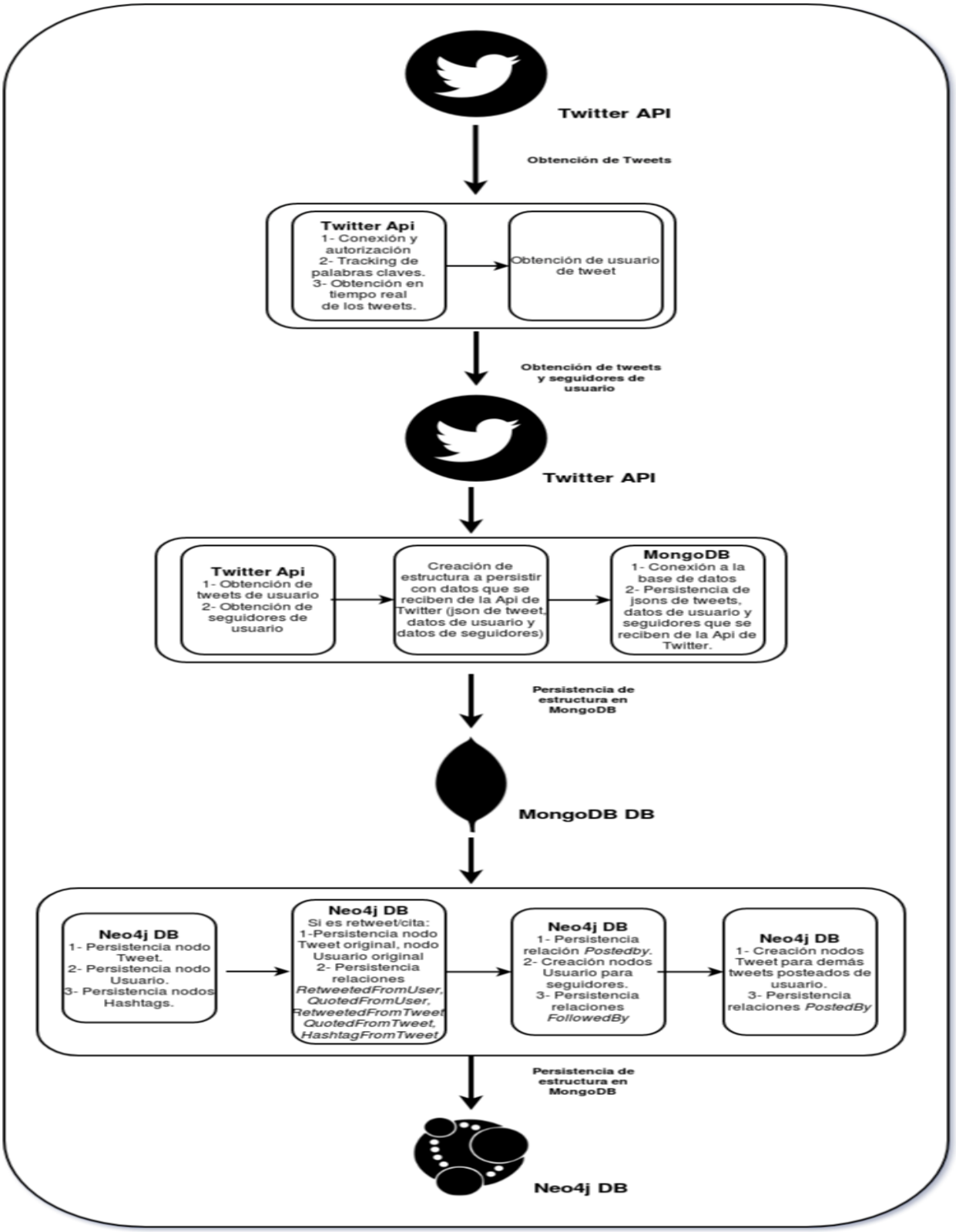

Figura 6. Diagrama de flujo de persistencia de los datos.

4- Cálculo de Valores de Normalización

Para normalizar las métricas se deben tomar ciertos valores de referencia. Algunos de estos valores son constantes fijadas en base a algún criterio y otros son dependientes del conjunto de datos obtenidos hasta el momento.

En la siguiente tabla (Tabla 19) se muestran los mismos y su cálculo en caso de que corresponda.

Nombre	Consulta MongoDB	Descripción
followers_avg_set	`tweets.aggregate([{'$group': {'_id':0,'followers_avg_set': {'$avg': "$user.user_followers_count"}}}])`	Valor promedio de seguidores que puede tener un usuario basándose en los datos recolectados hasta el momento.
tweets_retweets_pub_avg_set	`tweets.aggregate([{'$group': {'_id':0,'tweets_retweets_pub_avg_set': {'$avg': "$user.user_statuses_count"}}}])`	Valor promedio de tweets y re-tweets realizados por el usuario teniendo en cuenta la variable *statuses_count* del usuario, basándose en los datos recolectados hasta el momento.

statuses_avg_set	`commentedTweets = tweets.find({"tweet.tweet_in_reply_to_status_id": {'$ne': None}}).count(True)`
	`users_set = len(list(tweets.distinct('user.user_name')))`
	`status_avg_set = commentedTweets/users_set`
		Obtiene la cantidad de tweets del usuario que son respuestas a otro tweet (se identifica por tener el campo *tweet_in_reply_to_status_id* no nulo) y lo promedia con respecto a la cantidad total de tweets del usuario. Ambos valores se basan en los datos recolectados hasta el momento

statuses_max_ref	Se setea y se obtiene de la tabla parámetros	Valor máximo de tweets y re-tweets realizados por el usuario basándose en valores de referencia.
twitter_foundation_date	Se setea y se obtiene de la tabla parámetros	Fecha de fundación de Twitter, 21 de marzo de 2006.
avg_quote_count	`tweets.aggregate([{'$group': {'_id':0, 'quote_coun_avg': {'$avg': "$tweet.tweet_quote_count"}}}])`	Valor promedio de citas que tiene un tweet, basándose en los datos recolectados hasta el momento.
avg_favourite	`tweets.aggregate([{'$group': {'_id':0, 'favourites_count_avg': {'$avg': "$user.user_favourites_count"}}}])`	Valor promedio de me gustas que realice un usuario, basándose en los datos recolectados hasta el momento.
avg_favourite_count	`tweets.aggregate([{'$group': {'_id': 0, 'favouriteCount': {'$sum': "$tweet.tweet_favorite_count"}}}])`	Valor promedio de me gustas que tiene un tweet, basándose en los datos recolectados hasta el momento.
avg_retweets_count	`tweets.aggregate([{'$group': {'_id': 0, 'retweets_count_avg': {'$avg': "$tweet.tweet_retweet_count"}}}])`	Valor promedio de retweets que tiene un tweet, basándose en los datos recolectados hasta el momento.

Tabla 19. Constantes.

5- Cálculo de Métricas

Para el cálculo de las diferentes métricas se realiza la consulta en ambas bases de datos como muestra la siguiente tabla (Tabla 20).

Métrica	Consulta MongoDB	Consulta Neo4j
followers_quantity	agr = [{'$match': {"user.user_name": name}},{'$group': {'_id': 0, 'user_followers_count_max': {'$max': "$user.user_followers_count"}}}] userFollowersCountMax = list(tweets.aggregate(agr))	MATCH (n:TwitterUser {name:$name}) RETURN n.followers_count
tweets_and_retweets_published	agr = [{'$match': {"user.user_name": name}},{'$group': {'_id': 0, 'user_statuses_count_max': {'$max': "$user.user_statuses_count"}}}] userStatusesCountMax = list(tweets.aggregate(agr))	MATCH (n:TwitterUser {name:$name}) RETURN n.statuses_count
favorite	agr = [{'$match': {"user.user_name": name}},{'$group': {'_id': 0, 'user_favourites_count_max': {'$max': "$user.user_favourites_count"}}}] userFavouritesCountMax = list(tweets.aggregate(agr))	MATCH (n:TwitterUser {name:$name}) RETURN n.favourites_count
antiquity	agr = [{'$match': {"user.user_name": name}},{'$group': {'_id': 0, 'user_creation_date': {'$max': "$user.user_creation_date"}}}] antiquity = list(tweets.aggregate(agr))	MATCH (n:TwitterUser {name:$name}) RETURN n.creation_date
verified	agr = [{'$match': {"user.user_name": name}},{'$group': {'_id': 0, 'user_verified': {'$max': "$user.user_verified"}}}] verified = list(tweets.aggregate(agr))	MATCH (n:TwitterUser {name:$name}) RETURN n.verified
commented_tweets	commentedTweets = tweets.find({"$and": [{"user.user_name": name}, {"tweet.tweet_in_reply_to_status_id":{'$ne':None}}]}).count(True)	MATCH (t:Tweet)-[r:POSTED_BY]->(u:TwitterUser{name:$name}) return count(t.in_reply_to_status_id)
quote_Count	agr = [{'$match': {"user.user_name": name}},{'$group': {'_id': 0, 'quoteCount': { '$sum': "$tweet.tweet_quote_count" }}}] quoteCount = list(tweets.aggregate(agr))	MATCH (t:Tweet)-[r:POSTED_BY]->(u:TwitterUser{name:$name}) return sum(t.quote_count)
retweets_count_user	agr = [{'$match': {"user.user_name": name}},{'$group': {'_id': 0, 'retweetCount': { '$sum': "$tweet.tweet_retweet_count" }}}] retweetCount =	MATCH (t:Tweet)-[r:POSTED_BY]->(u:TwitterUser{name:$name}) return sum(t.retweet_count)

	list(tweets.aggregate(agr))	
favourite_count_user	agr = [{'$match': {"user.user_name": name}},{'$group': {'_id': 0, 'favouriteCount': { '$sum': "$tweet.tweet_favorite_count" }}}] favouriteCount = list(tweets.aggregate(agr))	MATCH (t:Tweet)-[r:POSTED_BY]->(u:TwitterUser{name:$name}) return sum(t.favorite_count)
message_invariability	tweets.find_one({"_id":tweet_id}, {'tweet.tweet_text': 1, '_id': 0})	session.write_transaction(self.get_messages_tweet, tweet_id)
followers_quality	userFollowers= tweets.find({'user.user_name':name}, {'followers': 1, '_id': 0}) userFollowers = list(userFollowers)	userFollowers= tweets.find({'user.user_name':name}, {'followers': 1, '_id': 0}) userFollowers = list(userFollowers)
similar_words	tweets.find({"user.user_name":name},{"tweet.tweet_text": 1}) list(tweetsByUser)	with self. driver.session() as session: tweets=session.write_transaction(self.get_messages_tweet_user_name, user_name)
same_location	tweet_place = tweets.find_one({"_id": tweet_id}, {'tweet.tweet_place': 1, '_id': 0}) tweet_coordinates = tweets.find_one({"_id":tweet_id}, {'tweet.tweet_coordinates': 1, '_id': 0}) if tweet_place and tweet_coordinates and tweet_place['tweet']['tweet_place'] and tweet_coordinates['tweet']['tweet_coordinates']: same_location = tweet_place['tweet']['tweet_place'] == tweet_coordinates['tweet']['tweet_coordinates']	session.write_transaction(self._query_get_tweet_place_by_tweet_id,tweet_id)== session.write_transaction(self._query_get_tweet_coordinates_by_tweet_id,tweet_id)

Tabla 20 Consultas para cálculo de métricas.

Se normalizan los valores a través de las constantes y luego se muestran los datos utilizando la biblioteca de Python Flask (1.2.1); donde se exponen webservices para cada una de ellas.

Las tres últimas métricas no basan su lógica en las consultas realizadas, por ende, ameritan una explicación más a detalle, la cual se brinda a continuación.

followers_quality

Obtiene todos los seguidores de un usuario usando las consultas antes especificadas y para cada seguidor, calcula su calidad en base a **followers_quantity, tweets_retweets_published, commented_tweets, favorite, quote_count, retweets_count_user, favorite_count_user, antiquity, verified, similar_words** y los pesos seteados en ese momento. Para la normalización de la calidad de cada seguidor se utiliza el promedio ponderado, es decir se divide entre la suma de los pesos de cada una de las métricas utilizadas.

```
followers =getFollowersUser(user_name)

quality_followers = 0

count = 0

if followers is not None:

   for f in followers:

       user_foll_name = f['user']['user_name']

        quality_followers = quality_followers + float(quality_user(user_foll_name,
option,flag_get_save))

       count = count + 1

quality_followers = quality_followers/ count
```

---

similar_words

Obtiene todos los tweets de un *user_name* utilizando la consulta antes mencionada y compara los vectores de palabras de los mismos, para esto utiliza dos funciones `getModelWords()` y `similar_vector()`, las cuales se explicaran a detalle más adelante.

```
tweet_usu = getMessagesTweetUser(user_name)

model = {}

for i in range(0, tweet_usu.len):

   if i not in model.keys():
```

```
        model[i] = getModelWords(tweet_usu[i])
    for j in range(i+1, len):
        count = count + 1
        if j not in model.keys():
            model[j] = getModelWords(tweet_usu[j])
        per_similar = similar_vector(model[i],model[j])
        similar_words = per_similar + similar_words
if count > 0:
    similar_words = similar_words / count
else:
    similar_words = 0
similar_words = round(similar_words/100,3)
```

---

reputation_path

Obtiene el texto del tweet origen y el texto del tweet destino y compara los vectores de palabras de los mismos, utilizando `getModelWords()` y `similar_vector()` al igual que *similar_words*.

```
msj_source = getMessagesTweet(tweet_id_source)
m1 = getModelWords(msj_source)
msj_target = getMessagesTweet(tweet_id_target)
m2 = getModelWords(msj_target)
if msj_target!='' and msj_source !='':
    reputation_path = similar_vector(m1,m2)
else:
    reputation_path = 0
```

---

similar_vector y getModelWords

Para comprar los textos de dos tweets se utilizan técnicas de PLN y realizan los siguientes pasos:

- Preprocesamiento de texto:

  Los textos a comparar pueden tener varios tipos de texto ruidoso. Esto debe limpiarse para poder realizar más tareas de PLN. Para esto se utiliza la función preprocess(), en una primera instancia se obtiene todo lo que sean palabras, se eliminan los links, números, emojis y signos de puntuación. Luego se pasa todo el texto a minúsculas y se eliminan las stopwords.[50]

```
def preprocess(raw_text):

# keep only words

letters_only_text = re.sub("[^a-zA-Z]", " ", raw_text)

# Eliminamos link

text=re.sub(
r'''(?i)\b((?:https?://|www\d{0,3}[.]|[a-z0-9.\-]+[.][a-z]{2,4}/)(?:[^\s()<
>]+|\(([^\s()<>]+|(\([^\s()<>]+\)))*\))+(?:\(([^\s()<>]+|(\([^\s()<>]+\)))*
\)|[^\s`!()\[\]{};:'".,<>?«»“”‘’]))''',"", letters_only_text)

# Eliminamos emojis

text = ''.join(c for c in unicodedata.normalize('NFC', text) if c <=
'\uFFFF')

# Eliminamos los signos de puntuación

text = re.sub('[%s]' % re.escape(string.punctuation), ' ', text)

# convert to lower case and split

words = text.lower().split()

# remove stopwords english

stopword_set=set(stopwords.words("english")).union(set(stopwords.words("spa
nish"))

cleaned_words = list(set([w for w in words if w not in stopword_set]))

return cleaned_words
```

- Codificación de Texto:

  Para comparar los textos es necesario que estén en formato numérico. Se están utilizando ampliamente varias técnicas de codificación para extraer las incrustaciones de palabras de los datos de texto, tales técnicas son bolsa de palabras, TF-IDF, word2vec [50]. Para este proyecto se cargan dos modelos de datos uno de palabras en inglés y otro en español en MongoDB, con los valores ya definidos. Mediante la función *getModelWords*() se extraen los valores necesarios para los vectores de palabras ya preprocesados.

```
def getModelWords(msj):
    msj = preprocess(msj)
    for word in msj:
       data_es =getModelWord(word,'es')
       if data_es is not None:
         model['es'][word] = data_es
       data_en = getModelWord(word, 'en')
       if data_en is not None:
         model['en'][word] = data_en
    return model
```

- Similitud de vectores:

  Para calcular la similitud entre dos textos se usan métodos estadísticos. En este proyecto se utiliza *Cosine Similarity*, el mismo es el método más utilizado para comparar dos vectores. Consiste en el producto escalar entre dos vectores, se halla el ángulo coseno entre los dos vectores, para el grado 0 el coseno es 1 y es menor que 1 para cualquier otro ángulo [50]. La implementación de este se encuentra en la función cosine_distance_wordembedding_method().

```
def similar_vector(self,m1,m2):
   return self.cosine_distance_wordembedding_method(self,m1,m2)


def cosine_distance_wordembedding_method(self,s1_model, s2_model):
#Espanol
   if not s1_model['es'] == {}:
       vector_1=np.mean([s1_model['es'][word] for word in s1_model['es'].keys()],
axis=0)


   if not s2_model['es'] == {}:
       vector_2=np.mean([s2_model['es'][word] for word in s2_model['es'].keys()],
axis=0)


   if vector_1 is not None and vector_2 is not None:
      cosine = scipy.spatial.distance.cosine(vector_1, vector_2)
      val_1 = (round((1 - cosine) * 100, 3))


#Ingles
   if not s1_model['en'] == {}:
       vector_1=np.mean([s1_model['en'][word] for word in s1_model['en'].keys()],
axis=0)
   if not s2_model['en'] == {}:
       vector_2=np.mean([s2_model['en'][word] for word in s2_model['en'].keys()],
axis=0)
   if vector_1 is not None and vector_2 is not None:
     cosine = scipy.spatial.distance.cosine(vector_1, vector_2)
     val_2 = (round((1 - cosine) * 100, 3))
```

```
#Promedio
    return (val_1 + val_2)/2
```

- Función de decisión:

  A partir del valor de similitud, se debe definir una función personalizada para decidir si dos textos son similares o no. Cosine Similarity devuelve un valor entre 0 y 1, en el cual 1 significa exactamente similar y 0 nada similar. Para este proyecto lo que se necesita es saber el rango de similitud, y no definir si son similares o no, por ende, se hace un promedio del resultado para las palabras en español e inglés y directamente se utiliza este valor.

6- Funcionalidades

En esta sección se muestran las diferentes funcionalidades que permite realizar el prototipo implementado. El mismo se compone de dos aplicaciones web, ***FakeBusters Server*** y ***FakeBusters Web***. La versión Server permite obtener la información de Twitter en base a ciertos criterios y guardarla en las bases de datos, para luego poder trabajar con la misma en el cálculo de las métricas. La versión Web permite el cálculo de las diferentes métricas implementadas, así como también la visualización de los resultados y la información de tweets guardadas en las bases.

6.1- FakeBusters Server

En ***FakeBusters Server*** (accesible desde el puerto 8080) se encuentran disponibles tres funcionalidades relativas a la obtención de los datos a través de la API de Twitter: *Get tweets by user*, *Get Followers tweets by user*, *Get tweets by streaming*.

En el anexo, sección 4.1 se explica en detalle cada una de estas funcionalidades.

6.2- FakeBusters Web

Se realizó una aplicación web denominada ***FakeBusters Web*** (accesible desde el puerto 5000) en la cual se pueden encontrar las métricas de Usuario y Clip descritas anteriormente y calcularlas por separado tanto en MongoDB,

como en Neo4j. Por otro lado, también se pueden combinar las mismas y asignarles un peso y una base de ejecución específicas, para así obtener una medida de credibilidad general.

Como funcionalidades complementarias, se puede ver el historial de las métricas calculadas, tanto específicas como generales, cambiar algunos parámetros que se necesitan para el cálculo de métricas y obtener información de tweets guardados en la base de datos usando diferentes criterios.

En el anexo, sección 4.2 se explica en detalle cada una de estas funcionalidades.

Capítulo 6 Caso de estudio

En esta sección se presentará un análisis de los datos obtenidos al hacer uso de la herramienta.

Como caso de estudio se decidió basarse en la temática de la pandemia COVID-19, dado que al estar afectando a nivel mundial en la actualidad existe una gran cantidad de datos relativos al tema fluyendo constantemente.

1- Obtención de Datos

Se definieron un conjunto de palabras claves relativas al COVID-19 para utilizar en la búsqueda por streaming y de esa manera poder generar una base con datos sesgados al COVID-19, así como también se acotó la búsqueda a tweets localizados en Uruguay y escritos en lenguaje español.

A continuación, en la Tabla 21 se muestran los parámetros que fueron utilizados para obtener los datos de Twitter mediante streaming con sus respectivos valores.

Parámetro	Valores
languages	Es
locations	30,53,38,58
track	coronavirus, virus, gripe, síntomas, enfermedades respiratorias, pandemia, Organización Mundial de la Salud, OMS, covid-19, SARS-CoV-2

Tabla 21 Parámetros para la obtención de tweets por streaming.

Se trabajó sobre una base de datos con un total de 114.662 entidades, de los cuales 61885 corresponden a tweets, 48673 a usuarios y 4104 a hashtags.

De los 61885 tweets, 24217 son retweets y 2351 son citas. Dentro de los 48673 usuarios 33641 son seguidores.

2- Definición de Parámetros y Pesos

Para obtener el valor de la métrica global *general_credibility* es necesario definir para cada una de las métricas que compone el cálculo un peso con el cual ponderarla en la suma total.
Como primera aproximación se definieron los pesos en base a los conocimientos obtenidos a partir del marco teórico, para luego ajustarlos en base a pruebas concretas.

Dado que tenemos dos bases sobre las cuales calcular las métricas, usamos las características particulares de cada una para definir y así optimizar el cálculo de cada métrica. La decisión se tomó en base a cómo se obtiene el valor de la métrica: si para su cálculo se necesitan datos correspondientes a relaciones con otros tweets y/o usuarios se utiliza la base de datos Neo4j, la cual es más eficiente ya que tiene mapeadas las relaciones; en caso contrario se calcula utilizando la base de datos MongoDB.

A continuación (Tabla 22) se detalla para cada métrica los valores utilizados de peso final y base de dato:

Metrics	Weights	DataBase
followers_quantity	0,700	MongoDB
tweets_retweets_pub	0,800	MongoDB
favourite	0,700	MongoDB
antiquity	0,800	MongoDB
verified	1,000	MongoDB
commented	0,800	Neo4j
quote_count	0,000	MongoDB
retweets_count	0,800	MongoDB
favourite_count	0,700	MongoDB
followers_credibility	0,500	Neo4j

similar_words	0,500	Neo4j
same_location	0,500	Neo4j
reputation_path	0,300	MongoDB

Tabla 22. Base de datos y pesos utilizados en cada métrica.

Para el cálculo de la métrica *antiquity* es necesario definir el parámetro *Twitter Foundation Date*; para las pruebas se utilizó la fecha real de la fundación de Twitter (21/03/2006).

Se tomó 1000 como valor máximo de referencia de comentarios (*Statuses Max Ref*), parámetro utilizado en la métrica *commented*.

Con el fin de concluir si un usuario resulta creíble o no se definió el parámetro *Credibility Range* a 0,5; es decir, si la métrica general *Credibility* da un resultado mayor o igual a 0,5 el usuario resulta creíble y no creíble en caso contrario.

3- Elección de Datos de Estudio

Con el fin de validar las métricas calculadas se realizó una búsqueda manual en las respectivas bases de datos, así como en Twitter, de usuarios y tweets que cumplen ciertas particularidades, como ser usuarios reales (usuarios oficiales, validados por Twitter), usuarios catalogados como bots, tweets cuyo contenido fuera una noticia falsa y sus respectivos usuarios. Se obtuvo una lista con siete usuarios y tweets a utilizar para el cálculo de las distintas métricas, a continuación, se detalla en una tabla las características de cada uno de ellos (Tabla 23).

Se utilizan nombres de usuarios ficticios para proteger la privacidad de estos.

Descripción Caso	Usuario	Tweet
Usuario Real **Tweet Creíble**	**usuario1**	**1327257932732502016**
Usuario Real	**usuario2**	

Tweet Creíble		**1327390306405404678**
Usuario Bot	**usuario3**	**1342991271351226374**
Usuario Poco Creíble Noticia Falsa	**usuario4**	**1328100611418681347**
Usuario Real	**usuario5**	**1318364161190092802**
Usuario Real	**usuario6**	**1329133123301494784**
Usuario Real	**usuario7**	**1345354301670100993**

Tabla 23. Usuarios y tweets seleccionados para cálculo de métricas.

A continuación, se detalla para cada uno de los usuarios los datos almacenados en las respectivas bases de datos:

3.1- Usuario1

- <id>: 88384
- community: 47062
- creation_date: Tue Sep 27 14:31:51 +0000 2011
- description: Cuenta oficial de la Presidencia de la República Oriental del Uruguay Norma de convivencia: https://t.co/Dy6AbUjzrb
- favourites_count: 5422
- followers_count: 139351
- friends_count: 151
- id: 380961080
- listed_count: 541
- location: Montevideo, Uruguay

- name: usuario1
- real_name: xxx
- statuses_count: 42122
- url: http://t.co/2hHWnIE1MZ
- verified: true
- Cantidad de tweets en base: 63
- Cantidad de Followers en base: 55

A continuación, en la Figura 18, se muestra la representación del grafo del usuario:

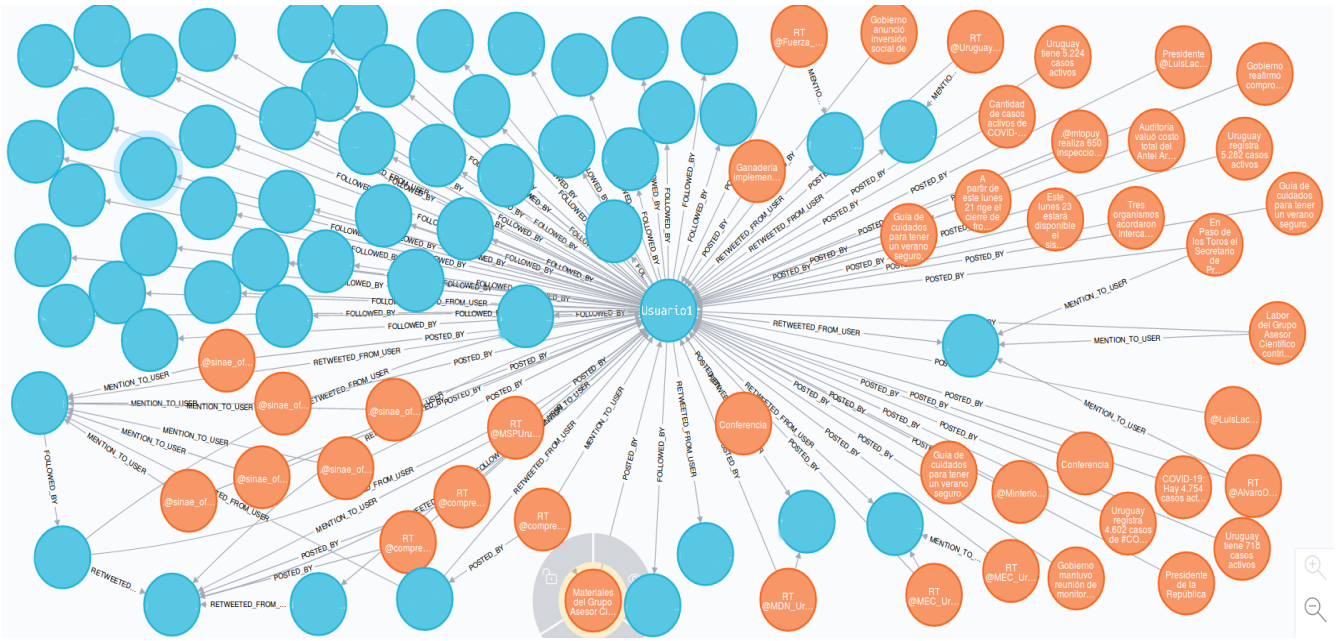

Figura 18. Representación del grafo del usuario1 con sus respectivos seguidores (azul) y tweets (anaranjado).

Tweet utilizado:

- <id>: 108860
- community: 108860
- coordinates: None

- created_at: Fri Nov 13 14:32:05 +0000 2020
- favorite_count: 141
- hashtags: []
- id: 1327257932732502016
- retweet_count: 59
- text: Materiales del Grupo Asesor Científico Honorario están disponibles en sección especial de portal de Presidencia https://t.co/8mJD3gopGy
- user_mentions: []

En la Figura 19 se muestra la representación del grafo del tweet:

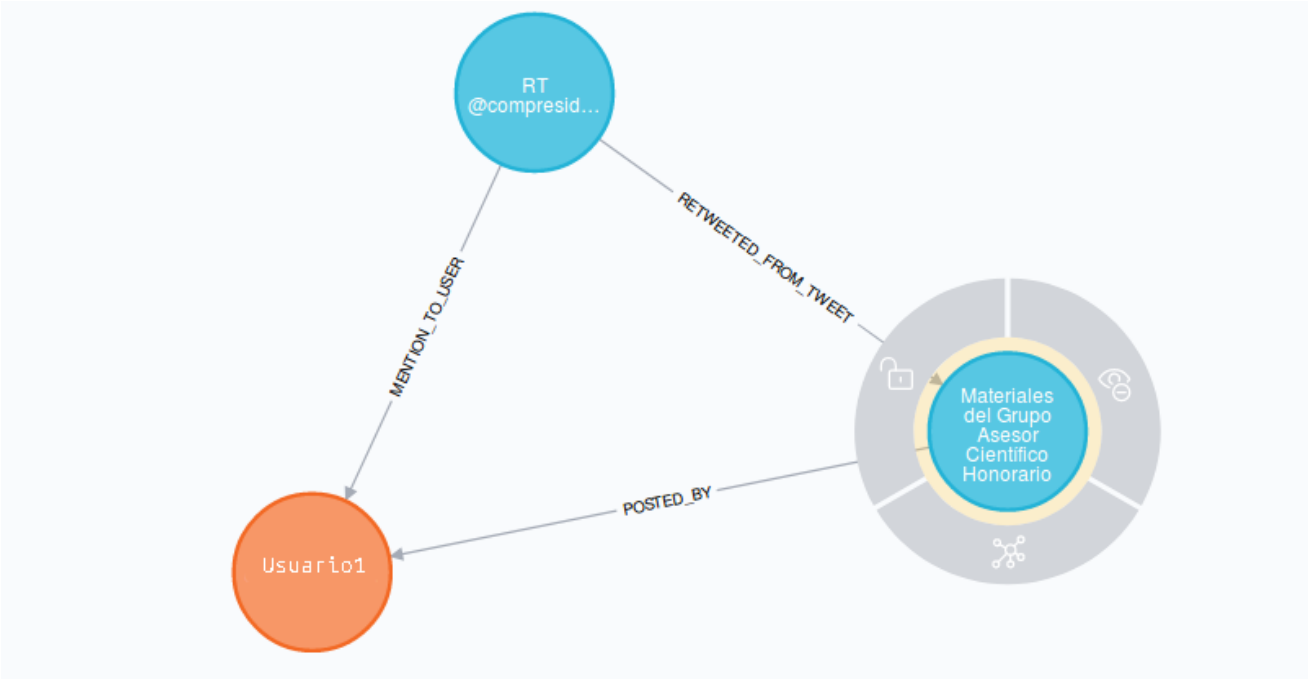

Figura 19. Representación del grafo del tweet de usuario1 con sus respectivas relaciones.

3.2- Usuario2

- <id>: 88373
- community: 22172

- ❏ creation_date: Sat Sep 22 19:42:14 +0000 2012
- ❏ description: Sistema Nacional de Emergencias. Promoviendo hábitos que nos ayuden en los momentos oportunos. Convivencia en redes: https://t.co/fIMVRyVlPC
- ❏ favourites_count: 621
- ❏ followers_count: 74037
- ❏ friends_count: 955
- ❏ id: 840325920
- ❏ listed_count: 147
- ❏ location: Uruguay
- ❏ name: usuario2
- ❏ real_name: xxx
- ❏ statuses_count: 14959
- ❏ url: http://t.co/1qLLTWKZ6d
- ❏ verified: true
- ❏ Cantidad de tweets en base: 43
- ❏ Cantidad de Followers en base: 52

A continuación, en la Figura 20, se muestra la representación del grafo del usuario:

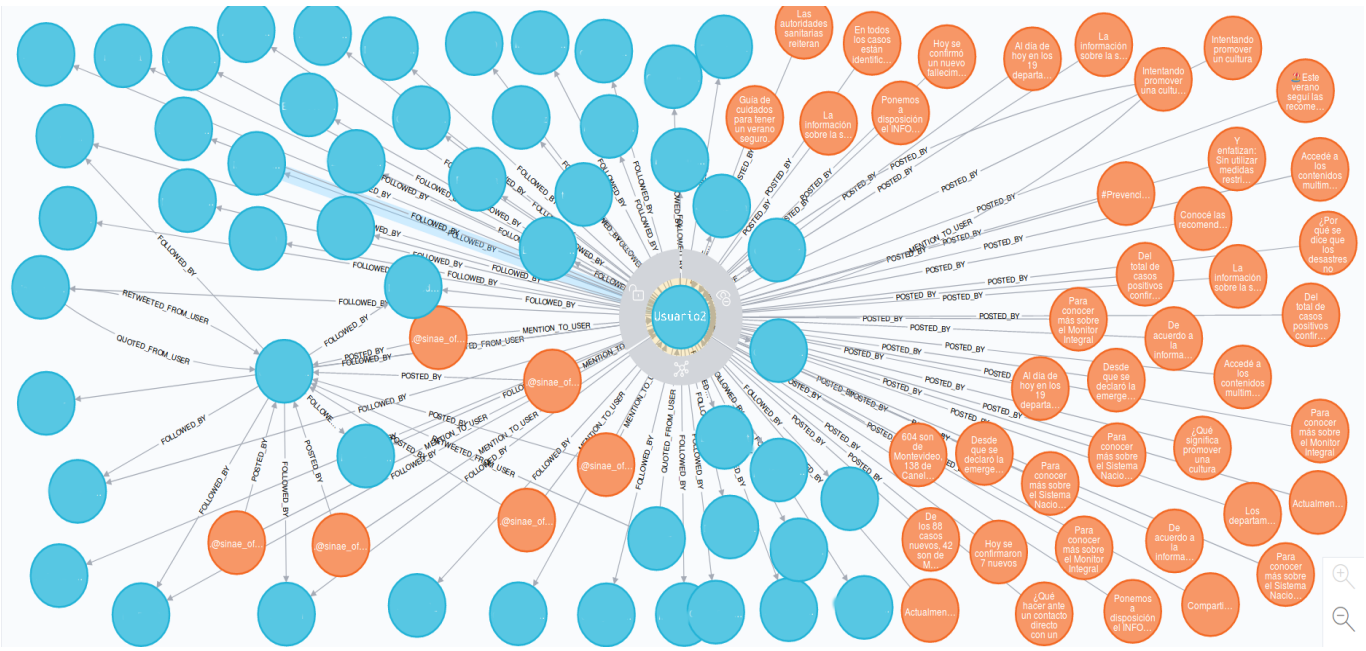

Figura 20. Representación del grafo del usuario usuario2 con sus respectivos seguidores (azul) y tweets (anaranjado).

Tweet utilizado:

- <id>: 108769
- community: 108769
- coordinates: None
- created_at: Fri Nov 13 23:18:05 +0000 2020
- favorite_count: 109
- hashtags: [{'text': 'coronavirus', 'indices': [60, 72]}]
- id: 1327390306405404678
- retweet_count: 67
- text: La información sobre la situación en Uruguay en relación al #coronavirus, están disponible en el visualizador del M… https://t.co/uGTeW5Ljxp
- user_mentions: []

En la Figura 21 se muestra la representación del grafo del tweet:

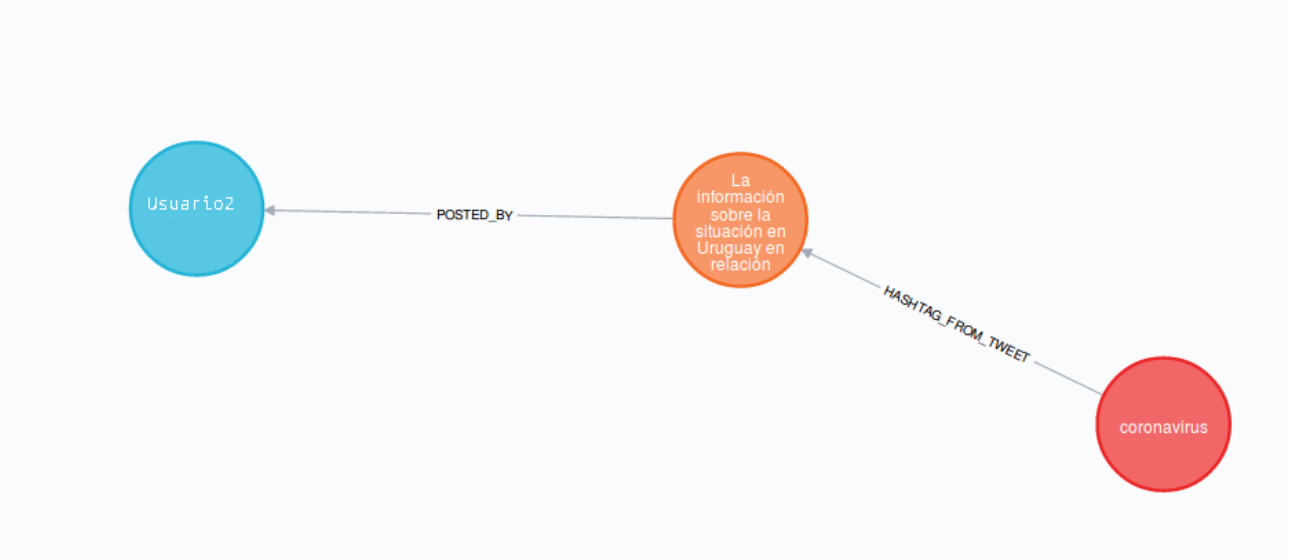

Figura 21. Representación del grafo del tweet de usuario2 con sus respectivas relaciones.

3.3- Usuario3

- <id>:114321
- creation_date: Wed Oct 14 12:12:52 +0000 2020
- description: ¡Hola! Soy un bot, seguime. Tiro un tweet por día con los casos real time del Coronavirus en 🇺🇾.
- favourites_count: 5
- followers_count: 102
- friends_count: 2
- id: 1316351229203812353
- listed_count: 0
- location: Uruguay
- name: usuario3
- real_name: xxx
- statuses_count: 89
- url: https://t.co/5g1d1glSIM

- verified: false
- Cantidad de tweets en base: 22
- Cantidad de Followers en base: 20

A continuación, en la Figura 22, se muestra la representación del grafo del usuario:

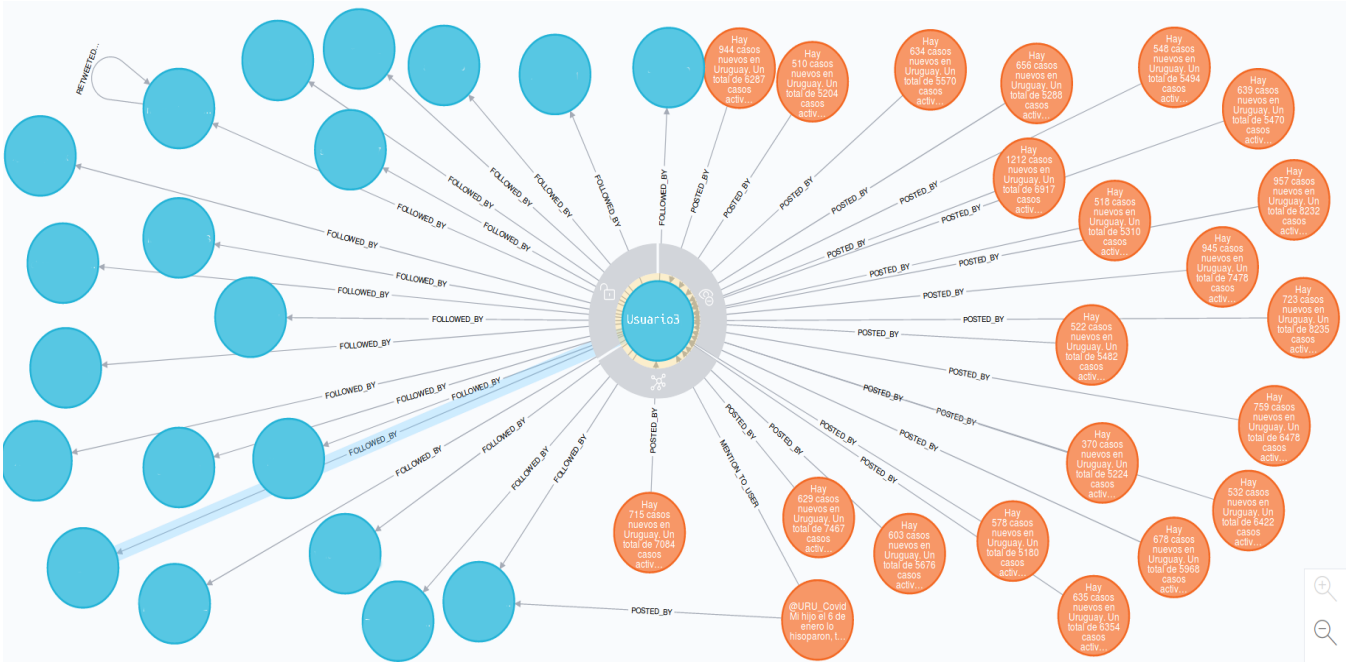

Figura 22. Representación del grafo del usuario usuario3 con sus respectivos seguidores (azul) y tweets (anaranjado).

Tweet utilizado:

- <id>: 114341
- coordinates: None
- created_at: Sun Dec 27 00:30:45 +0000 2020
- favorite_count: 1
- hashtags: []
- id: 1342991271351226374
- retweet_count: 0

- ❏ text: Hay 370 casos nuevos en Uruguay. Un total de 5224 casos activos, 147 personas fallecidas y 10847 personas recuperadas.
- ❏ user_mentions: []

En la Figura 23 se muestra la representación del grafo del tweet:

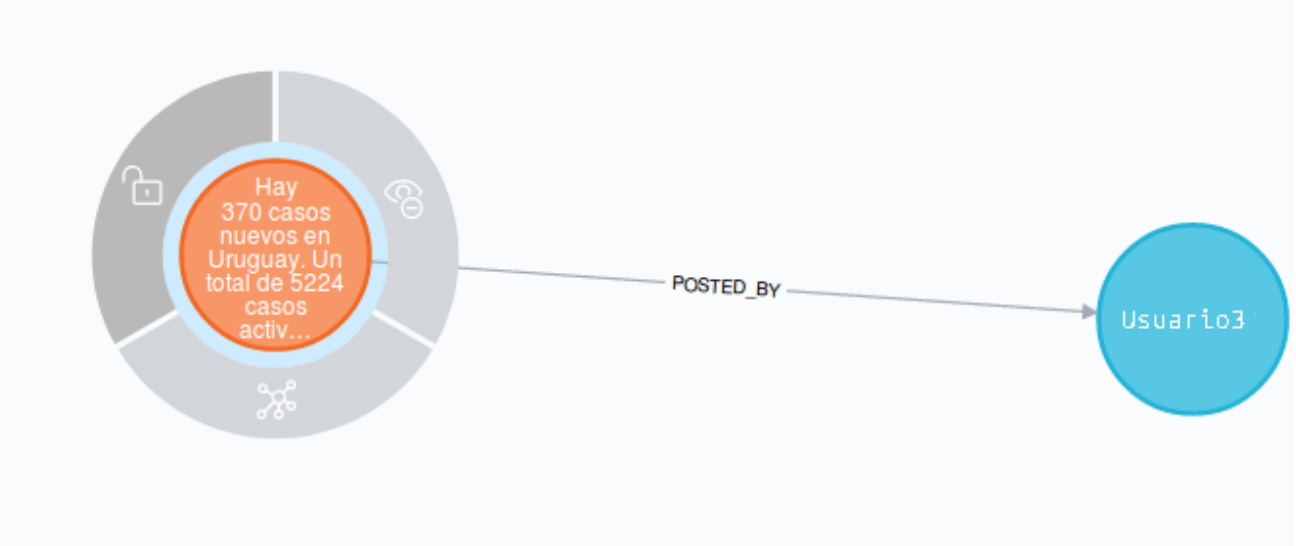

Figura 23. Representación del grafo del tweet de usuario3 con sus respectivas relaciones.

3.4- Usuario4

- ❏ <id>:111174
- ❏ community: 111174
- ❏ creation_date: Sun Nov 15 21:56:16 +0000 2020
- ❏ description:
- ❏ favourites_count: 0
- ❏ followers_count: 0
- ❏ friends_count: 11
- ❏ id: 1328094388040458241
- ❏ listed_count: 0
- ❏ location:

- ❏ name: usuario4
- ❏ real_name: xxx
- ❏ statuses_count: 4
- ❏ verified: false
- ❏ Cantidad de tweets en base: 4
- ❏ Cantidad de Followers en base: 2

A continuación, en la Figura 24, se muestra la representación del grafo del usuario:

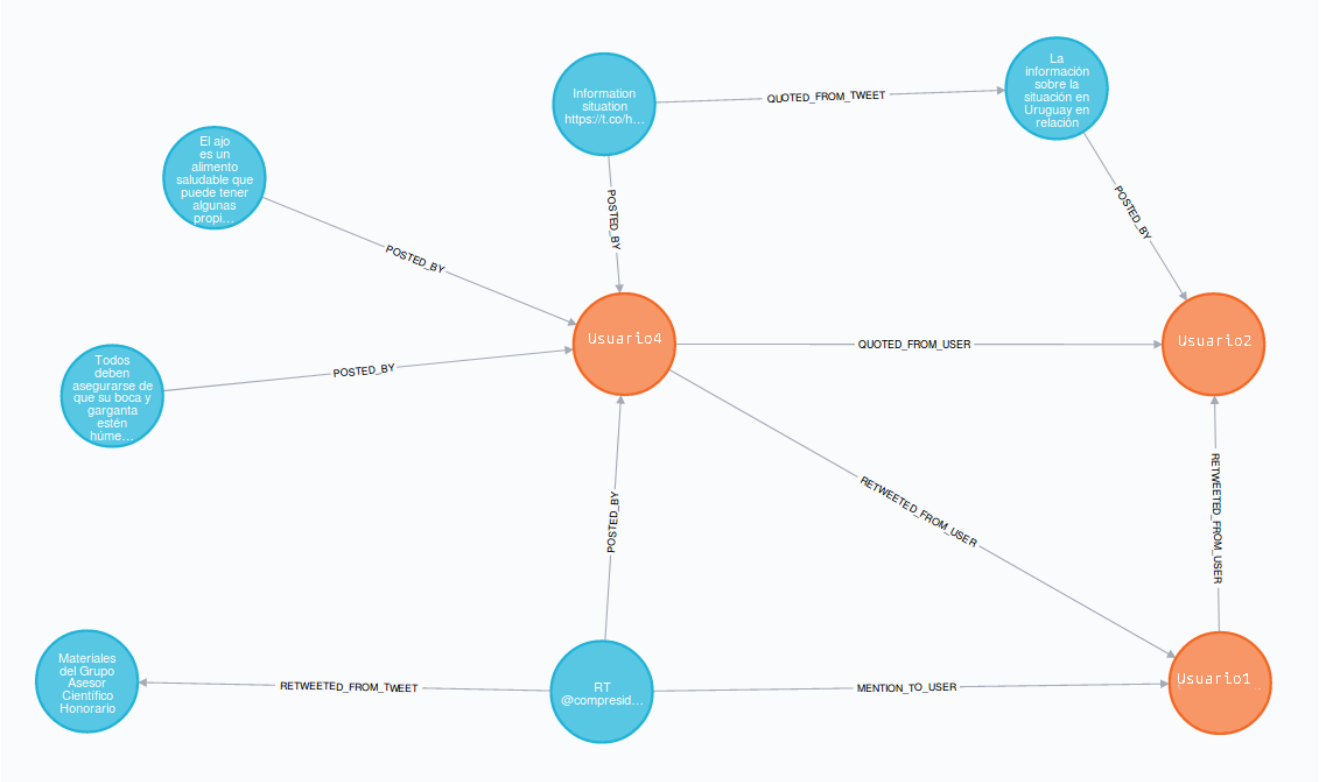

Figura 24. Representación del grafo del usuario usuario4 con sus respectivos seguidores (anaranjado) y tweets (azul).

Tweet utilizado:

- ❏ <id>: 111186
- ❏ community: 111186

- coordinates: None
- created_at: Sun Nov 15 22:20:35 +0000 2020
- favorite_count: 0
- hashtags: []
- id: 1328100611418681347
- retweet_count: 0
- text: El ajo es un alimento saludable que puede tener algunas propiedades antimicrobianas, las cuales ayudan a curar el coronavirus
- user_mentions: []

En la Figura 25 se muestra la representación del grafo del tweet:

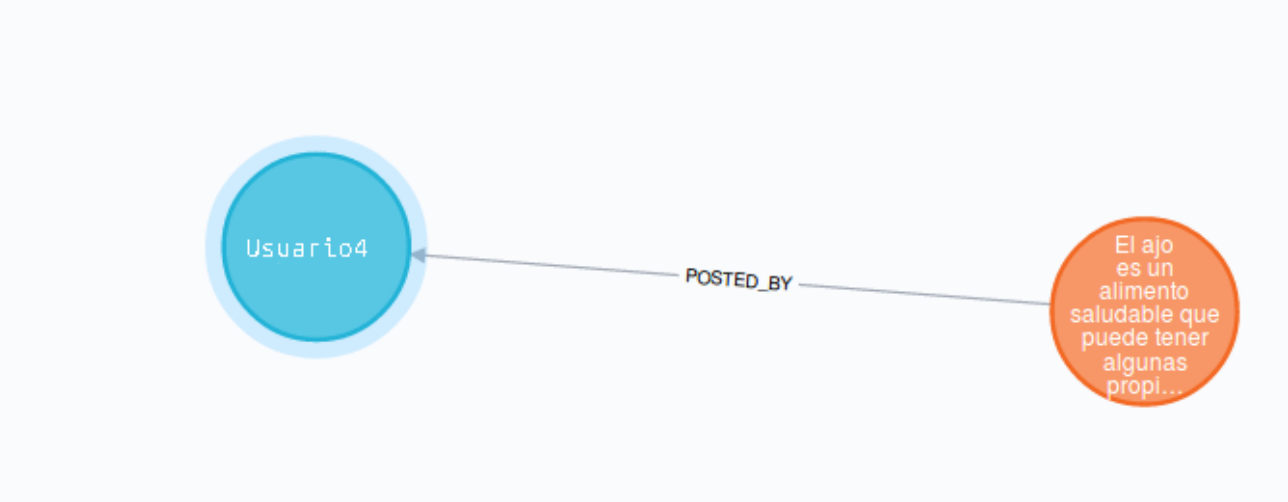

Figura 25. Representación del grafo del tweet de usuario4 con sus respectivas relaciones.

3.5- Usuario5

- <id>:88309
- community: 4440
- creation_date: Sun Jan 24 18:00:15 +0000 2010
- description: Reina

- favourites_count: 31159
- followers_count: 10609
- friends_count: 1788
- id: 108058203
- listed_count: 41
- location:
- name: usuario5
- real_name: xxx
- statuses_count: 80680
- verified: false
- Cantidad de tweets en base: 60
- Cantidad de Followers en base: 39

A continuación, en la Figura 26, se muestra la representación del grafo del usuario:

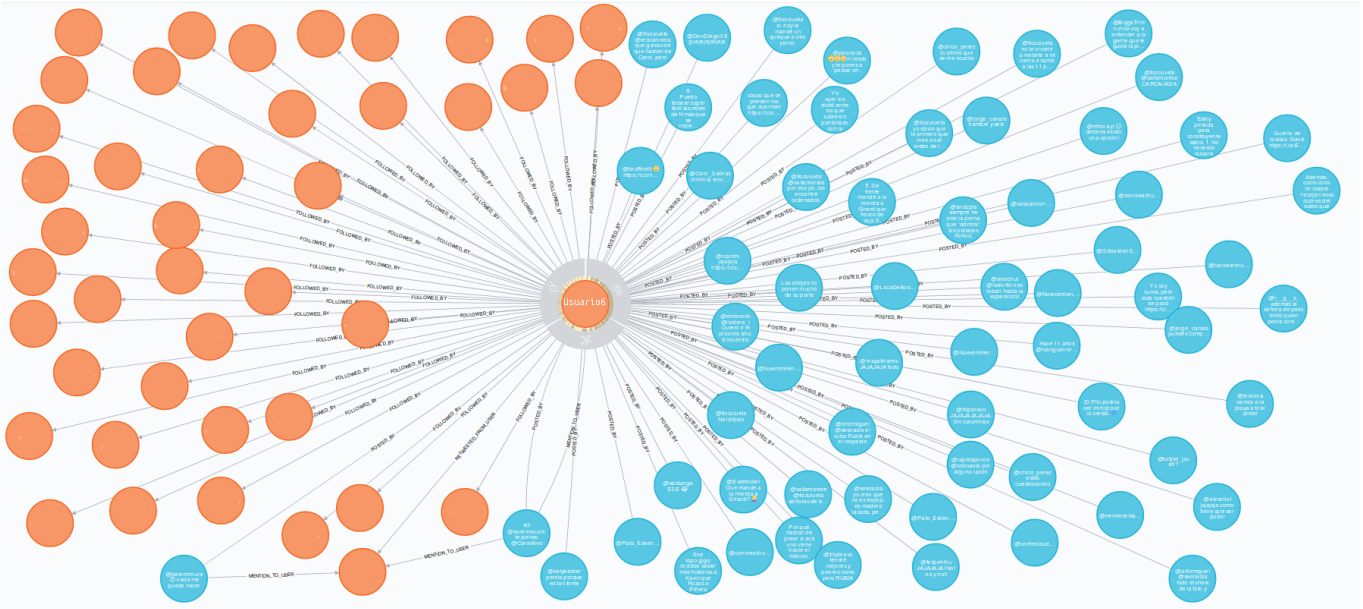

Figura 26. Representación del grafo del usuario usuario5 con sus respectivos seguidores (anaranjado) y tweets (azul).

3.6- Usuario6

- <id>: 108756
- community: 108756
- creation_date: Sat Nov 14 17:47:50 +0000 2020
- description: 💙El tiempo contesta tus preguntas o hace que ya no te importe las respuestas 💛
- favourites_count: 31
- followers_count: 2
- friends_count: 56
- id: 1327669415857582081
- listed_count: 0
- location: República oriental del Uruguay
- name: usuario6
- real_name: xxx
- statuses_count: 8
- verified: false
- Cantidad de tweets en base: 20
- Cantidad de Followers en base: 4

A continuación, en la Figura 27, se muestra la representación del grafo del usuario:

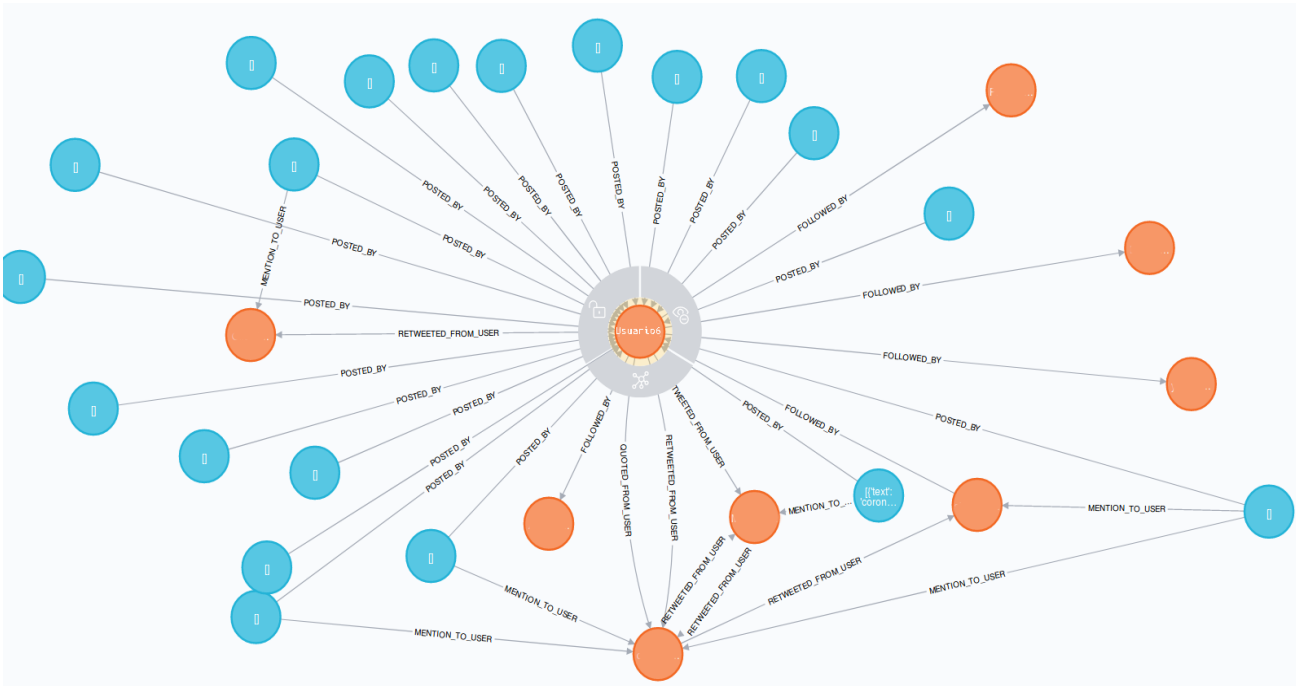

Figura 27. Representación del grafo del usuario usuario6 con sus respectivos seguidores (anaranjado) y tweets (azul).

Tweet utilizado:

- <id>: 113978
- coordinates: None
- created_at: Wed Nov 18 18:43:25 +0000 2020
- favorite_count: 0
- hashtags: []
- id: 1329133123301494784
- retweet_count: 9
- text: RT @####: Con el apoyo de nuestro hidratador oficial @#### , experimentamos un nuevo Test de Sudor. Descubrimos qué necesita c…
- user_mentions: [{'screen_name': ####, 'name': 'PEÑAROL', 'id': 484987132, 'id_str': '484987132', 'indices': [3, 14]}, {'screen_name':

####, 'name': '####', 'id': 491590706, 'id_str': '491590706', 'indices': [59, 71]}]

En la Figura 28 se muestra la representación del grafo del tweet:

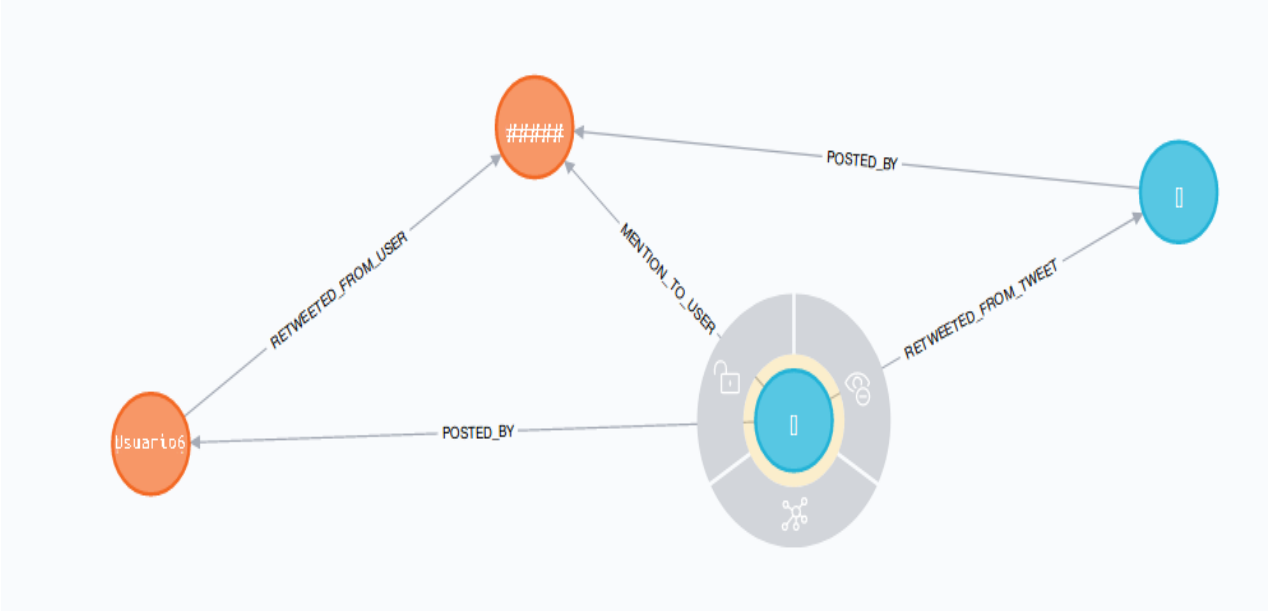

Figura 28. Representación del grafo del tweet de usuario6 con sus respectivas relaciones.

3.7- Usuario7

- <id>: 108845
- community: 113740
- creation_date: Fri Nov 13 15:12:52 +0000 2020
- description: Entre las dificultades se esconde la oportunidad.
- favourites_count: 1
- followers_count: 3
- friends_count: 45
- id: 1327268098345463809
- listed_count: 0
- location:

- name: usuario7
- real_name: xxx
- statuses_count: 0
- verified: false
- Cantidad de tweets en base: 9
- Cantidad de Followers en base: 20

A continuación, en la Figura 29, se muestra la representación del grafo del usuario:

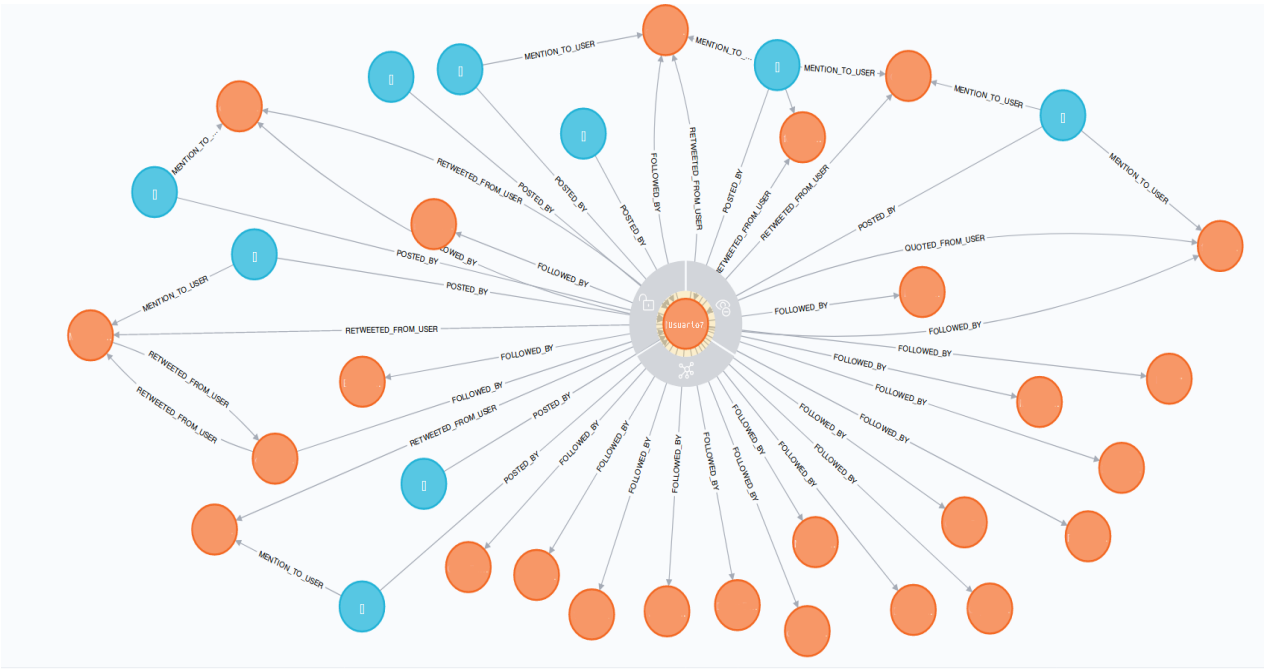

Figura 29. Representación del grafo del usuario usuario7 con sus respectivos seguidores (anaranjado) y tweets (azul).

Tweet utilizado:

- <id>: 113763
- community: 113763
- coordinates: None

- created_at: Sat Jan 02 13:00:35 +0000 2021
- favorite_count: 0
- hashtags: []
- id: 1345354301670100993
- retweet_count: 34
- text: RT @####: La UTE ayudó este año 20!!! No le cortó la luz a la gente...bastó pedir para refinanciar. Le bajó al q riega para q pueda…
- user_mentions: [{'screen_name': '###', 'name': '####', 'id': 1472513011, 'id_str': '1472513011', 'indices': [3, 16]}]

En la Figura 30 se muestra la representación del grafo del tweet:

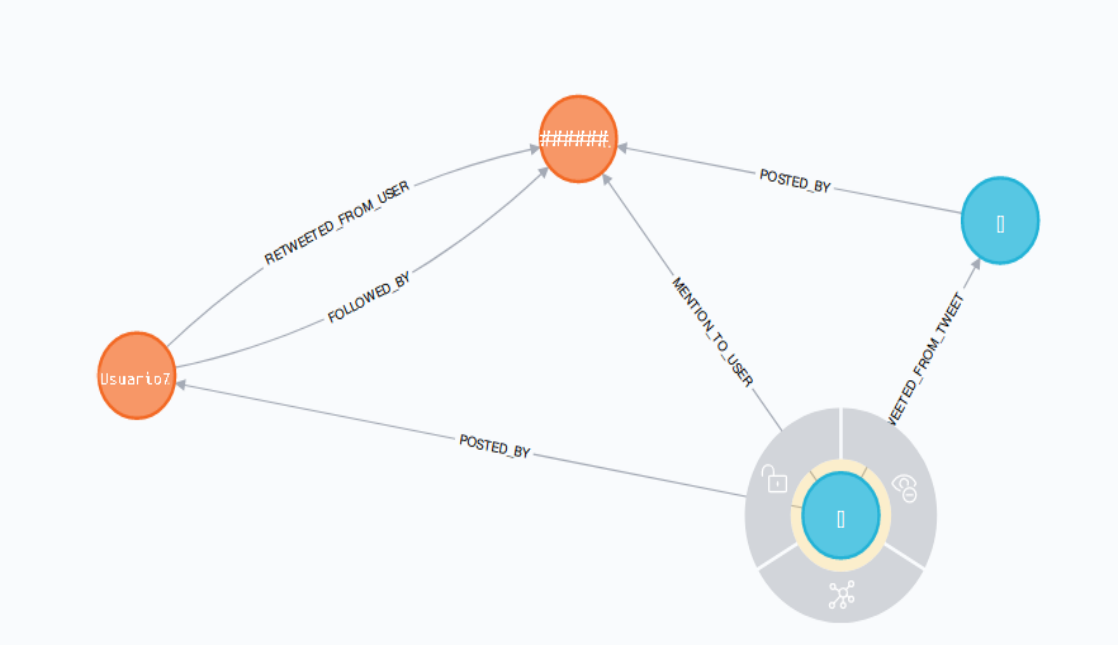

Figura 30. Representación del grafo del tweet de usuario7 con sus respectivas relaciones.

4- Valores de Normalización de las Distintas Métricas

A continuación, se detallan (Tabla 24) los distintos valores obtenidos para la normalización de las métricas teniendo en cuenta los datos de la base de datos utilizada.

Metrics	Variable	Value
followers_quantity	followers_avg_set	9.197,459
tweets_retweets_pub	tweets_retweets_pub_avg_set	50.869,245
favourite	avg_favourite	38.998,076
antiquity	twitter_foundation_date	21/03/2006
verified	-	-
commented	statuses_avg_set	3,697
quote_count	avg_quote_count	-
retweets_count	avg_retweets_count	2.032,109
favourite_count	avg_favourite_count	43.134,000
followers_credibility		8.1
similar_words	-	-
same_location	-	-
reputation_path	-	-

Tabla 24. Valores de normalización para las distintas métricas.

5- Cálculo de Métricas

En esta sección se exponen los cálculos de las diferentes métricas para los usuarios descritos en la sección anterior.

5.1- Followers_quantity

Se refiere a la cantidad de seguidores que tiene un usuario.

Como se refleja en la Tabla 25 para los usuarios usuario1 (3.1), usuario2 (3.2), usuario5 (3.5) la métrica vale 1, dado que estos tres usuarios tienen una cantidad de seguidores mayor al promedio obtenido (9197,459) (Tabla 24).

user_name	followers_quantity	avg: 9197.459684056817
	mongodb	neo4j
usuario1	1,000	1,000
usuario2	1,000	1,000
usuario3	0,011	0,011
usuario4	0,000	0,000
usuario5	1,000	1,000
usuario6	0,000	0,000
usuario7	0,003	0,000

Tabla 25. Valor métrica Followers_Quantity para los distintos usuarios.

A continuación, se muestra gráficamente (Figura 31) los valores obtenidos:

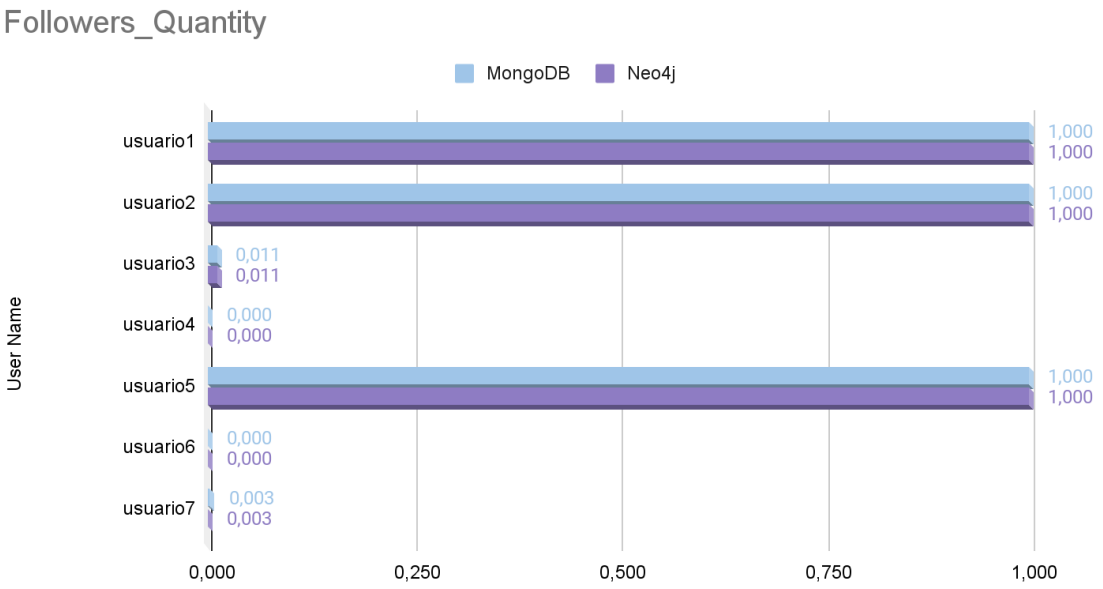

Figura 31. Gráfica métrica Followers_Quantity para los distintos usuarios.

5.2- Tweets_retweets_pub

Se refiere a la cantidad de tweets publicados por el usuario.

En este caso (Tabla 26) el usuario usuario5 (3.5) obtuvo una métrica con valor a 1 dado que el valor statuses_count (80680) supera el valor promedio obtenido (Tabla 24). Como era de esperar los usuarios usuario2 (3.2), usuario3 (3.3) obtuvieron valores bajos dado que son usuarios más que nada informativos, se encargan de publicar y no tanto de interactuar con el resto de los usuarios.

user_name	tweets_retweets_pub	avg: 50869.24450753556
	mongodb	neo4j
usuario1	0,834	0,828
usuario2	0,324	0,294
usuario3	0,000	0,000
usuario4	0,000	0,000

usuario5	1,000	1,000
usuario6	1,000	1,000
usuario7	0,000	0,000

Tabla 26. Valor métrica Tweets_Retweets_Pub para los distintos usuarios.

A continuación, se pueden apreciar gráficamente (Figura 32) los valores obtenidos:

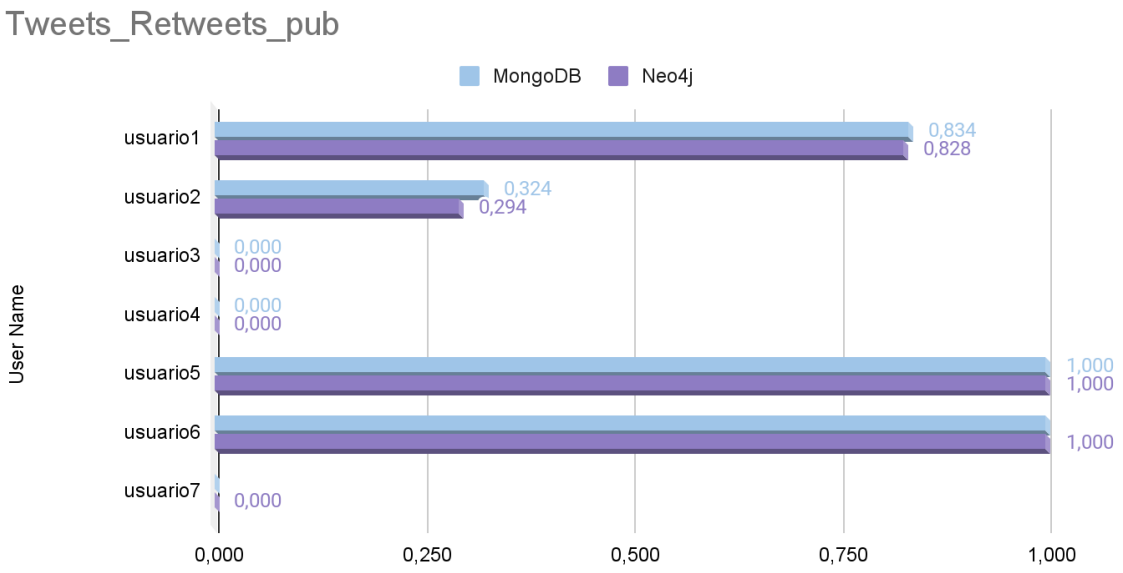

Figura 32. Gráfica métrica Tweets_Retweets_Pub para los distintos usuarios.

5.3- Favourite

Cantidad de me gusta realizados por el usuario.

Como se puede apreciar en la tabla (Tabla 27), el usuario usuario5 (3.5) es el que obtuvo mayor valor para esta métrica. Este dato concuerda con los datos de los diferentes usuarios, el campo *favourite_count* vale 10609, con respecto a usuario1 (3.1) por ejemplo, que vale 5422; lo cual tiene sentido dado que los usuarios usuario1, usuario2 (3.2) y usuario3 (3.3) son sobre todo informativos.

user_name	Favourite	avg: 38998.07595911324
	mongodb	neo4j
usuario1	0,139	0,139
usuario2	0,016	0,016
usuario3	0,000	0,000
usuario4	0,000	0,000
usuario5	0,861	0,799
usuario6	0,010	0,001
usuario7	0,001	0,000

Tabla 27. Valor métrica Favourite para los distintos usuarios.

A continuación, se pueden apreciar gráficamente (Figura 33) los valores obtenidos:

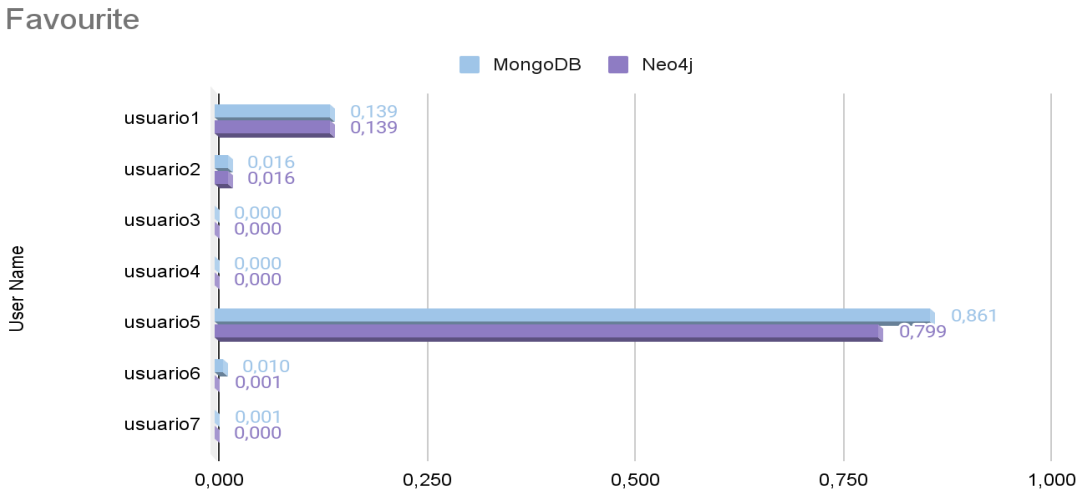

Figura 33. Gráfica métrica Favourite para los distintos usuarios.

5.4- Antiquity

Diferencia entre la fecha de creación del perfil del usuario en Twitter y la fecha actual.

Entre los usuarios utilizados los más antiguos son usuario5 (3.5), creado el 24/01/2010 y usuario1 (3.1), con fecha de creación el 27/09/2011. Según reflejan los datos obtenidos (Tabla 28), las métricas para ambos son las mayores.

user_name	Antiquity	
	mongodb	neo4j
usuario1	0,628	0,628
usuario2	0,562	0,562
usuario3	0,018	0,018
usuario4	0,013	0,013
usuario5	0,741	0,741
usuario6	0,013	0,013
usuario7	0,028	0,000

Tabla 28. Valor métrica Antiquity para los distintos usuarios.

A continuación, se muestra gráficamente (Figura 34) los valores obtenidos:

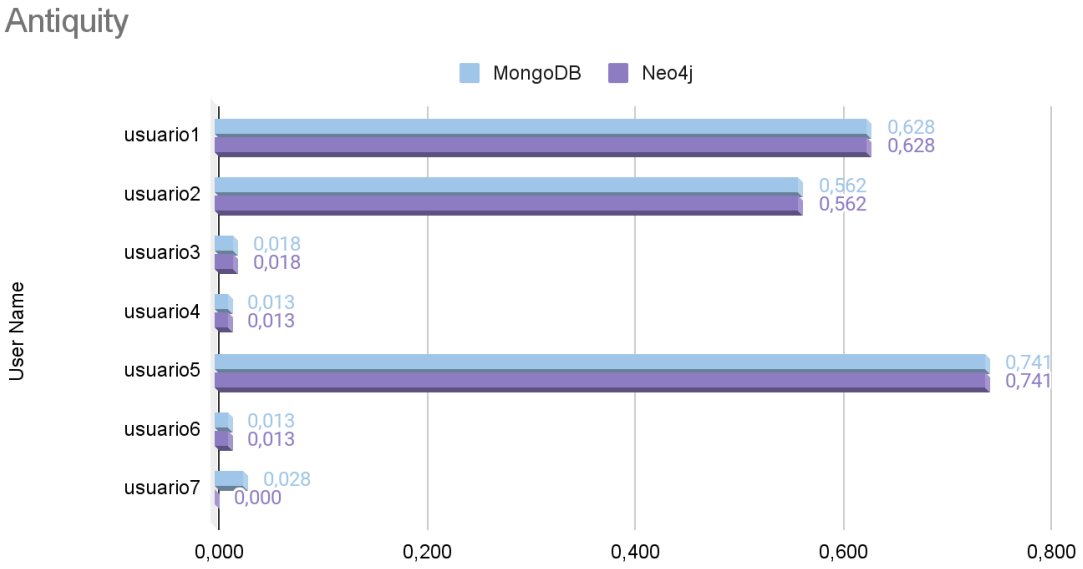

Figura 34. Gráfica métrica Antiquity para los distintos usuarios.

5.5- Verified

Categorización de usuario verificado realizada por Twitter.

Esta métrica es un valor booleano que representa si el usuario está verificado por Twitter. Los datos obtenidos corresponden a lo esperado dado que los únicos usuarios verificados son usuario1 (3.1) y usuario2 (3.2), ambos con valor 1 (Tabla 29).

user_name	Verified	
	mongodb	neo4j
usuario1	1,000	1,000
usuario2	1,000	1,000
usuario3	0,000	0,000
usuario4	0,000	0,000
usuario5	0,000	0,000
usuario6	0,000	0,000

usuario7	0,000	0,000

Tabla 29. Valor métrica Verified para los distintos usuarios.

A continuación, se pueden apreciar gráficamente (Figura 35) los valores obtenidos:

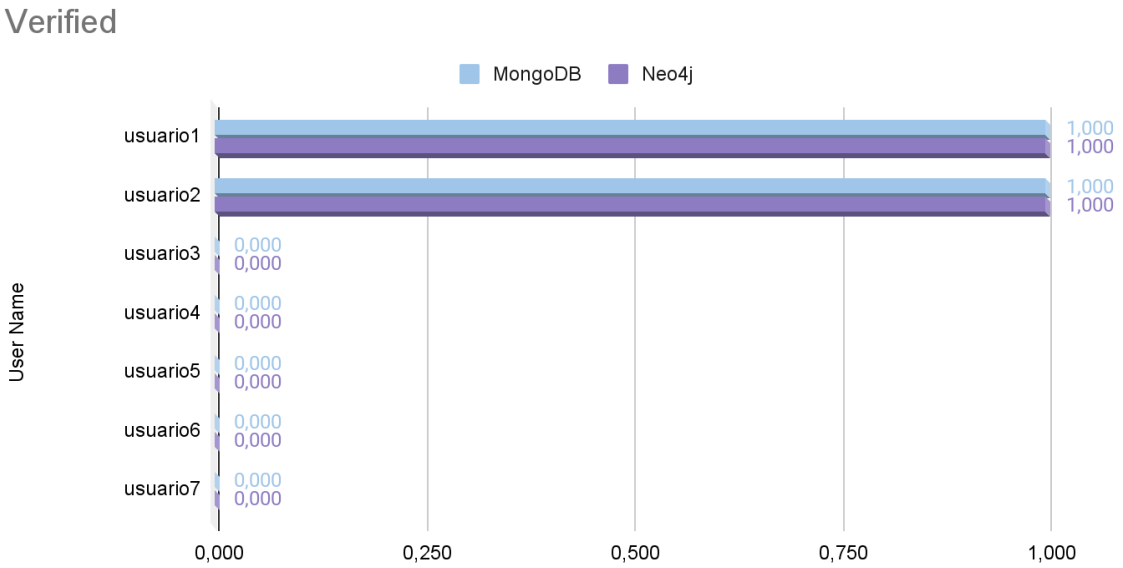

Figura 35. Gráfica métrica Verified para los distintos usuarios.

5.6- Commented

Cantidad de comentarios realizados por el usuario.

Como los usuarios usuario1 (3.1), usuario2 (3.2) y usuario3 (3.3) son usuarios principalmente informativos era de esperar obtener valores bajos en esta métrica; sin embargo, como se puede apreciar en los resultados (Tabla 30) para usuario2 se obtuvo un valor igual a 1.

Al verificar en la base de datos los tweets de usuario2, se obtuvo que de los 43 tweets posteados 24 corresponden a respuestas a otros tweets, mayor que el valor de *statuses_avg_set* (Tabla 24), por lo que es coherente con el resultado obtenido.

Lo mismo sucede con el usuario usuario6 (3.6), cuyo valor fue 1, de los 20 tweets que se tienen en base 15 de ellos son respuestas a otros tweets.

user_name	Commented	avg: 3.697340425531915
	mongodb	neo4j
usuario1	0,000	0,000
usuario2	1,000	1,000
usuario3	0,000	0,000
usuario4	0,000	0,000
usuario5	1,000	1,000
usuario6	1,000	1,000
usuario7	0,000	0,000

Tabla 30. Valor métrica Commented para los distintos usuarios.

A continuación, se muestra gráficamente (Figura 36) los valores obtenidos:

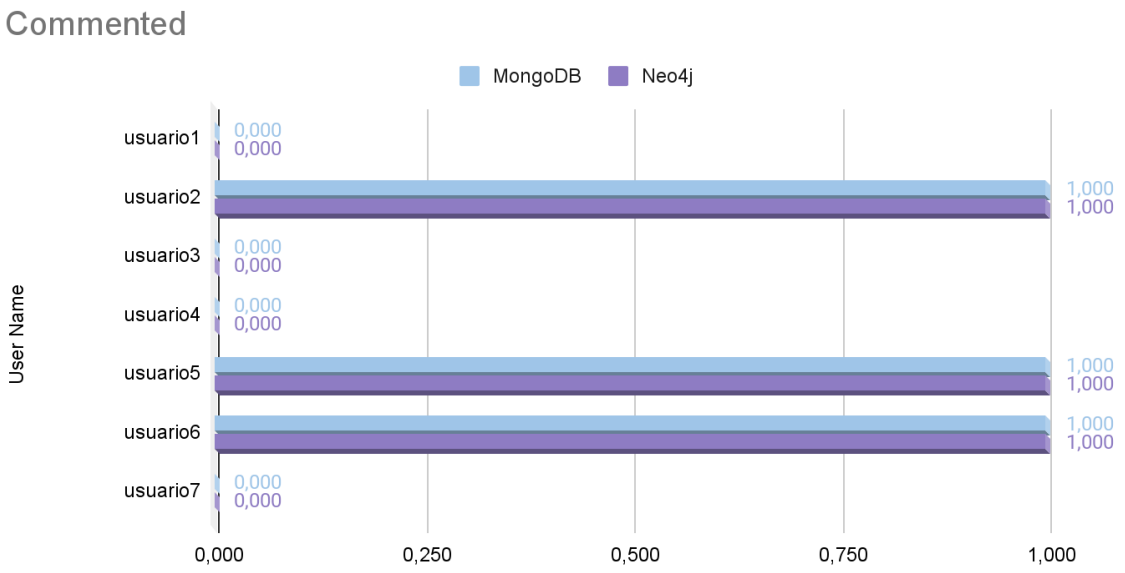

Figura 36. Gráfica métrica Commented para los distintos usuarios.

5.7- Quote_count

Cantidad de citas que tienen todos los tweets de un usuario.

Dado que desde la API de Twitter no se obtuvo ningún valor en el tag *quote_count* para los usuarios, se decidió asignar un peso igual a 0 a esta métrica con el objetivo de que no repercuta de manera negativa en el cálculo de la métrica general. Por tal motivo, como muestra la Tabla 31, el valor de esta métrica es 0 en todos los casos.

user_name	quote_count	
	Mongodb	neo4j
usuario1	0,000	0,000
usuario2	0,000	0,000
usuario3	0,000	0,000
usuario4	0,000	0,000
usuario5	0,000	0,000
usuario6	0,000	0,000
usuario7	0,000	0,000

Tabla 31. Valor métrica Quote_Count para los distintos usuarios.

A continuación, se pueden apreciar gráficamente (Figura 37) los valores obtenidos:

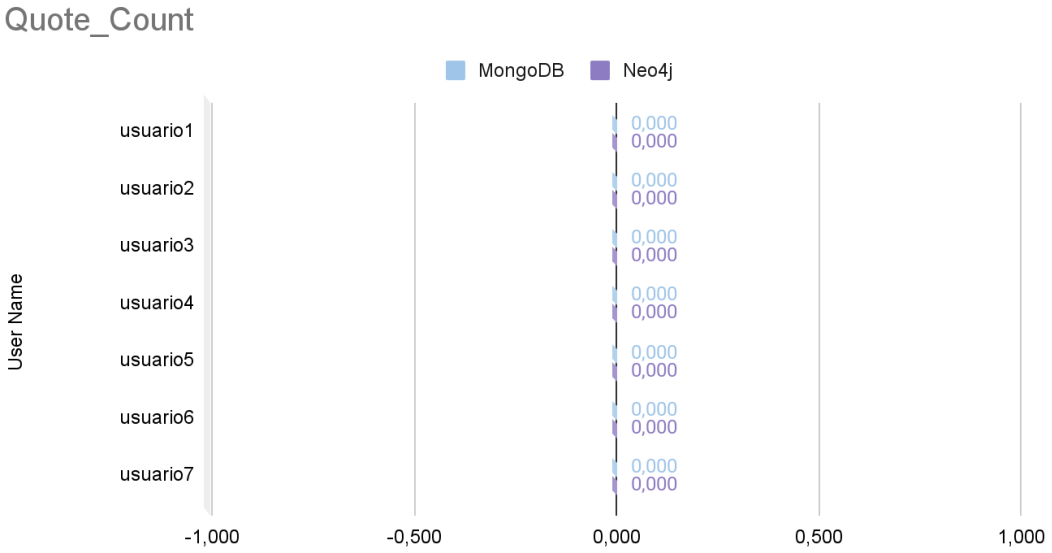

Figura 37. Gráfica métrica Quote_Count para los distintos usuarios.

5.8- Retweets_count

Cantidad de veces que fueron retwitteados los tweets de un usuario.

En este sentido es esperable que tanto los usuarios usuario1 como usuario2 sean los que tengan valores más altos en esta métrica (Tabla 32), dado que al ser cuentas oficiales y estar publicando información relevante a la actualidad con respecto al Covid-19, sus tweets tienen mayor probabilidad de ser compartidos (retwitteados) por sus seguidores.

user_name	retweets_count	avg: 2032.1094384707287
	mongodb	neo4j
usuario1	1,000	1,000
usuario2	0,249	0,233
usuario3	0,000	0,000
usuario4	0,030	0,030
usuario5	0,008	0,008
usuario6	0,017	0,017

usuario7	0,096	0,096

Tabla 32. Valor métrica Retweets_Count para los distintos usuarios.

A continuación, se muestra gráficamente (Figura 38) los valores obtenidos:

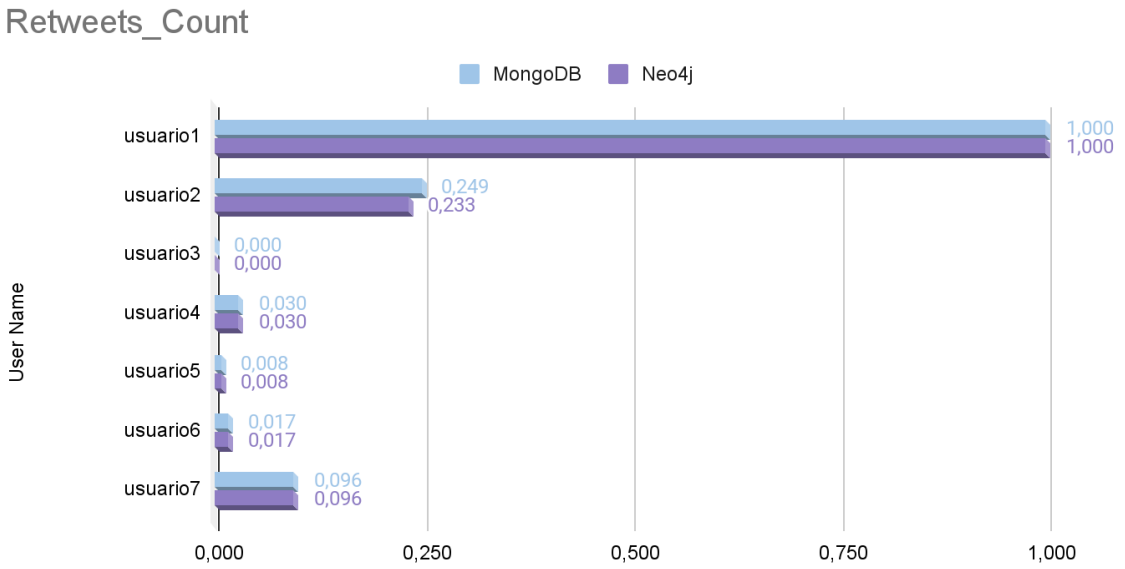

Figura 38. Gráfica métrica Retweets_Count para los distintos usuarios.

5.9- Favourite_count

Cantidad de me gustas que tienen todos los tweets de un usuario.

El valor de la métrica favourite_count muestra la aprobación de los seguidores de un usuario con respecto a los tweets que publica, en este caso nuevamente usuario1 y usuario2 son los que tienen el valor más alto (Tabla 33).

user_name	favourite_count	avg: 43134
	mongodb	neo4j
usuario1	0,105	0,118
usuario2	0,023	0,022
usuario3	0,001	0,001

usuario4	0,000	0,000
usuario5	0,015	0,014
usuario6	0,000	0,000
usuario7	0,000	0,000

Tabla 33. Valor métrica Favourite_Count para los distintos usuarios.

A continuación, se pueden apreciar gráficamente (Figura 39) los valores obtenidos:

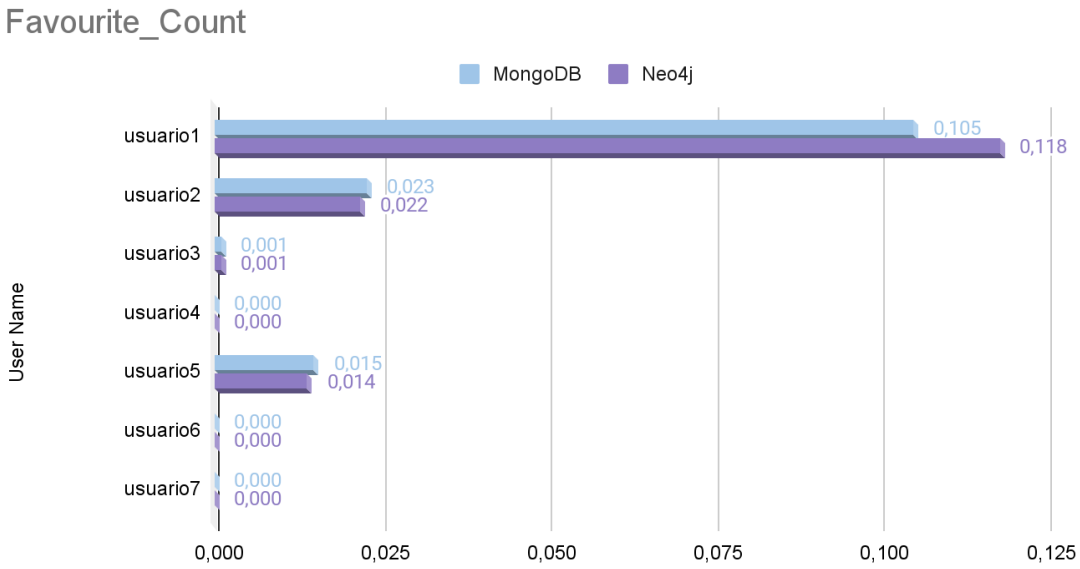

Figura 39. Gráfica métrica Favourite_Count para los distintos usuarios.

5.10- Followers_credibility

Se refiere al promedio de calidad de los seguidores que tiene un usuario.

En este caso la calidad de los seguidores de todos los usuarios es relativamente baja, esto se debe a que no tenemos un dominio que contenga todos los seguidores de un usuario, por tanto, la métrica depende fuertemente de la calidad y cantidad de seguidores que se fueron traídos por la API de Twitter.(Tabla 34).

user_name	followers_credibility	ponderador 8.1
	Mongodb	neo4j
usuario1	0,157	0,152
usuario2	0,201	0,200
usuario3	0,304	0,304
usuario4	0,000	0,000
usuario5	0,175	0,175
usuario6	0,163	0,163
usuario7	0,135	0,135

Tabla 34. Valor métrica Followers_credibility para los distintos usuarios.

A continuación, se pueden apreciar gráficamente (Figura 40) los valores obtenidos:

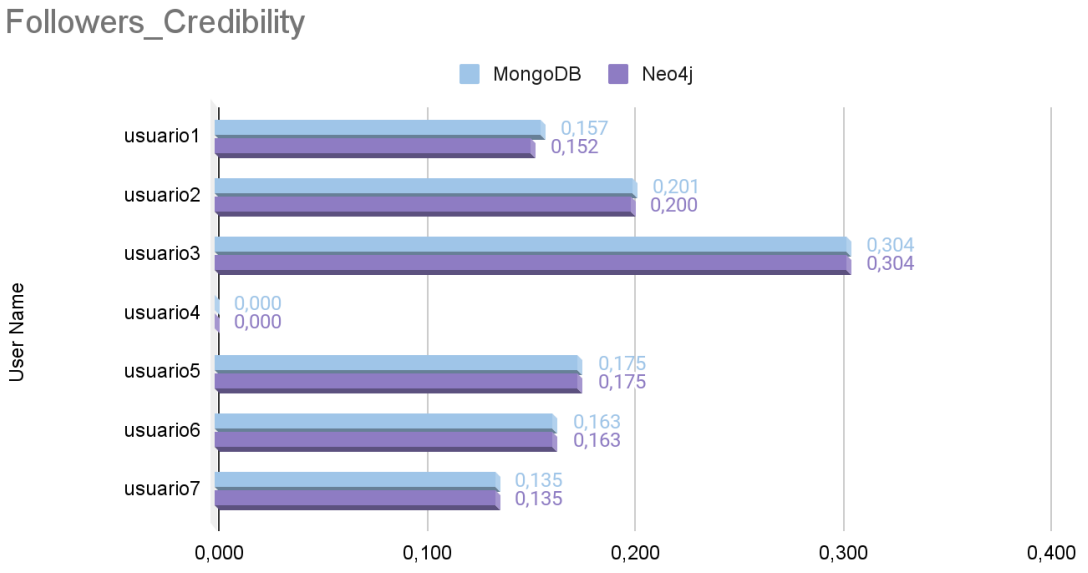

Figura 40. Gráfica métrica Followers_Credibility para los distintos usuarios.

5.11- Similar_words

Cantidad de palabras similares entre los diferentes tweets publicados o retwitteados por el usuario.

Como era de esperar, observando los resultados (Tabla 35), el usuario usuario3 (3.3), que es catalogado como un bot, obtiene un valor en esta métrica igual a 1; esto se debe a que todos sus tweets tienen prácticamente el mismo mensaje: *Hay ... casos nuevos en Uruguay. Un total de ... casos activos, ... personas fallecidas y ... personas recuperadas.*

user_name	similar_words	
	mongodb	neo4j
usuario1	0,693	0,694
usuario2	0,676	0,676
usuario3	1,000	1,000
usuario4	0,488	0,488
usuario5	0,467	0,467
usuario6	0,000	0,000
usuario7	0,000	0,000

Tabla 35. Valor métrica Similar_Words para los distintos usuarios.

A continuación, se pueden apreciar gráficamente (Figura 41) los valores obtenidos:

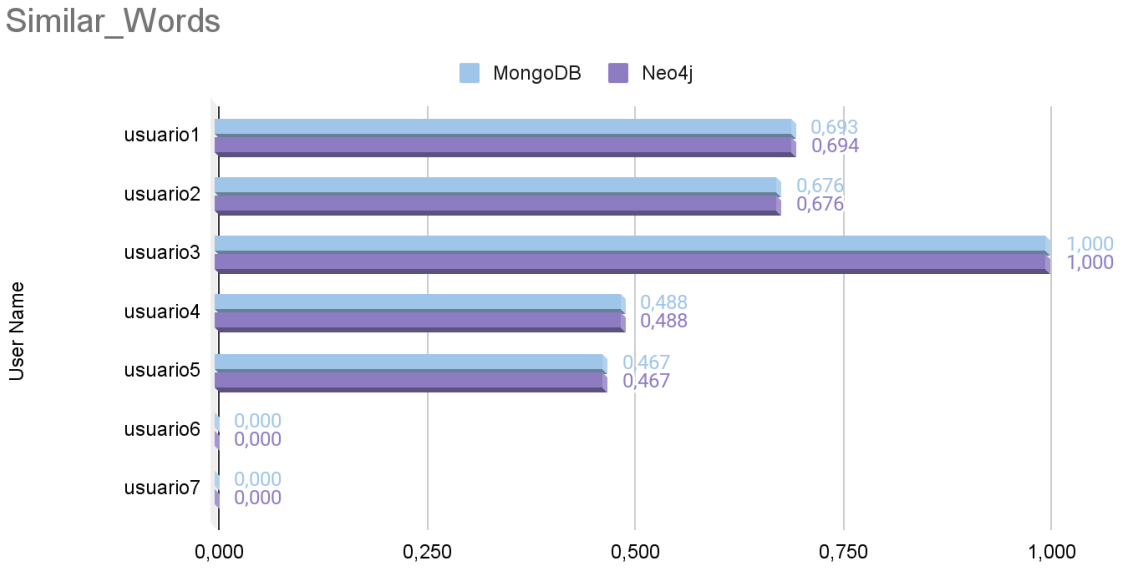

Figura 41. Gráfica métrica Similar_Words para los distintos usuarios.

5.12- Same_location

Coincidencia entre la ubicación del mensaje y la ubicación válida del mismo.

Los tweets utilizados para todos los usuarios (3) no tienen asignado ningún valor en el tag coordinates (este campo es opcional en Twitter), por lo que la métrica en todos los casos dio 0,5 (valor medio definido para que, en caso de no poder calcular la métrica, no pese de una manera positiva ni negativa al total).

Los resultados se pueden ver en la siguiente tabla (Tabla 36):

user_name	same_location	
	mongodb	neo4j
usuario1	0,500	0,500
usuario2	0,500	0,500
usuario3	0,500	0,500
usuario4	0,500	0,500

usuario5	0,500	0,500
usuario6	0,500	0,500
usuario7	0,500	0,500

Tabla 36. Valor métrica Same_location para los distintos usuarios.

A continuación, se muestra gráficamente (Figura 42) los valores obtenidos:

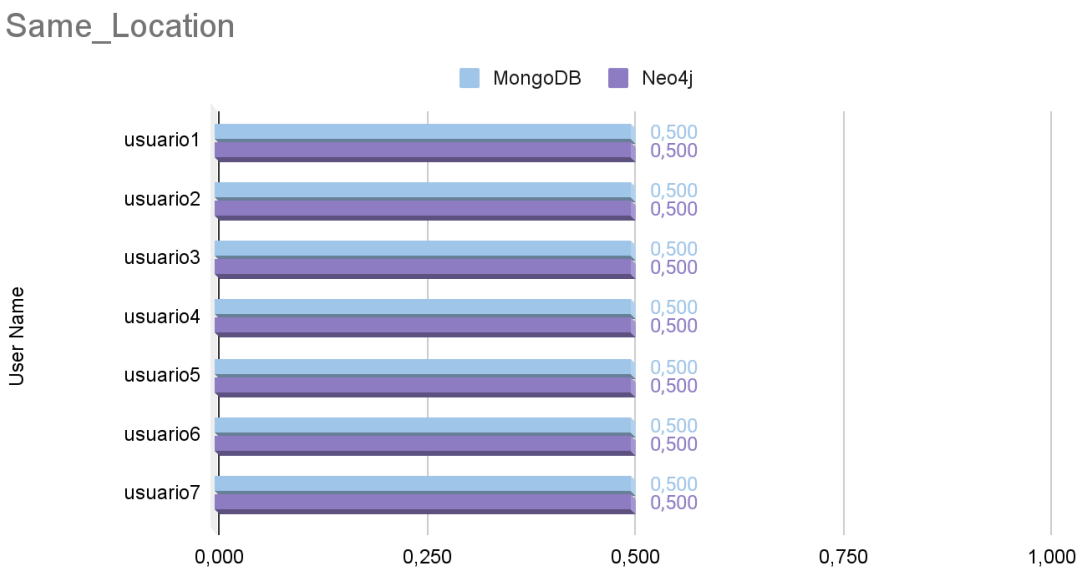

Figura 42. Gráfica métrica Same_Location para los distintos usuarios.

5.13- Reputation_path

El grado en el mensaje no varía desde el origen al destino.

Dado que se utilizó el mismo tweet como origen y destino era de esperar que el valor de esta métrica para todos los usuarios fuera igual a 1 (Tabla 37).

user_name	**reputation_path**	
	mongodb	**neo4j**
usuario1	1,000	1,000
usuario2	1,000	1,000

usuario3	1,000	1,000
usuario4	1,000	1,000
usuario5	1,000	1,000
usuario6	1,000	1,000
usuario7	1,000	1,000

Tabla 37. Valor métrica Reputation_Path para los distintos usuarios.

A continuación, se pueden apreciar gráficamente (Figura 43) los valores obtenidos:

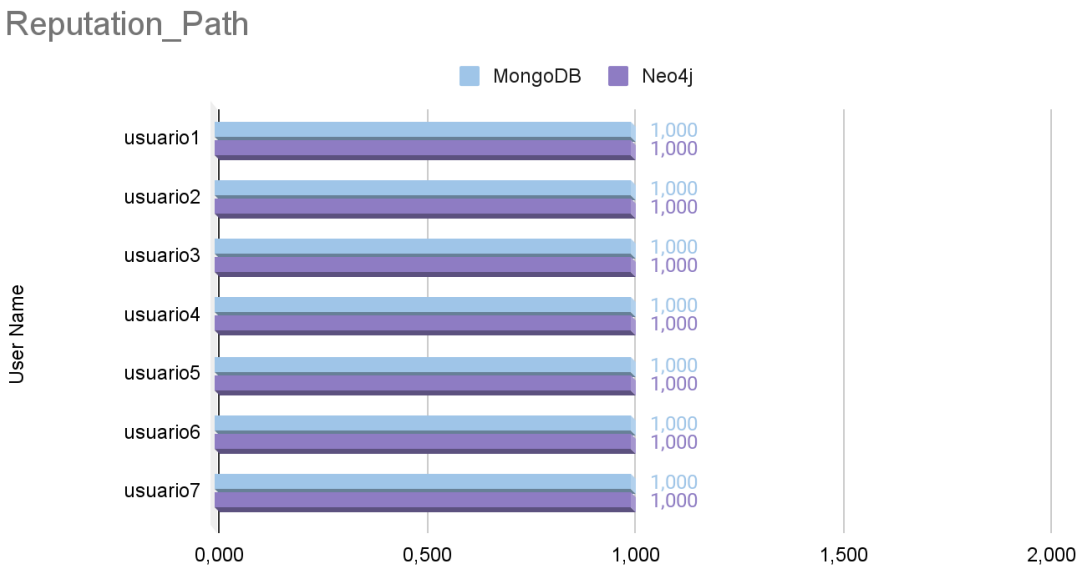

Figura 43. Gráfica métrica Reputation_Path para los distintos usuarios.

5.14- General_credibility

Se realiza un promedio de los resultados obtenidos en cada métrica, corridos en la base seleccionada y con los respectivos ponderadores seteados (Tabla 22).

Dentro de los cinco usuarios evaluados y teniendo en cuenta el parámetro de credibilidad *credibility_range*, cuyo valor fue definido a 0,5, se obtuvo como resultado que los usuarios usuario1 (3.1) y usuario2 (3.2) y usuario5 (3.5)

son más tendientes a ser creíbles que el resto, rondando el 0,5 con respecto al 0,1 en los demás casos.

En la Tabla 39 se muestran los valores de la métrica *general_credibility* obtenidos para cada usuario.

user_name	general_credibility
usuario1	0,567
usuario2	0,524
usuario3	0,090
usuario4	0,104
usuario5	0,545
usuario6	0,341
usuario7	0,151

Tabla 39. Valor métrica General_Credibility para los distintos usuarios.

A continuación, se pueden apreciar gráficamente (Figura 44) los valores obtenidos:

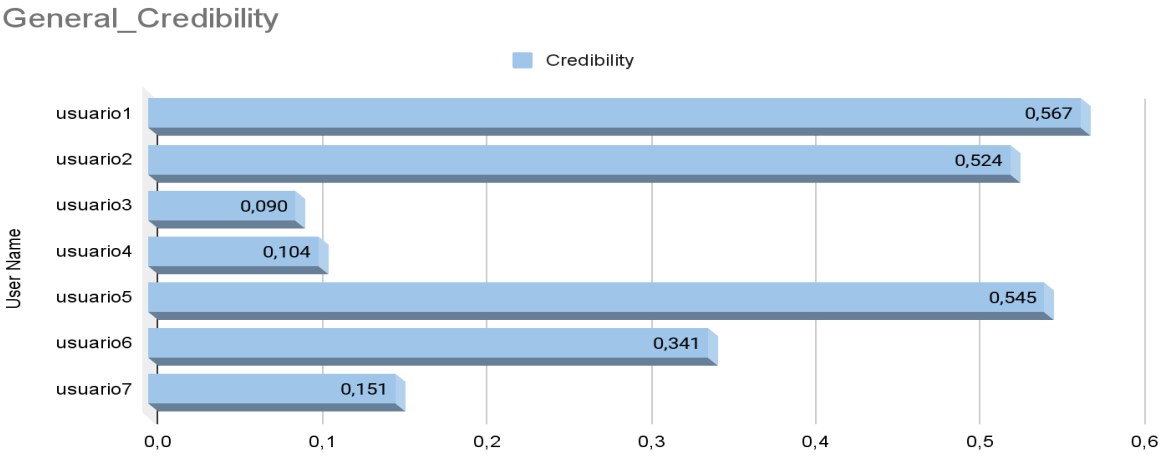

Figura 44. Gráfica métrica General_Credibility para los distintos usuarios.

6- Análisis de Resultados

De los resultados obtenidos podemos ver que en la métrica general los usuarios **usuario1**, **usuario2** obtienen un resultado mayor a 0.5 sin embargo son usuarios oficiales los dos y su valor ronda entre 0.53 y 0.56; tendríamos a pensar que debería rondar más cerca del 1 que de la mitad para estos usuarios.
Si analizamos los resultados de cada métrica por separado vemos que para *followers_quantity, tweets_retweets_pub, verified, reputation_path* obtenemos el valor máximo y tiene buenos ponderadores. En la métrica *similar_words* obtenemos un valor entre 0.6 y 0.7, lo cual representa la realidad ya que estos usuarios en esta época twittean muchas cosas relacionadas con el covid, y por ende podemos considerarlos bastante monotemáticos. La métrica *antiquity* nos revela que los usuarios no son ni muy antiguos, ni muy recientes obteniendo valores entre 0.55 y 0.65 para estos y la métrica *same_location* nos devuelve 0.5 para los dos casos, lo cual indica que no hay forma de comparar las locaciones, por tanto, no da un valor ni positivo, ni negativo.

Las métricas *favourite_count, followers_credibility, favourite* nos dan valores entre 0 y 0.2; si analizamos en detalle cada una podemos ver que favourite representa la cantidad de me gustas que realiza el usuario, estos usuarios son usuarios oficiales, que se dedican a informar cosas sobre todo y no tanto a interactuar, por otro lado *favourite_count* y *followers_credibility* los cuales representan la cantidad de me gustas a los tweet del usuario y la calidad de los seguidores de los mismos, dependen fuertemente de los tweet y seguidores que hay en el dominio, para estas métricas consideramos que si tuviéramos un dominio más cercano a la realidad tendríamos valores en la mitad superior y no cercanos a 0.

Las métricas *commented, retweets_count* representan la cantidad de comentarios hechos por el usuario y las veces que se retuitean los tweets de estos. En el caso de *commented* las métricas nos dicen que **usuario2** realiza varios comentarios y **usuario1** no muchos. Por otro lado, los dos deberían de tener un valor alto en *retweets_count*, en este caso vemos que mientras **usuario1** tiene un 1 en esta métrica **usuario2** tiene un valor muy bajo 0,249,

creemos que esto se debe al dominio de tweets, ya que este valor también depende de los tweets seleccionados.

Para los casos de **usuario3** y **usuario4** los cuales son un bot y un usuario falso respectivamente la métrica general logra identificar la no validez de estos usuarios y nos devuelve los valores 0,157 y 0,073. Si analizamos métrica por métrica todas dan valores dentro de lo esperado para ese tipo de usuario, ya que todas arrojan valores muy bajos a excepción de *reputation_path*, que verifica el origen y destino del tweet y por ende no está mal que devuelva ese valor, y *similar_words* que devuelve 1 y 0.48 respectivamente; lo cual está bien ya que esta métrica es negativa y lo que intenta verificar es si un usuario es monotemático.

usuario5 es un usuario normal con una actividad estándar en Twitter, si analizamos las métricas una por una vemos que las mismas logran determinar que **usuario5** es un usuario real, ya que las métricas asociadas al usuario dan en su conjunto valores esperados.

Sin embargo, **usuario6**, **usuario7** son usuarios reales, pero la métrica general da 0,341 y 0,151 respectivamente. Si nos fijamos a detalle, vemos que son usuarios que no tienen mucha actividad en Twitter *favourite_count*, retweets_count y *favourite* dan valores muy bajos. Por otro lado, tampoco tiene mucha antigüedad por lo cual la métrica de antigüedad da baja y la cantidad de seguidores del dominio es poca, lo cual hace que *followers_quantity* sea baja también.

Nuestras métricas apuntan mucho a determinar si un usuario es real o falso, y no tanto a determinar si el mensaje es real o falso, con un usuario real y un mensaje falso, no logra determinar muy bien la diferencia, creemos que la implementación de otras métricas como la experiencia del usuario en el tema a tratar o la oportunidad del mensaje podrían aportar en la solución de este problema. Faltan tal vez algunas métricas más efectivas asociadas al clip, tal vez agregando más técnicas de PLN se podría agregar y mejorar las métricas asociadas al clip. Por otro lado, hay varias métricas que dependen mucho del dominio en el que estemos trabajando. Las asociadas a la cantidad de comentarios, citas y retweets, dependen mucho de los tweets traídos y la información que hay en el dominio.

Capítulo 7 Conclusiones

En esta sección se evalúan los resultados obtenidos en la realización del proyecto, los problemas encontrados y la solución dada. Se comenta el objetivo inicial y la solución a la cual finalmente se arribó. Se especifica el trabajo a futuro que se considera puede realizarse y se da una conclusión final del proyecto.

1- Resultados Obtenidos

Se planteó realizar un estado del arte vinculado a la dimensión de calidad *credibilidad* aplicado a datos provenientes de plataformas de social media. Así como definir un conjunto de factores, métricas y dimensiones basados en la dimensión de credibilidad y desarrollar un prototipo de una herramienta para la evaluación de las dimensiones de calidad propuestas.

En base a estos objetivos se realizó un relevamiento bibliográfico científico y entrevistas a usuarios expertos, para definir las dimensiones y métricas trabajadas en el proyecto; permitiendo de esta manera tener una amplitud de visiones consideradas para la definición de una propuesta basada en dimensiones de calidad de datos con sus respectivos factores y métricas.

Se decidió evidenciar los resultados obtenidos llevándolo a la práctica por medio de un prototipo web basándose en datos de la red social Twitter, implementando las métricas viables y comparando resultados esperados con los recabados. Para acotar el alcance se optó por utilizar tweets geolocalizados próximos al territorio Uruguayo y que estuviesen relacionados con temas relativos al Covid 19.

Los resultados obtenidos en la prueba de concepto muestran que las métricas implementadas logran identificar usuarios reales y falsos. Sin embargo, con un usuario real y un mensaje falso, no logra determinar siempre la diferencia. Se cree que la implementación de otras métricas, como ser la experiencia del usuario en el tema a tratar o la oportunidad del mensaje podrían aportar en la solución de este problema. Desarrollar algunas otras métricas asociadas al clip, utilizando más técnicas de PLN (área en la cual no se hizo foco en este proyecto) podrían llegar a contribuir en la mejora de resultados. Por otro lado

también, hay varias métricas que dependen mucho del dominio, es probable que si se contara con un dominio que se aproxime más a la realidad se hubieran reportado mejores resultados.

En general los resultados obtenidos permiten afirmar que el enfoque aplicado es un enfoque válido, que posibilita representar las características más importantes a tener en cuenta para determinar la credibilidad de una publicación proveniente de redes sociales.

Se cree que el proyecto logró cumplir con todos los objetivos definidos, logrando realizar un relevamiento amplio y desde diferentes perspectivas del estado del arte vinculado a la dimensión de calidad *credibilidad* aplicado a datos provenientes de plataformas de social media. Las entrevistas a expertos fueron un gran aporte a la hora de obtener todas las visiones sobre el tema. En base a este relevamiento se definieron dimensiones, factores y métricas a las cuales luego se las validó mediante la creación del prototipo.

2- Problemas Enfrentados

Una de las mayores problemáticas enfrentadas fue la obtención de datos, la solución se basa fuertemente en el dominio de datos con el cual se trabaja, de este depende la validez del cálculo de las métricas. No es posible contar con una base de datos cercana a la realidad ya que Twitter limita la cantidad de datos que se pueden obtener a partir de la API de uso libre. Para intentar dar solución a este problema y minimizar el error, fueron recabados todos los datos posibles. A su vez se agregaron funcionalidades para traer información de usuarios específicos y sus seguidores, con la intención de generar datos sobre los usuarios o tweets de interés. También casi todas las normalizaciones realizadas dependen del dominio con el que se está trabajando y no de la realidad.

La métrica *quote_count*, la cual refiere a la cantidad de citas de los tweets de un usuario, siempre obtiene un valor igual a 0 dado que el tag *quote_count* del json del tweet retornado por la API (en su versión libre) en todos los casos es nulo. Por ende, si bien dejamos su implementación, le asociamos un peso 0, para no tenerla en cuenta, ya que su valor siempre es 0 por falta de información.

3- Limitaciones

Si bien se intentó obtener un dominio cercano a la realidad del usuario a estudiar, la API de Twitter limita mucho la cantidad y calidad de los tweets traídos, los tweet y seguidores son aleatorios y limitados; se cree que algunos resultados no esperados para ciertas métricas dependen fuertemente de esto.

Las métricas de la dimensión *Expertise*, si bien fueron definidas, no fueron implementadas. Dado que se decidió basarse en datos obtenidos a partir de la red Twitter y ésta no cuenta con demasiada información sobre nivel académico o profesión de un usuario; y es un problema aparte automatizar la integración con otras redes que proporcionen estos datos; se resolvió no implementar los métodos para estas métricas.

La métrica *excess_activity* requiere que se guarde el historial de actividad de todos los usuarios, es decir se deberían tener los valores de actividad de usuario (me gustas, comentarios, seguidores, etc) en un tiempo inicial y un tiempo final para así poder medir la actividad del usuario. La solución propuesta no maneja esto ya que la forma que obtiene los datos no permite guardar un historial de actividad del usuario.

La implementación de la métrica *timeliness*, requería procesamiento del lenguaje (PLN) para determinar si el contenido de un mensaje era actual u oportuno, se dejó esta métrica fuera del alcance del proyecto, ya que se entendió que el objetivo de este no está entrar en profundidad en esta área.

4- Trabajo a Futuro

Se cree que uno de los principales trabajos a futuro consiste en generar una base de datos más cercana a la realidad, que contenga todos los datos de Twitter, utilizando la API paga y probar las métricas con la misma.

La implementación de todas las métricas asociadas a la dimensión Expertise, sería un aporte muy significativo en la construcción de la credibilidad general. En este proyecto se deja la implementación de las mismas fuera del alcance por no trabajar con una base de datos que proporciona la información pertinente. Se deja, por tanto, como trabajo a futuro el relacionar Twitter con alguna red social o fuente externas como LinkedIn[25], ResearchGate[51] o alguna otra que contenga información académica de los usuarios. En este

proyecto se llega a describir las métricas y su implementación de manera superficial, suponiendo que se cuenta con la información necesaria proporcionada por alguna fuente de datos.

Para implementar la métrica *excess_activity* se debe mantener un historial de la actividad de cada usuario. Actualmente cuando de manera no intencional (consumiendo la API por streaming) o intencional (obteniendo información de algún usuario en particular) se vuelven a traer los datos de actividad de un usuario, los viejos se sobrescriben y no se mantiene un historial de la actividad del usuario. Queda como trabajo a futuro realizar esto último para implementar esta métrica. Una opción para mantener el historial es recorrer, cada cierto tiempo, uno a uno todos los usuarios de la base, ir a buscar información a Twitter de los mismos y guardar cada recorrida con su respectiva información.

Referencias Bibliográficas

[1] C. Batini y M. Scannapieco, *Data and Information Quality*. Cham: Springer International Publishing, 2016. doi: 10.1007/978-3-319-24106-7.

[2] E. Dumbill, «What is big data? - O'Reilly Radar». http://radar.oreilly.com/2012/01/what-is-big-data.html (accedido ago. 11, 2021).

[3] «Twitter is sweeping out fake accounts like never before, putting user growth at risk», *Washington Post*. https://www.washingtonpost.com/technology/2018/07/06/twitter-is-sweeping-out-fake-accounts-like-never-before-putting-user-growth-risk/ (accedido mar. 26, 2019).

[4] L. J. Aguilar, *Big Data, Análisis de grandes volúmenes de datos en organizaciones*. Alfaomega Grupo Editor, 2016.

[5] «We Are Social - Digital Report 2018». https://digitalreport.wearesocial.com/ (accedido ene. 07, 2019).

[6] «Crowdfire: The only social media manager you'll ever need», *Crowdfire: The only social media manager you'll ever need*. https://www.crowdfireapp.com/ (accedido ene. 07, 2019).

[7] «Social Media Marketing & Management Software | Spredfast». https://www.spredfast.com/#geyQmOJXG1IMg7qw.97 (accedido ene. 07, 2019).

[8] «Twitter Developer Platform». https://developer.twitter.com/content/developer-twitter/en.html (accedido ene. 07, 2019).

[9] «Información sobre las API de Twitter». https://help.twitter.com/es/rules-and-policies/twitter-api (accedido sep. 16, 2018).

[10] J. Gomide *et al.*, «Dengue surveillance based on a computational model of spatio-temporal locality of Twitter∗», p. 8.

[11] T. Sakaki, F. Toriumi, y. Matsuo, «Tweet trend analysis in an emergency situation», p. 8.

[12] T. Sakaki, M. Okazaki, y. Matsuo, «Earthquake Shakes Twitter Users: Real-time Event Detection by Social Sensors», p. 10.

[13] A. Tumasjan, T. O. Sprenger, P. G. Sandner, y I. M. Welpe, «Predicting

Elections with Twitter: What 140 Characters Reveal about Political Sentiment», p. 8.

[14] A. Sarker *et al.*, «Utilizing social media data for pharmacovigilance: A review», *J. Biomed. Inform.*, vol. 54, pp. 202-212, abr. 2015, doi: 10.1016/j.jbi.2015.02.004.

[15] N. Prat y S. Madnick, «Evaluating and Aggregating Data Believability across Quality Sub-Dimensions and Data Lineage», Social Science Research Network, Rochester, NY, SSRN Scholarly Paper ID 1075722, dic. 2007. Accedido: ago. 18, 2018. [En línea]. Disponible en: https://papers.ssrn.com/abstract=1075722

[16] R. Y. Wang y D. M. Strong, «Beyond Accuracy: What Data Quality Means to Data Consumers», *J. Manag. Inf. Syst.*, vol. 12, n.º 4, pp. 5-33, 1996.

[17] B. J. Fogg y H. Tseng, «The elements of computer credibility», en *Proceedings of the SIGCHI conference on Human factors in computing systems the CHI is the limit - CHI '99*, Pittsburgh, Pennsylvania, United States, 1999, pp. 80-87. doi: 10.1145/302979.303001.

[18] Y. L. Simmhan, B. Plale, y D. Gannon, «A survey of data provenance in e-science», *ACM SIGMOD Rec.*, vol. 34, n.º 3, pp. 31-36, sep. 2005, doi: 10.1145/1084805.1084812.

[19] Y. Lee, L. Pipino, J. Funk, and R. Wang, «Journey to Data Quality, MIT Press, Cambridge, MA, 2006.», vol. 2006.

[20] N. Prat y S. E. Madnick, «Measuring Data Believability: A Provenance Approach», p. 11.

[21] B. J. Fogg *et al.*, «What makes Web sites credible?: a report on a large quantitative study», en *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '01*, Seattle, Washington, United States, 2001, pp. 61-68. doi: 10.1145/365024.365037.

[22] M. Kang, «Measuring Social Media Credibility: A Study on a Measure of Blog Credibility», p. 31, 2010.

[23] C. Dai, D. Lin, E. Bertino, y M. Kantarcioglu, «An Approach to Evaluate Data Trustworthiness Based on Data Provenance», en *Secure Data Management*, ago. 2008, pp. 82-98. doi: 10.1007/978-3-540-85259-9_6.

[24] C. Dyreson *et al.*, «A consensus glossary of temporal database concepts», *ACM SIGMOD Rec.*, vol. 23, n.º 1, pp. 52-64, mar. 1994, doi:

10.1145/181550.181560.

[25] «Acerca de LinkedIn». https://about.linkedin.com/es-es (accedido may 14, 2021).

[26] «What is Docker?», *Opensource.com*. https://opensource.com/resources/what-docker (accedido ago. 29, 2020).

[27] «Docker overview», *Docker Documentation*, ago. 26, 2020. https://docs.docker.com/get-started/overview/ (accedido ago. 29, 2020).

[28] «Welcome to Python.org», *Python.org*. https://www.python.org/doc/ (accedido may 14, 2021).

[29] «Welcome to Flask — Flask Documentation (1.1.x)». https://flask.palletsprojects.com/en/1.1.x/ (accedido may 14, 2021).

[30] «Flask · PyPI». https://pypi.org/project/Flask/ (accedido may 14, 2021).

[31] J. J. Miller, «Graph Database Applications and Concepts with Neo4j», 2013.

[32] «What is a Graph Database? - Developer Guides», *Neo4j Graph Database Platform*. https://neo4j.com/developer/graph-database/ (accedido may 14, 2021).

[33] «Graph Databases for Beginners: Why Graph Technology Is the Future», *Neo4j Graph Database Platform*, jul. 12, 2018. https://neo4j.com/blog/why-graph-databases-are-the-future/ (accedido ago. 29, 2020).

[34] «Chapter 3. Cypher - The Neo4j Developer Manual v3.2». https://neo4j.com/docs/developer-manual/3.2/cypher/ (accedido sep. 17, 2018).

[35] «Cypher Query Language - Developer Guides», *Neo4j Graph Database Platform*. https://neo4j.com/developer/cypher/ (accedido may 14, 2021).

[36] «Chapter 1. Introduction - The Neo4j Graph Algorithms User Guide v3.5». https://neo4j.com/docs/graph-algorithms/current/introduction/ (accedido may 14, 2021).

[37] «PageRank - Neo4j Graph Data Science», *Neo4j Graph Database Platform*. https://neo4j.com/docs/graph-data-science/1.5/algorithms/page-rank/ (accedido may 14, 2021).

[38] «Label Propagation - Neo4j Graph Data Science», *Neo4j Graph Database Platform*. https://neo4j.com/docs/graph-data-science/1.5/algorithms/label-propagation/

(accedido may 14, 2021).

[39] *neo4j/neo4j-javascript-driver*. Neo4j, 2021. Accedido: may 14, 2021. [En línea]. Disponible en: https://github.com/neo4j/neo4j-javascript-driver

[40] «vis.js». https://visjs.org/ (accedido may 14, 2021).

[41] «Graph Visualization Tools - Developer Guides», *Neo4j Graph Database Platform*. https://neo4j.com/developer/tools-graph-visualization/ (accedido may 14, 2021).

[42] «PFC_Sergio_Bellido_Sanchez%2FTema5_mongodb.pdf». Accedido: may 14, 2021. [En línea]. Disponible en: http://bibing.us.es/proyectos/abreproy/12037/fichero/PFC_Sergio_Bellido_Sanchez%252FTema5_mongodb.pdf

[43] «¿Qué es el sharding en MongoDB? ¿Cómo funciona el sharding en MongoDB? | ramoncarrasco.es». https://www.ramoncarrasco.es/es/content/es/kb/141/que-es-el-sharding-en-mongodb-como-funciona-el-sharding-en-mongodb (accedido may 14, 2021).

[44] «Streaming message types». https://developer.twitter.com/en/docs/tweets/filter-realtime/guides/streaming-message-types.html (accedido abr. 22, 2019).

[45] «Twython — Twython 3.6.0 documentation». https://twython.readthedocs.io/en/latest/ (accedido abr. 22, 2019).

[46] «Streaming API — Twython 3.8.0 documentation». https://twython.readthedocs.io/en/latest/usage/streaming_api.html (accedido nov. 22, 2020).

[47] «POST statuses/filter | Docs | Twitter Developer». https://developer.twitter.com/en/docs/twitter-api/v1/tweets/filter-realtime/api-reference/post-statuses-filter (accedido nov. 22, 2020).

[48] «GET followers/list». https://developer.twitter.com/en/docs/twitter-api/v1/accounts-and-users/follow-search-get-users/api-reference/get-followers-list (accedido may 14, 2021).

[49] «GET statuses/user_timeline | Twitter Developer». https://developer.twitter.com/en/docs/twitter-api/v1/tweets/timelines/api-reference/get-statuses-user_timeline (accedido may 14, 2021).

[50] Intellica.AI, «Comparison of different Word Embeddings on Text Similarity —

A use case in NLP», *Medium*, oct. 04, 2019. https://intellica-ai.medium.com/comparison-of-different-word-embeddings-on-text-similarity-a-use-case-in-nlp-e83e08469c1c (accedido may 14, 2021).

[51] «ResearchGate | Find and share research». https://www.researchgate.net/ (accedido may 14, 2021).

[52] «About Facebook». https://about.facebook.com/ (accedido may 14, 2021).

[53] «Tagged (website) - Wikipedia». https://en.wikipedia.org/wiki/Tagged_(website) (accedido may 14, 2021).

[54] «YY.com - YY.com - other.wiki». https://es.other.wiki/wiki/YY.com (accedido may 14, 2021).

[55] «¡Bienvenido! | VK». https://vk.com/?lang=es (accedido may 14, 2021).

[56] «Acerca de WhatsApp», *WhatsApp.com*. https://www.whatsapp.com/about/ (accedido may 14, 2021).

[57] «WeChat», *Wikipedia, la enciclopedia libre*. abr. 09, 2021. Accedido: may 14, 2021. [En línea]. Disponible en: https://es.wikipedia.org/w/index.php?title=WeChat&oldid=134625055

[58] «QQ - Wikipedia, la enciclopedia libre». https://es.wikipedia.org/wiki/QQ (accedido may 14, 2021).

[59] «Acerca de Skype | Contacto | ¿Qué es Skype?» https://www.skype.com/es/about/ (accedido may 14, 2021).

[60] «Acerca de Viber | Viber». https://www.viber.com/es/about/ (accedido may 14, 2021).

[61] «Snapchat, ¡la forma más rápida de compartir un momento!» https://www.snapchat.com/l/es/ (accedido may 14, 2021).

[62] «LINE Corporation», *LINE Corporation*. https://linecorp.com/en/company/info (accedido may 14, 2021).

[63] «Preguntas frecuentes», *Telegram*. https://telegram.org/faq (accedido may 14, 2021).

[64] «BlackBerry Messenger», *Wikipedia, la enciclopedia libre*. ene. 20, 2021. Accedido: may 14, 2021. [En línea]. Disponible en: https://es.wikipedia.org/w/index.php?title=BlackBerry_Messenger&oldid=132567

689

[65] «Kakao», *kakaocorp.com*. //www.kakaocorp.com/kakao/introduce/kakaoCulture (accedido may 14, 2021).

[66] «DailyStrength: Online Support Groups and Forums». https://www.dailystrength.org/ (accedido ene. 07, 2019).

[67] «Expert Health Information - Questions and Answers», *Sharecare*. https://www.sharecare.com/ (accedido ene. 07, 2019).

[68] «MedHelp - Health community, health information, medical questions, and medical apps». https://www.medhelp.org/ (accedido ene. 07, 2019).

[69] «Acerca de Tumblr | Tumblr». https://www.tumblr.com/about (accedido may 14, 2021).

[70] «What is Qzone? - Definition from WhatIs.com», *WhatIs.com*. https://whatis.techtarget.com/definition/Qzone (accedido may 14, 2021).

[71] «Sina Weibo», *Wikipedia, la enciclopedia libre*. ene. 16, 2020. Accedido: may 14, 2021. [En línea]. Disponible en: https://es.wikipedia.org/w/index.php?title=Sina_Weibo&oldid=122803474

[72] «About Twitter | Our company and priorities». https://about.twitter.com/en.html (accedido may 14, 2021).

[73] «Blogger.com - Crea un blog atractivo y original. Es fácil y gratuito.» https://www.blogger.com (accedido may 14, 2021).

[74] «Baidu», *Wikipedia, la enciclopedia libre*. may 02, 2021. Accedido: may 14, 2021. [En línea]. Disponible en: https://es.wikipedia.org/w/index.php?title=Baidu&oldid=135245593

[75] «Taringa!», *Wikipedia, la enciclopedia libre*. abr. 15, 2021. Accedido: may 14, 2021. [En línea]. Disponible en: https://es.wikipedia.org/w/index.php?title=Taringa!&oldid=134774414

[76] «About Reddit». https://www.reddit.com/r/aboutreddit/ (accedido may 14, 2021).

[77] «About - Foursquare». https://es.foursquare.com/about (accedido may 14, 2021).

[78] «Información sobre YouTube - YouTube».

https://www.youtube.com/intl/es/about/ (accedido may 14, 2021).

[79] «About Us • Instagram». https://www.instagram.com/about/us/ (accedido may 14, 2021).

[80] «Todo acerca de Pinterest», *Pinterest Help*. https://help.pinterest.com/es/guide/all-about-pinterest (accedido may 14, 2021).

[81] «Acerca de Last.fm | Last.fm». https://www.last.fm/es/about (accedido may 14, 2021).

[82] «About SoundCloud on SoundCloud», *SoundCloud*. https://soundcloud.com/pages/contact (accedido may 14, 2021).

[83] «What is Spotify?», *Spotify*. https://support.spotify.com/us/article/what-is-spotify/ (accedido may 14, 2021).

[84] «About Us». https://es.slideshare.net/about (accedido may 14, 2021).

[85] «Developer Guide | Geocoding API», *Google Developers*. https://developers.google.com/maps/documentation/geocoding/intro (accedido ene. 07, 2019).

[86] «What is Elasticsearch: Tutorial for Beginners», *Logz.io*, may 28, 2019. https://logz.io/blog/elasticsearch-tutorial/ (accedido may 14, 2021).

[87] «MedlinePlus - Información de Salud de la Biblioteca Nacional de Medicina». https://medlineplus.gov/spanish/ (accedido may 14, 2021).

[88] «PubMed», *PubMed*. https://pubmed.ncbi.nlm.nih.gov/ (accedido may 14, 2021).

[89] «About Embase - Biomedical research | Elsevier». https://www.elsevier.com/solutions/embase-biomedical-research (accedido may 14, 2021).

[90] O. of the Commissioner, «MedWatch: The FDA Safety Information and Adverse Event Reporting Program». https://www.fda.gov/Safety/MedWatch/default.htm (accedido ene. 07, 2019).

[91] «Coding Symbols for a Thesaurus of Adverse Reaction Terms - Summary | NCBO BioPortal». http://bioportal.bioontology.org/ontologies/COSTART (accedido ene. 07, 2019).

[92] «UMLS Metathesaurus - CHV (Consumer Health Vocabulary) - Source Representation».

https://www.nlm.nih.gov/research/umls/sourcereleasedocs/current/CHV/sourcerepresentation.html (accedido may 14, 2021).

[93] H. Canada y H. Canada, «MedEffect Canada», *aem*, jul. 13, 2011. https://www.canada.ca/en/health-canada/services/drugs-health-products/medeffect-canada.html (accedido ene. 07, 2019).

[94] «Unified Medical Language System (UMLS)». https://www.nlm.nih.gov/research/umls/ (accedido ene. 07, 2019).

[95] «MedDRA |». https://www.meddra.org/ (accedido ene. 07, 2019).

[96] jjperez_admin, «Web Of Science», *FECYT*, nov. 12, 2014. https://www.fecyt.es/es/recurso/web-science (accedido ene. 07, 2019).

[97] «MapReduce», *Wikipedia, la enciclopedia libre*. ene. 04, 2019. Accedido: ene. 07, 2019. [En línea]. Disponible en: https://es.wikipedia.org/w/index.php?title=MapReduce&oldid=113059090

[98] «Apache Hadoop», *Wikipedia, la enciclopedia libre*. oct. 10, 2018. Accedido: ene. 07, 2019. [En línea]. Disponible en: https://es.wikipedia.org/w/index.php?title=Apache_Hadoop&oldid=111182265

[99] «Apache Spark», *Wikipedia, la enciclopedia libre*. nov. 30, 2018. Accedido: ene. 07, 2019. [En línea]. Disponible en: https://es.wikipedia.org/w/index.php?title=Apache_Spark&oldid=112362316

[100] «Apache Flink», *Wikipedia*. dic. 03, 2018. Accedido: ene. 07, 2019. [En línea]. Disponible en: https://en.wikipedia.org/w/index.php?title=Apache_Flink&oldid=871852581

[101] «Apache Mahout». https://mahout.apache.org/ (accedido ene. 07, 2019).

[102] «MLlib | Apache Spark». https://spark.apache.org/mllib/ (accedido ene. 07, 2019).

[103] «Apache Flink 1.7 Documentation: Apache Flink Documentation». https://ci.apache.org/projects/flink/flink-docs-stable/ (accedido ene. 07, 2019).

[104] «Home», *Open Source Leader in AI and ML*. https://www.h2o.ai/ (accedido ene. 07, 2019).

Tabla de Términos

La siguiente tabla (Tabla 40) representa, una recopilación de términos usados en el documento. Se puede encontrar terminología en español e inglés; en caso de esta última se agrega su traducción al español.

Términos Similares	Concepto	Traducción
Accessibility	Los datos pueden ser accedidos en un contexto específico de uso.	Accesibilidad
API	Es un conjunto de subrutinas, funciones y procedimientos que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción.	API
Authentic	Acreditado como cierto y verdadero por los caracteres o requisitos que en ello concurren.	Auténtico
Authority	Producir sobre algo ciertos efectos.	Autoridad o influencia
Availability	El grado en que los datos tienen atributos que le permiten ser leídos e interpretados por los usuarios, y se expresan en los idiomas, símbolos y unidades apropiados en un contexto específico de uso.	Disponibilidad
Completeness	Los datos de sujeto asociados con una entidad tienen valores para todos los atributos esperados y las instancias relacionadas con la entidad en un contexto específico de uso.	Completitud
Compliance	El grado en que los datos tienen atributos que se adhieren a las normas, convenciones o regulaciones vigentes y reglas similares relacionadas con los datos en un contexto específico de uso.	Conformidad
Confidentiality	El grado en que los datos tienen atributos que aseguran que solo sea accesible e interpretable por los usuarios autorizados en un contexto específico de uso.	Confidencialidad
Consistency, consistent	El grado en que los datos tienen atributos libres de contradicción y son coherentes con otros datos en un	Consistencia

	contexto específico de uso.	
Consistency Over Sources	Los datos son coherentes teniendo en cuenta diferentes fuentes.	Consistencia sobre las fuentes
Consistency Over Time	Los datos son coherentes con respecto a un cierto tiempo.	Consistencia sobre el tiempo
Correctness	El grado en el que los datos tienen atributos que representan correctamente el verdadero valor del atributo deseado de un concepto o evento es un contexto específico de uso.	Exactitud
Credibility, believability	El grado en que los datos tienen atributos que los usuarios consideran verdaderos y creíbles en un contexto específico de uso.	Credibilidad
Currentness	El grado en que los datos tienen atributos que tienen la vigencia adecuada en un contexto específico de uso.	Actualidad
Data Conflict	Elementos que comprometen la confiabilidad de los demás.	Conflicto de datos
Data Deduction	Inferir una determinada conclusión de los datos.	Deducción de datos
Data Similarity	Elementos de soporte mutuo.	Similitud de los datos
Dimension, Attribute	La calidad es descrita por atributos que ayudan a calificar los datos, los cuales son denominados dimensiones.	Dimensión, Atributo
Efficiency	El grado en que los datos tienen atributos que pueden procesarse y proporcionar los niveles de rendimiento esperados al usar las cantidades y los tipos de recursos apropiados en un contexto de uso específico.	Eficiencia
Expertise	Se define por términos tales como experto, experimentado, competente.	Experiencia

Factor	Cuando hablamos de un aspecto particular de una dimensión esto se denota como factor de calidad, es decir una dimensión es un conjunto de factores de calidad que tienen el mismo propósito.	Factor
Fair	Preciso, medible.	Justo
Focused	Entender los datos desde una perspectiva de datos y no desde una perspectiva empresarial.	Enfocado
Informative	Que brinda información útil. Datos que a través de su análisis permiten predecir patrones y comportamientos.	Informativo
Insightful	Comprensión profunda que se obtiene al analizar la información.	Perspicaz
Intrinsic	Que es propio o característico de la cosa que se expresa por sí misma y no depende de las circunstancias. (Wikipedia)	Intrínseco
Lineage	El linaje de los datos es el historial de los datos, incluido el lugar donde los datos han viajado a lo largo de su existencia.	Linaje
Measure	Expresión del resultado de una medición. (RAE)	Medida
Method of measurement	Proceso que implementa una métrica.	Método de medición
Metrics	Cuando hablamos de métrica nos referimos al instrumento utilizado para definir la forma de medir los factores de calidad.	Métrica
Path Similarity	No varía en la trayectoria.	Similitud de ruta
Portability	El grado en que los datos tienen atributos que le permiten ser instalado, reemplazado o movido de un sistema a otro preservando la calidad existente en un contexto específico de uso.	Portabilidad
Possibitily	Hasta qué punto el valor de un dato es posible	Posibilidad
Precision, Accurancy	El grado en que los datos tienen atributos que son exactos o que proporcionan discriminación en un contexto específico de uso.	Precisión

Provenance	La procedencia de un dato es un conjunto de metadatos que contienen descripciones de entidades y actividades involucradas en la producción y entrega de este.	Procedencia
Reasonableness	Hasta qué punto el valor de un dato es probable	Razonabilidad
Recoverability	El grado en que los datos tienen atributos que le permiten mantener y preservar un nivel específico de operaciones y calidad, incluso en caso de falla, en un contexto específico de uso.	Recuperabilidad
Messaging Network	Red donde se intercambian mensajes.	Red de Mensajería
Social Network	Plataforma digital de comunicación global que pone en contacto a gran número de usuarios. RAE	Red Social
Reliability	Entendido como la completitud y precisión de los datos.	Fiabilidad
Reputation	Enjuiciamiento hecho por el usuario para determinar la integridad de una fuente.	Reputación
Sentiment Analysis	Pretende saber si una palabra o una opinión acerca del producto y/o servicio, de una marca es positiva o negativa para la empresa y si es capaz de incitar a la compra o generar impactos influyentes.	Análisis de los sentimientos
Social Analytics	Permite integrar y analizar los datos no estructurados, que se encuentran en el correo electrónico, la mensajería instantánea, los portales Web, los blogs y otros medios sociales, usando las herramientas de obtención de datos existentes	Analítica social
Social Authority	Es la medida de la influencia de una persona en las redes sociales.	Autoridad Social
Temporal believability	La idea general es que el valor de un dato estimado (calculado de antemano) es más confiable a medida que se acerca al tiempo válido, sobre todo al tiempo válido final.	Credibilidad temporal
Temporality	Transcurso de tiempo en el que los datos son válidos	Temporalidad
Traceability	El grado en que los datos tienen atributos que proporcionan una pista de auditoría de acceso a los datos y de cualquier cambio realizado en los datos en	Trazabilidad

	un contexto específico de uso.	
Transaction time	Valor del dato como el tiempo de ejecución del proceso que generó ese dato en el caso de ser un valor de tipo resultado.	Tiempo de transacción
Trustworthiness	Es definido por los términos bien intencionado, veraz, imparcial, pronto y confiabilidad.	Veracidad
Understandability	El grado en que los datos tienen atributos que le permiten ser leídos e interpretados por los usuarios, y se expresan en los idiomas, símbolos y unidades apropiados en un contexto específico de uso.	Comprensibilidad
Valid time	El tiempo válido de un hecho es el tiempo donde el hecho es verdadero en la realidad modelada.	Tiempo válido
Variety	La cual se refiere a la heterogeneidad de la adquisición de los datos, de la representación y de su interpretación semántica.	Variedad
Velocity	Se refiere a la tasa de obtención de los datos y al tiempo de vida útil de los mismos.	Velocidad
Verifiability	Grado y facilidad con que se puede verificar la corrección de información, permite decidir si aceptar la información provista por la misma o no.	Verificabilidad
Volume	La cual refiere al tamaño de los datos, el cual aumenta a un ritmo acelerado hoy en día.	Volumen
RDF	El Marco de Descripción de Recursos es una recomendación del W3C (World Wide Web Consortium, es un consorcio internacional que genera recomendaciones y estándares) que define un lenguaje para describir recursos. Fue diseñado para describir Recursos web como páginas web.	RDF
Web Analytics	Análisis de los datos que fluyen a través de sitios y páginas Web.	Análisis Web

Tabla 40. Tabla de Términos

Anexo

En esta sección se realiza una revisión de herramientas y redes sociales investigadas y diferente información que no es fundamental para entender el proyecto, pero se cree que aporta.

1- Redes Sociales de Uso General

Facebook[52]

Red social para compartir imágenes, pensamientos, videos, chatear, crear grupos, páginas y eventos. Hoy por hoy es la red social más popular en prácticamente todo el mundo, excepto en China. Ideal para conectar con clientes potenciales a través de páginas de empresa o fidelizar clientes a través de grupos.

Tagged[53]

Tagged es una mezcla de funciones de distintas redes sociales. Tagged se diseñó para ayudar a los usuarios a conocer gente nueva con intereses similares en un corto período de tiempo.

YY[54]

YY es una plataforma de comunicación social interactiva con actividades en grupo y en tiempo real a través de la voz, texto y video. Cuenta con una moneda virtual que los usuarios pueden ganar a través de actividades como el karaoke o la creación de videotutoriales.

VK[55]

Es la red social conocida como el Facebook ruso, aunque su uso es más sencillo e intuitivo. Permite a los usuarios crear mensajes privados, actualizaciones de estado, compartir fotos, crear grupos, páginas y eventos públicos, tal y como sucede en Facebook.

2- Redes Sociales de Uso Específico

2.1- Mensajería

WhatsApp[56]

WhatsApp es una aplicación de mensajería que permite enviar y recibir mensajes sin pagar. Además de la mensajería básica, los usuarios de WhatsApp pueden crear grupos y enviar entre ellos un número ilimitado de imágenes, vídeos y mensajes de audio.

WeChat[57]

Es un servicio de mensajería chino de texto móvil y servicio de comunicación de mensajes de voz creado por Tencent lanzado en 2011. Es la competencia de WhatsApp y tiene más de 40 millones de usuarios fuera de China.

QQ[58]

QQ vendría a ser el equivalente al Messenger, Facebook y Twitter juntos, además de ofrecer otros servicios: puedes enviar un e-mail (QQMail), disponer de un disco duro virtual, escribir un blog (QQZone), un microblog (Tencent Weibo), (QQYinyue), comprar online (Paipai) y jugar en red (QQYouxi). También permite reservar viajes, buscar pareja (QQTongchang) o mantener a una mascota virtual, al estilo del Tamagotchi.

Skype[59]

Millones de personas y empresas usan Skype para hacer llamadas y videollamadas gratis individuales y grupales, enviar mensajes instantáneos y compartir archivos con otras personas que usan la misma red.

Viber[60]

Viber es una aplicación para dispositivos móviles con la que se puede enviar mensajes de texto y hacer llamadas telefónicas a cualquier otro usuario Viber, gratis.

Snapchat[61]

Snapchat es una aplicación móvil dedicada al envío de archivos, los cuales *desaparecen* del dispositivo del destinatario entre uno y diez segundos después de haberlos visto.

Line[62]

Aplicación japonesa para mensajería en móviles, competencia de WhatsApp. Además, ofrece tomar fotografías gracias a Line Camera, que permite hacer fotos e introducir *stickers* con mensajes. A su vez se puede confeccionar tarjetas de felicitación de cumpleaños y sirve para crear dibujos que se pueden enviar a los contactos.

Telegram[63]

Servicio de mensajería estrenado en 2013. Tiene la particularidad de que puedes crear chats secretos. Es una de las aplicaciones de mensajería instantánea que más énfasis ha puesto en la seguridad y privacidad.

BBM[64]

Es una aplicación de mensajería creada por BlackBerry Limited. (BBRY LTD) para sus teléfonos inteligentes Blackberry y a partir del 21 de octubre de 2013, para teléfonos inteligentes con sistema operativo iOS, Android, Windows Phone.

KakaoTalk[65]

Es una aplicación de mensajería multiplataforma que permite enviar y recibir gratuitamente mensajes a través de teléfonos inteligentes y realizar llamadas gratuitas. Está disponible en los sistemas operativos iOS, Android, Bada OS, BlackBerry, Windows Phone, y en computadoras personales. Además, los usuarios pueden compartir contenido multimedia como fotos, videos, mensajes de voz y enlaces URL.

2.2- Medicina

DailyStrength [66]

Red social centrada en grupos de ayuda donde los usuarios se brindan apoyo emocional mutuamente compartiendo sus luchas y éxitos.

ShareCare [67]

Plataforma de salud que provee a los consumidores de información y programas para mejorar su salud.

MedHelp [68]

MedHelp se asocia con médicos de hospitales e instituciones de investigación médica para generar foros de discusión en línea sobre más de 80 temas relacionados con cuidados de salud.

2.3- Recursos Humanos

LinkedIn[25]

LinkedIn es una red social para profesionales, especialmente interesante para negocios B2B (servicios de empresas a empresas). Es importante destacar que un 45% de los usuarios de esta red son decision makers (tomadores de decisiones).

2.4- Microblogging

Tumblr[69]

Tumblr es una plataforma de microblogging que permite seguir las publicaciones de otros.

Los usuarios de Tumblr pueden crear publicaciones que incluyen texto, fotos, vídeos, vínculos, frases, chat y audio. Estas publicaciones pueden ser creadas a través de un ordenador, enviadas por correo electrónico, mensajes de texto o por teléfono. Además de la plataforma de blog común, Tumblr incluye un elemento interactivo que anima a los usuarios a compartir, intercambiar y seguir las publicaciones de otros.

Qzone[70]

Es una red social china, creada en 2005. Permite a los usuarios escribir blogs, diarios digitales, enviar y alojar fotos, música etc. La mayoría de sus servicios ofertados no son gratuitos, aunque existe la posibilidad de adquirir un pase denominado diamante canario que permite el acceso a casi la totalidad de sus aplicaciones. Ofrece un chat de mensajería instantánea que llega a tener al día a 50 millones de usuarios conectados en línea.

Weibe[71]

Es el equivalente en China a Twitter, pero ahora incluye también funcionalidades parecidas a Facebook.

Twitter[72]

Ofrece un sencillo servicio que puede ayudar a una empresa a estar en contacto con seguidores y clientes a través de su servicio de mensajería con un máximo de 140 caracteres por mensaje.

En cualquier caso, Twitter ofrece mucho más que la posibilidad de envío de mensajes y es la red con mayor número de aplicaciones. Para las empresas quizá sea la más desaprovechada de las *grandes redes populares.*

Blogger[73]

Es un servicio creado por Pyra Labs y adquirido por Google en el año 2003, que permite crear y publicar una bitácora en línea. Para publicar contenidos, el usuario no tiene que escribir ningún código o instalar programas de servidor o de scripting. Las opciones que ofrecen son gratuitas y sencillas de utilizar.

Baidu Tieba[74]

Baidu Tieba es una plataforma de acceso libre que recoge foros de debate ofreciendo a sus 300 millones de usuarios un entorno para compartir opiniones.

Taringa[75]

Taringa es una comunidad virtual en la que los usuarios pueden compartir todo tipo de información por medio de mensajes a través de un sistema colaborativo de interacción.

Reddit[76]

Si tienes un artículo interesante, vídeo, imagen o cualquier otra cosa, subirlo a Reddit con un título interesante podría dar lugar a tráfico importante de la página de Reddit.

2.5- Geolocalización

Foursquare[77]

La idea principal de la red es marcar (check-in) lugares específicos donde uno se encuentra e ir ganando puntos por descubrir nuevos lugares; la recompensa son las *Badges*, una especie de medallas, y las *Alcaldías* (Mayorships), que son ganadas por las personas que más hacen *check-ins* en un cierto lugar en los últimos 60 días.

2.6- Video

YouTube[78]

Es un sitio web dedicado a compartir vídeos. Presenta una variedad de clips de películas, programas de televisión y vídeos musicales, así como contenidos amateur como videoblogs y YouTube Gaming.

Un vídeo bien hecho con el título y el contenido adecuado puede tener un enorme impacto viral para una marca, sobre todo si el vídeo alcanza las páginas más vistas.

Instagram[79]

Red social para compartir fotografías y vídeos de corta duración. Los usuarios utilizan mucho los hashtags acompañando las imágenes y vídeos que comparten. El sistema de seguimiento es similar al de Twitter. Aunque fue creada en 2010, ha tenido un crecimiento espectacular en muy poco tiempo.

2.7- Imágenes

Pinterest[80]

Pinterest es una red social para compartir imágenes que permite a los usuarios crear y administrar, en tableros personales temáticos, colecciones de imágenes.

Los usuarios de Pinterest pueden subir, guardar, ordenar y administrar imágenes, conocidos como pins, y otros contenidos multimedia (vídeos, por ejemplo) a través de colecciones llamadas pinboards o tableros. Los pinboards son generalmente personalizados, esto quiere decir que los pins o

publicaciones pueden ser fácilmente organizados, clasificados y encontrados por otros usuarios.

2.8- Música

Last.fm[81]

Es una red social, una radio vía Internet y además un sistema de recomendación de música que construye perfiles y estadísticas sobre gustos musicales, basándose en los datos enviados por los usuarios registrados. Algunos de estos servicios son de pago, pero aún existen países donde sigue siendo gratuito. En la radio se puede seleccionar las canciones según las preferencias personales (de acuerdo con un algoritmo y a las estadísticas) o la de otros usuarios.

SoundCloud[82]

SoundCloud es una red social para músicos, en la cual se les proporcionan canales para la distribución de su música. SoundCloud analiza la canción con el objetivo de que cualquiera que la esté escuchando pueda dejar su comentario en un momento determinado del audio.

Spotify[83]

Spotify es una aplicación empleada para la reproducción de música.

2.9- Presentaciones

Slideshare[84]

SlideShare es un sitio de alojamiento de diapositivas que ofrece a los usuarios la posibilidad de subir y compartir en público o en privado presentaciones de diapositivas.

GoogleMaps API (Geocodificación) [85]

Provee un camino directo para acceder a los diferentes servicios Google Maps vía http request, por ejemplo, permite obtener coordenadas geográficas a partir de direcciones.

3- Herramientas

En esta sección se hace una revisión de herramientas y tecnologías relacionadas con las temáticas investigadas, con la intención de generar conocimiento sobre las mismas para tomar decisiones acertadas más adelante.

3.1- ElasticSearch

Elasticsearch está basado en Apache Lucene, es uno de los buscadores de texto completo más importantes de Internet. Elasticsearch tiene ciertas ventajas con respecto a otros motores, por ejemplo, la búsqueda se realiza mediante un índice, pero en lugar de examinar todos los documentos, el programa comprueba un índice de documentos que se ha creado previamente en el que todo el contenido se almacena de forma preparada. Este proceso requiere mucho menos tiempo que la consulta de todos los documentos.

Es más amigable, facilita los primeros pasos; es posible construir un servidor de búsqueda estable en poco tiempo que también se puede distribuir fácilmente entre varios equipos.

Utilizando el llamado sharding, varios nodos (diferentes servidores) se unen para formar un clúster: Elasticsearch desglosa el índice y distribuye las partes individuales (shards) entre varios nodos, dividiendo así la carga de cálculo. Para proyectos grandes, la búsqueda de texto completo es mucho más estable y, en caso de buscar mayor seguridad, también se pueden copiar los fragmentos a varios nodos.

Elasticsearch se basa en Java y produce los resultados en formato JSON y los entrega a través de un servicio web REST. La API facilita la integración de la función de búsqueda en un sitio web. Ofrece con Kibana, Beats y Logstash (conocidos juntos como Elastic-Stack) servicios adicionales que se pueden utilizar para analizar la búsqueda de texto completo. La empresa Elastic, que está detrás del motor, también ofrece servicios de pago, como el cloud hosting.
Es open source, lo que significa que, una vez se ha implementado la función

de búsqueda de texto completo permite que pueda personalizarse por completo.

Además, Elasticsearch es popular por su manejo de datos dinámicos: gracias a un procedimiento especial de almacenamiento en caché, hace posible que los cambios no tengan que ser introducidos en la caché global. En su lugar, es suficiente con cambiar un pequeño segmento, lo cual lo hace mucho más flexible.

Términos básicos:

- Index: una búsqueda en Elasticsearch nunca arroja el contenido como respuesta, sino el índice, en el cual se almacenan, ya preparados, todos los contenidos de todos los documentos. De esta forma la búsqueda requiere muy poco tiempo. Es el llamado inverted index (índice invertido): para cada término de búsqueda se indica el lugar donde se puede encontrar dicho término.
- Document: la salida para el índice son los documentos, en los cuales aparecen los datos. No tienen que ser necesariamente textos completos (por ejemplo, artículos de blogs), es suficiente que se trate de archivos con información.
- Field: un documento, a su vez, consta de varios campos. Además del campo de contenido propiamente dicho, hay otros metadatos que también forman parte de un documento. Por ejemplo, Elasticsearch puede utilizarse para buscar metadatos sobre el autor o el momento de la creación.

El proceso de preparación de datos se le llama tokenizing; en el cual un algoritmo crea términos individuales a partir de un texto completo. No existen las palabras, sino que para él un texto consiste en una larga cadena de caracteres y una letra tiene el mismo valor que un espacio. Para que un texto sea formateado con sentido, primero debe ser dividido en tokens. Para ello se toma el espacio en blanco (whitespace, es decir, espacios y líneas en blanco) como marcador para palabras. Durante la preparación, también se normaliza: todas las palabras se escriben en minúsculas y los signos de puntuación se

ignoran. Elasticsearch ha adoptado todos estos métodos de Apache Lucene.[86]

3.2- Recuperación de Información

MedLine[87]

Base de datos de bibliografía médica.

PubMed[88]

Motor de búsqueda de libre acceso a la base de datos MedLine.

Embase[89]

Base de datos de publicaciones farmacológicas y biomédicas.

FDA AERS [90]

Es una base de datos que contiene reportes de reacciones adversas, de errores en medicación y quejas en cuanto a calidad de productos que fueron registrados al FDA (Food and Drug Administration). Se reportan en MedWatch.

COSTART [91]

Sirve para codificar, completar y obtener informes de reacciones adversas posteriores a la comercialización.

CHV (Consumers Health Vocabulary)[92]

Base de datos creada para mapear términos usados por personas con términos médicos.

MedEffect Canadá [93]

Acceso a reacciones adversas de diferentes medicamentos.

UMLS (Unfied Medical Language System) [94]

Vocabulario biomédico.

SIDER (Side Effects Resource) [95]

Base de conocimiento que usa MedDRA (Terminología médica para productos internacionales).

WebOfScience [96]

Plataforma basada en tecnología Web que recoge las referencias de las principales publicaciones científicas de cualquier disciplina de conocimiento.

3.3- Herramientas para Manejo de BigData

MapReduce [97]

Procesamiento de datos masivos mediante el particionamiento de ficheros de datos. Diseñada por Google (2003). Limitaciones especialmente en escenarios que requieren la reutilización de datos como procesos iterativos, procesamiento de grafos, etc.

Apache Hadoop [98]

Versión de código abierto de MapReduce.

Apache Spark [99]

Alternativa que intenta dar solución a las limitaciones de MapReduce/Hadoop. Más eficiente que Hadoop. Perfecto para procesos iterativos. Emula el procesamiento en tiempo real usando mini-lotes de datos.

Apache Flink [100]

Plataforma distribuida para flujos de datos que también puede trabajar con datos secuenciales. Muestra un buen rendimiento en el procesamiento de datos en sistemas de baja latencia.

3.4- Herramientas para la Analítica de Datos Masivos

Mahout [101]

Biblioteca que ofrece implementaciones basadas en Hadoop MapReduce para varias tareas de analítica como agrupamiento, clasificación, o filtrado colaborativo.

MLlib [102]

Biblioteca de Aprendizaje Automático que contiene varias utilidades estadísticas y algoritmos de aprendizaje. Nacida junto con Spark.

FlinkML [103]

Biblioteca nativa para análisis distribuido de datos Flink. Incluye algoritmos escalables para tareas como clasificación, agrupamiento, preprocesamiento de datos.

H2O [104]

Plataforma de código abierto para análisis de BigData Selección de atributos escalables: Algoritmo Fast-mRMR

4- Funcionalidades detalladas del prototipo

En esta sección se detallan las diferentes funcionalidades que permite realizar el prototipo implementado, para ambas webs, ***FakeBusters Server*** y ***FakeBusters Web***.

4.1- FakeBusters Server

En ***FakeBusters Server*** (accesible desde el puerto 8080) se encuentran disponibles tres funcionalidades relativas a la obtención de los datos a través de la API de Twitter: *Get tweets by user*, *Get Followers tweets by user*, *Get tweets by streaming*.

Get tweets by user permite a partir de un nombre de usuario válido, obtener desde Twitter sus datos, así como sus tweets y guardarlos en las respectivas bases de datos, para ello utiliza la función detallada en 3.1.2.

La funcionalidad *Get Followers tweets by user* recibe como datos de entrada un nombre de usuario válido y existente en el sistema y un valor entero correspondiente a un nivel de profundidad. A partir de estos parámetros de entrada obtiene de Twitter los datos de sus seguidores y de los seguidores de éstos últimos (según sea el nivel), por medio de la función previamente descrita en 3.1.3.

Como tercera funcionalidad se encuentra *Get tweets by streaming* que obtiene datos de tweets (y sus respectivos usuarios) en tiempo real, a partir de la búsqueda por streaming explicada en 3.1.1.

A continuación (Figura 7) se muestra un ejemplo de la interfaz del server, en este caso *Get Follower's Tweets By User* (siendo las otras dos funcionalidades similares).

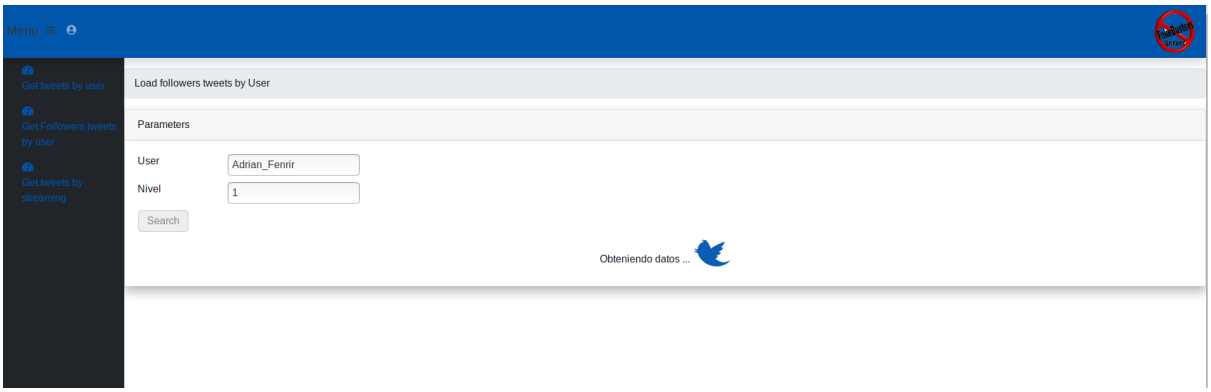

Figura 7. Interfaz ejemplo funcionalidad FakeBusters Server.

4.2- FakeBusters Web

En ***FakeBusters Web*** (accesible desde el puerto 5000) se pueden encontrar las métricas de Usuario y Clip descritas anteriormente y calcularlas por separado tanto en MongoDB, como en Neo4j. Por otro lado, también se pueden combinar las mismas y asignarles un peso y una base de ejecución específicas, para así obtener una medida de credibilidad general.

Como funcionalidades complementarias, se puede ver el historial de las métricas calculadas, tanto específicas como generales, cambiar algunos parámetros que se necesitan para el cálculo de métricas y obtener información de tweets guardados en la base de datos usando diferentes criterios.

4.2.1- Parámetros

Para el cálculo de las métricas *Antiquity* y *Commented*, son necesarias las constantes *twitter_foundation_date* y *statuses_max_ref* respectivamente; en el menú parámetros es posible definir estas constantes Figura 8. A su vez, se

define el parámetro *credibility_range* para indicar si un usuario es creíble o no.

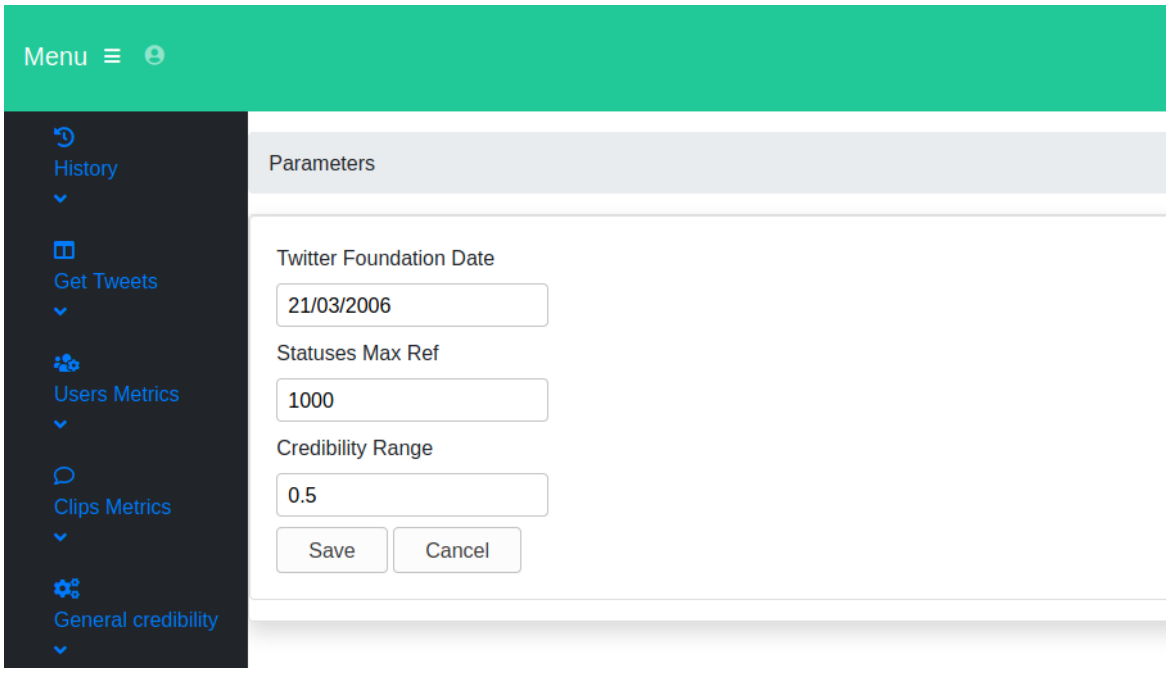

Figura 8 - Pantalla Parameters

4.2.2- Historial

Cada vez que se ejecuta una métrica ya sea específica o general la información del cálculo y los valores de entrada es guardada en MongoDB y se puede visualizar bajo los submenús debajo del menú *History*.

Data User Metrics muestra para un usuario dado el resultado de las métricas ejecutadas sobre él, tanto en MongoDB, como en Neo4j; y la fecha del último cálculo de estas, como se muestra en la Figura 9. Si una métrica es calculada más de una vez sobre un mismo usuario se guarda siempre el último cálculo y la fecha de este.

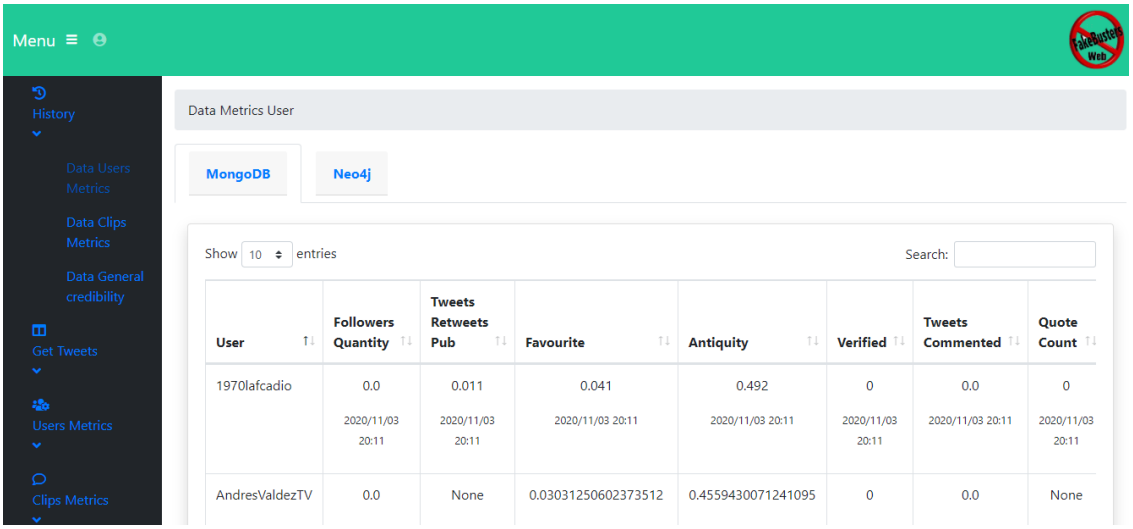

Figura 9 - Pantalla Data User Metrics

Data Clip Metrics muestra para un usuario, tweet o par de tweets (tweet source, tweet target) dados el resultado de las métricas de clips ejecutadas sobre ellos. Al igual que las métricas de usuario, si una métrica es ejecutada con igual entrada, se guarda la última ejecución de esta y se muestra la fecha, representado en la Figura 10.

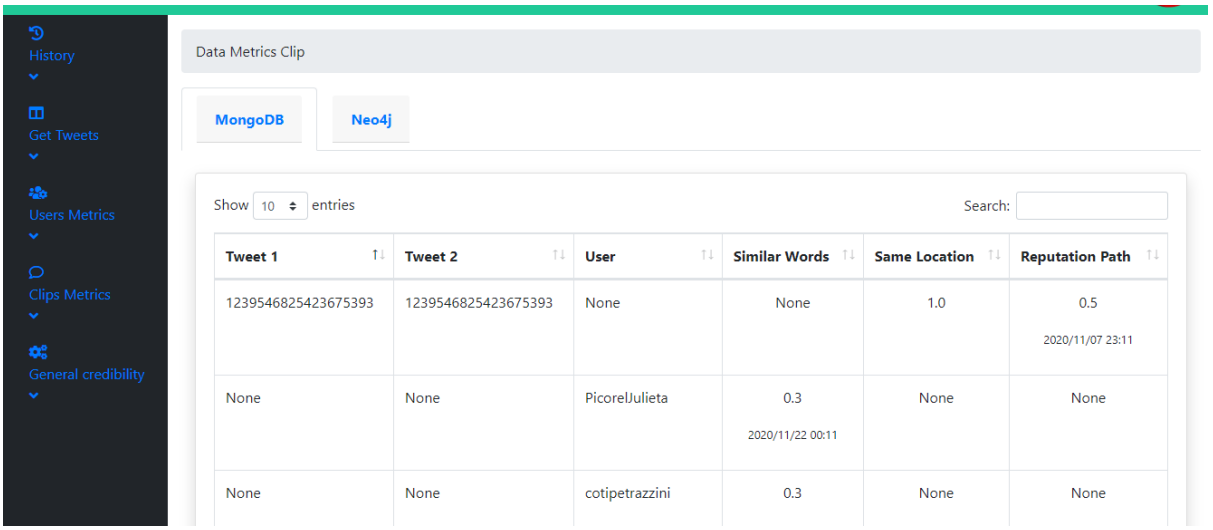

Figura 10 - Pantalla Data Clip Metrics

Data History Metrics muestra las corridas de Credibilidad General ejecutadas, la fecha de ejecución de esta, los datos de entradas utilizados, el

valor resultado de la ejecución y los valores de pesos especificados para la corrida.

Si una métrica no es seleccionada o el valor de su peso es 0, en la columna correspondiente a la métrica se especificará un 0. Figura 11

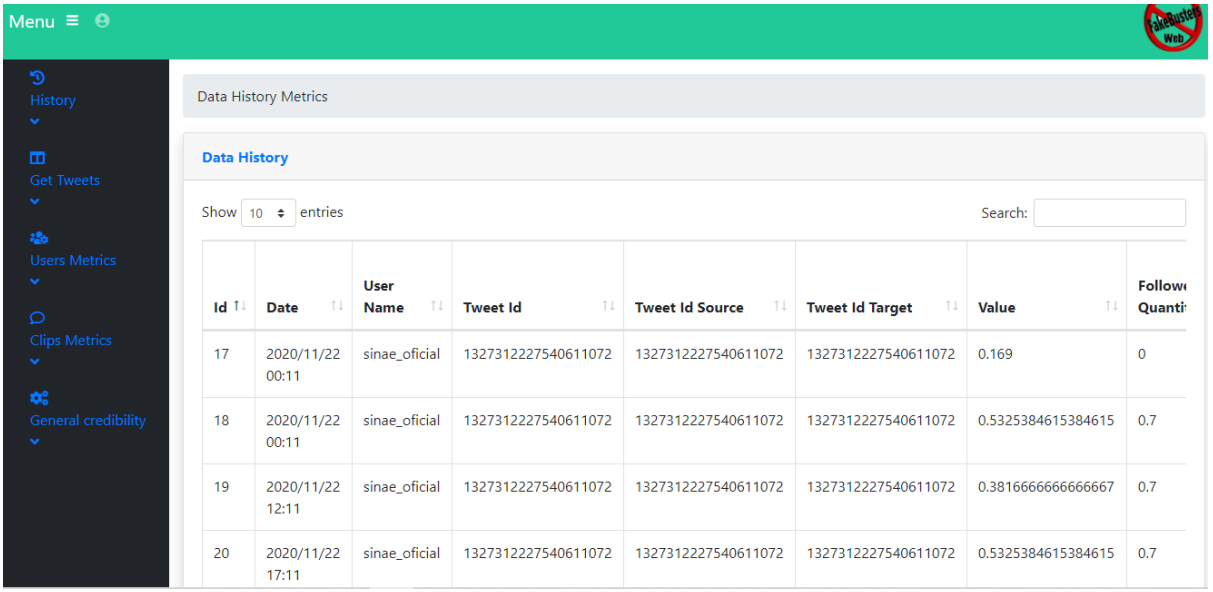

Figura 11 - Pantalla Data History Metrics

4.2.3- Obtener Información Tweet

En el menú *Get Tweets*, se encuentra la posibilidad de buscar información de los tweets que se encuentran en la base, filtrando por su hashtag, contenido del tweet o usuario que lo realizó. Figura 12.

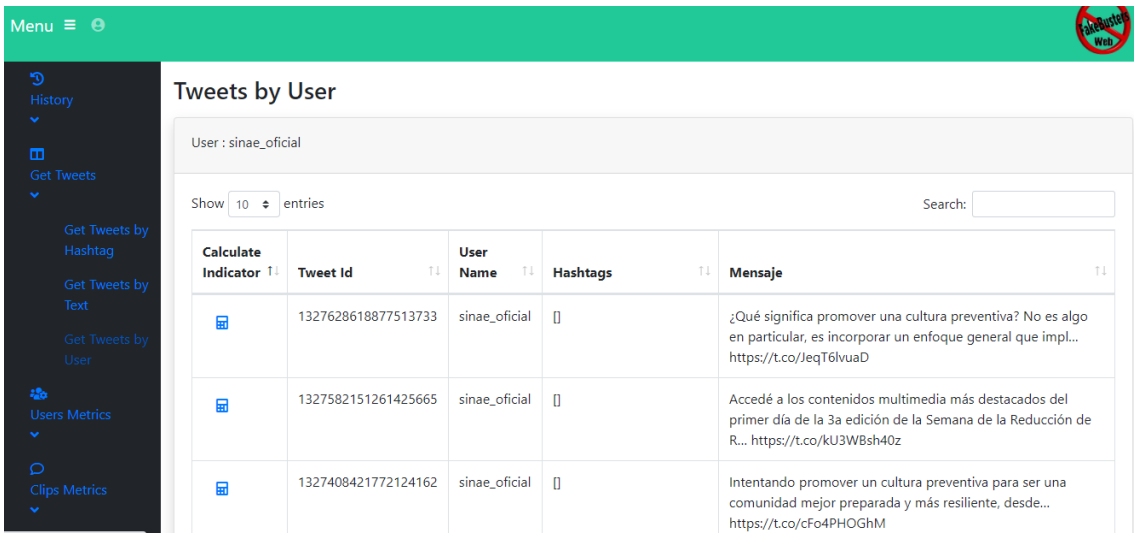

Figura 12 - Pantallas para obtener información tweets

Desde esta pantalla se puede seleccionar un tweet para el cual se quiera calcular su credibilidad general y por medio del botón calcular , dirigirse a la pantalla de *Calculate Credibility* para calcular la credibilidad del mismo.

4.2.4- Métrica de Usuarios y Clips

A través de los menús *User's Metrics* (métricas relativas a usuarios) y *Clip's Metrics* (métricas relativas a los tweets) se pueden calcular los valores de cada métrica (tanto de usuarios como de clips) de manera individual.

Las métricas que se pueden calcular a través del menú *User's Metrics* son: *Followers Quantity*, *Tweets & Retweets Pub*, *Favourite, Antiquity*, *Verified*, *Commented*, *Quote Count*, *Retweets Count*, *Favourite Count*, *Follower's Credibility*. A través del menú *Clip's Metrics* se encuentran las métricas *Similar Words*, *Same Location* y *Reputation Path*.

Para el caso de las métricas de usuario se toma como valor de entrada el nombre del usuario al cual se le quiere calcular dicha métrica. En el caso de las métricas de clips se requiere como parámetro de entrada el valor del id del tweet a calcular, a excepción de la métrica *Reputation Path* que requiere como parámetros de entrada el id del tweet origen y el del tweet destino.

Para todas las métricas se tiene como opción elegir desde cuál base de datos ejecutar las consultas (*Get from*), así como también elegir calcular los valores en ambas (opción *Both*).

Dado que se mantiene un historial de los resultados obtenidos para las distintas métricas por usuario o tweet (según corresponda), se tiene la opción de recalcular la métrica (*Recalculate Always*) u obtener el valor previamente calculado, en caso de existir (seleccionado por defecto).

A continuación, se muestra un ejemplo de la interfaz de entrada para la métrica *Followers Quantity* (Figura 13):

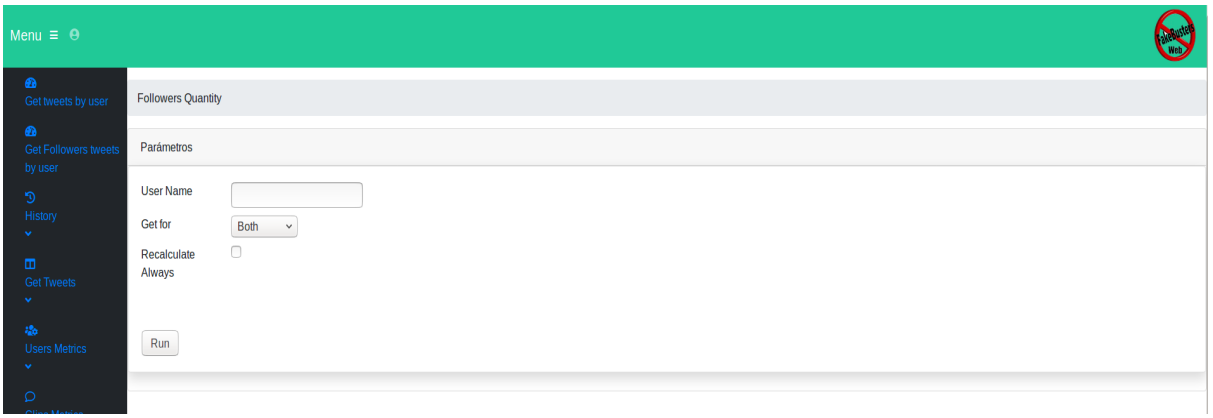

Figura 13 - Ejemplo interfaz de entrada cálculo métricas.

Como resultado de la métrica se muestra la dimensión, factor y métrica a la que corresponde el cálculo; el usuario para el cual se realizó dicho cálculo y los resultados obtenidos tanto para MongoDB como para Neo4j (en caso de que no se haya seleccionado alguno de ellos muestra como resultado None).

Junto con el resultado de la métrica se muestra un grafo con los nodos y relaciones comprendidas para su cálculo; para ello se utiliza la biblioteca Neovis.js (1.3.2). Cada uno de estos nodos, ya sea un nodo de Usuario o de tweet es coloreado dependiendo de la propiedad *community* calculada a partir del algoritmo *Label Propagation* de Neo4j (1.3.2), es decir nodos pertenecientes a una misma comunidad tienen igual color; a su vez, se hizo uso de la propiedad *size* para mostrar el valor obtenido mediante el algoritmo *PageRank* de Neo4j (1.3.2), cuánto mayor *PageRank*, mayor va a ser el tamaño del nodo. En el caso de las relaciones se utilizó la propiedad *thickness* la cual muestra a través de la variación del grosor de la flecha

(relación) el valor de cantidad de retweets (en el caso de la relación *posted_by*) y cantidad de seguidores (para la relación *followed_by*).

A continuación, se muestra la configuración utilizada en Neovis.js para la representación de los grafos:

```
labels: {
            "TwitterUser": {
                    "caption": "name",
                "size": "ppr",
                "community" : "community"

            },
             "Tweet": {
                "caption": "hashtags",
                "size": "ppr",
                "community": "community"
            }
},
relationships: {
          POSTED_BY": {
                "thickness": "retweet_count",
                "caption": true
                },
           "FOLLOWED_BY": {
                "thickness": "followers_count",
                "caption": false
                }
```

Con el fin de calcular los diferentes valores de *PageRank* de un usuario (guardados en la propiedad *ppr* del nodo) se corrió el algoritmo mediante la siguiente consulta:

```
MATCH (u:TwitterUser {name:$name})
        CALL algo.pageRank('TwitterUser', 'FOLLOWED_BY', {iterations:20, dampingFactor:0.85,
sourceNodes: [u], write: true, writeProperty:'ppr', weightProperty:'followers_count'})
YIELD nodes
MATCH p=(u)-[r:FOLLOWED_BY]->(n)
SET n.ppr = n.ppr*1000
RETURN *
```

Se utiliza como etiqueta los usuarios, nodo origen el usuario consultado, la relación *followed_by*, se itera el algoritmo 20 veces con un factor de amortiguación de 0.85 y tomando como peso la cantidad de seguidores (*followers_count*). Dado que el valor obtenido era poco apreciable al dibujar el grafo se multiplicó su valor por 1000. Por lo que la propiedad *ppr* obtenida a través del algoritmo representa la influencia o conectividad de un usuario con el resto de la red a través de sus seguidores (relación *followed_by*).

Para el cálculo de esta propiedad para los diferentes tweets se ejecutó el algoritmo *PageRank* mediante la consulta:

```
MATCH (u:TwitterUser {name:$name})
        CALL algo.pageRank('Tweet', 'POSTED_BY', {iterations:20, dampingFactor:0.85,
sourceNodes: [u], write: true, writeProperty:'ppr', weightProperty:'retweet_count'})
YIELD nodes
MATCH p=(n)-[r:POSTED_BY]->(u)
RETURN *
```

En este caso se utiliza como etiqueta los tweets, nodo origen el usuario consultado, la relación *posted_by*, se itera el algoritmo 20 veces con un factor de amortiguación de 0.85 y tomando como peso la cantidad de retweets

(*retweet_count*). La propiedad ppr obtenida luego de la ejecución del algoritmo muestra la influencia de los tweets de un usuario en la red a través la relación *posted_by.*

Para obtener los valores de la propiedad *community* se ejecutó el algoritmo *Label Propagation* tanto para los tweets como para sus usuarios.

La ejecución del algoritmo para los diferentes usuarios se realizó por medio de la siguiente consulta:

```
CALL algo.labelPropagation('TwitterUser', 'FOLLOWED_BY', {iterations: 10, writeProperty:
'community', write: true})

YIELD nodes
```

Como etiqueta se utilizan los usuarios, tomando como relación sus seguidores (*followed_by*) y se escribe el dato en la propiedad *community*, el algoritmo va a iterar como máximo 10 veces. En este sentido se representan las diferentes interconexiones de usuarios que se obtienen mediante los seguidores.
Por otro lado, para obtener las diferentes comunidades en relación con los tweets, el algoritmo fue ejecutado mediante la consulta que se muestra a continuación:

```
CALL algo.labelPropagation('Tweet', 'POSTED_BY', {iterations: 10, writeProperty: 'community',
write: true})

YIELD nodes
```

En este caso como etiqueta se utilizan a los tweets y como relación *posted_by,* el resto de los parámetros se mantienen iguales que en el caso anterior.

A continuación, se ilustra un ejemplo de cómo se muestran los resultados (Figura 14):

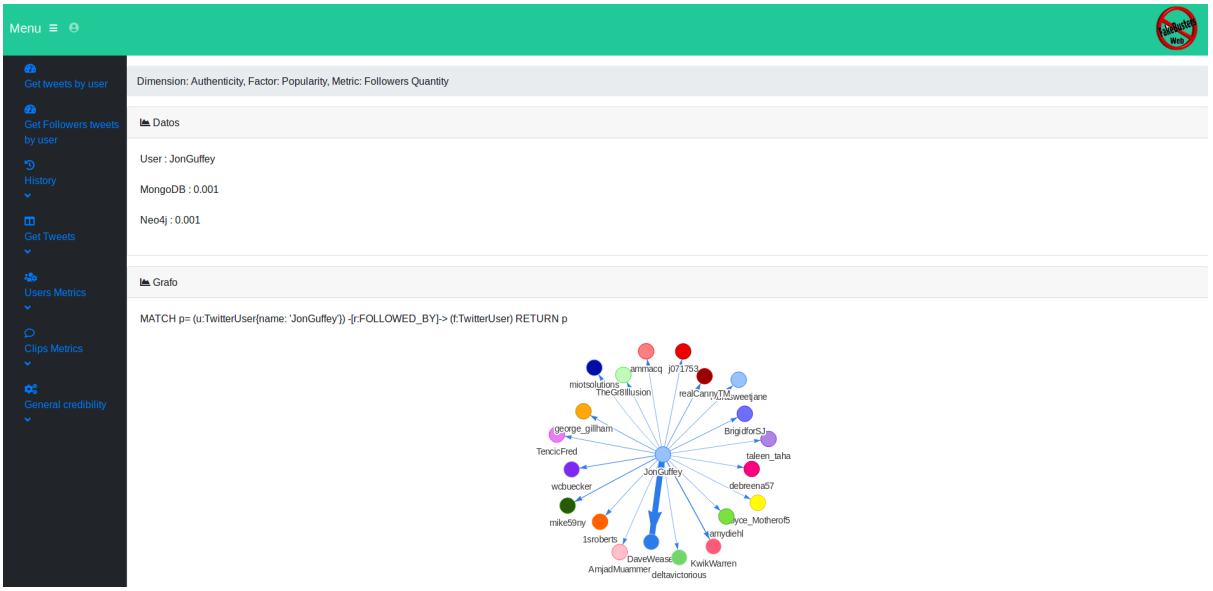

Figura 14 - Ejemplo interfaz de salida cálculo métricas.

4.2.5- Credibilidad General

Para poder tener una medida general de la credibilidad de un tweet, se necesita poder combinar diferentes métricas y asignarles un peso específico para cada una. Esto se puede hacer utilizando el menú *General Credibility*.

En el submenú *Weights* se definen los pesos y la base en el que se va a ejecutar cada métrica Figura 15.

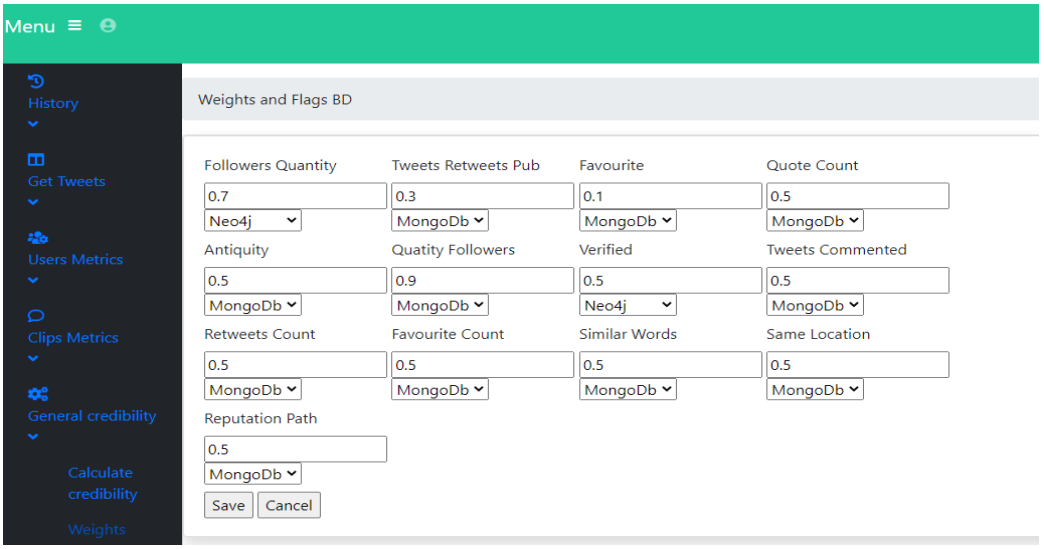

Figura 15. Pantalla de Weights

En el submenú *Calculate Credibility*, se eligen las métricas a calcular y se ingresan los valores de entrada para las mismas.

En esta pantalla existen tres pestañas, en las dos primeras se encuentran agrupadas por dimensión y factor, las métricas de usuario y clip respectivamente, en la tercera se encuentran los datos de entrada a ingresar dependiendo de las métricas seleccionadas. Figura 16

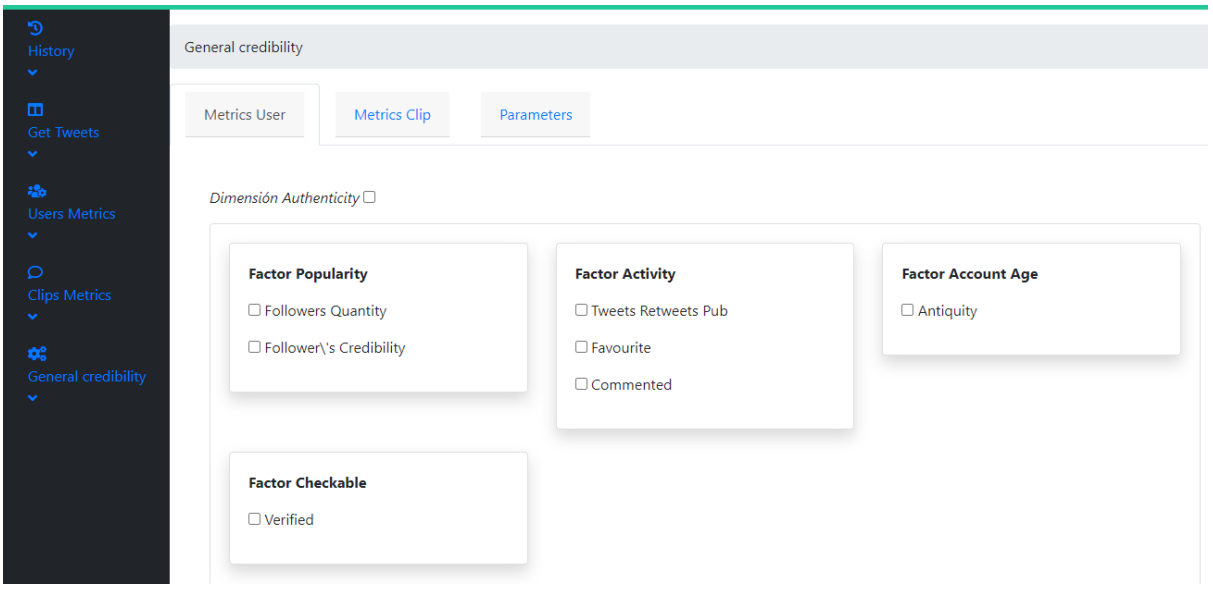

Figura 16. Pantalla Calculate Credibility

Luego de ejecutarla se visualizan las métricas, el resultado y los datos de entradas ingresados. Figura 17

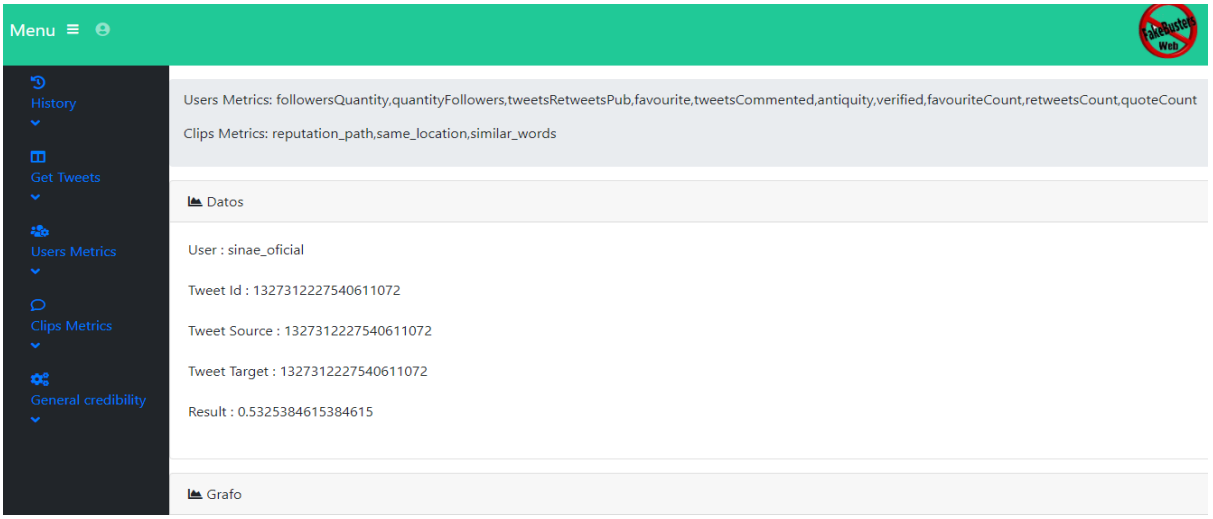

Figura 17. Pantalla Resultado Calculate Credibility