UNIVERSIDAD NACIONAL DEL ALTIPLANO

ESCUELA DE POSGRADO

MAESTRÍA EN INFORMÁTICA

TESIS

IMPLEMENTACIÓN DE UNA PLATAFORMA BIG DATA PARA EL ESTUDIO DE CASOS DE ANEMIA EN AMÉRICA LATINA

PRESENTADA POR:

ROSARIO BUSTAMANTE ROJAS

PARA OPTAR EL GRADO ACADÉMICO DE:

MAGÍSTER SCIENTIAE EN INFORMÁTICA

MENCIÓN EN INGENIERÍA DE SOFTWARE

PUNO, PERÚ

2019

UNIVERSIDAD NACIONAL DEL ALTIPLANO

ESCUELA DE POSGRADO

MAESTRÍA EN INFORMÁTICA

TESIS

IMPLEMENTACIÓN DE UNA PLATAFORMA BIG DATA PARA EL ESTUDIO DE CASOS DE ANEMIA EN AMÉRICA LATINA

PRESENTADA POR:

ROSARIO BUSTAMANTE ROJAS

PARA OPTAR EL GRADO ACADÉMICO DE:

MAGÍSTER SCIENTIAE EN INFORMÁTICA

MENCIÓN EN INGENIERÍA DE SOFTWARE

APROBADA POR EL SIGUIENTE JURADO:

PRESIDENTE

Dr. ALEJANDRO APAZA TARQUI

PRIMER MIEMBRO

D. Sc. PERCY HUATA PANCA

SEGUNDO MIEMBRO

M. Sc. NESTOR TIPULA QUISPE

ASESOR DE TESIS

Dr. EDGAR CARPIO VARGAS

Puno, 13 de febrero de 2019

ÁREA: Ingeniería de software.
TEMA: Big Data.

DEDICATORIA

A la memoria de mi abuelita Mercedes, mis papás Antonio Ronald y María Rosario por su apoyo incondicional.

A mis queridas hijas Paola Mercedes y María del Carmen.

A mi hermano Juan Antonio, a mi cuñada Katerine y mi sobrino Adrián Antonio y toda mi familia y amigos por el apoyo que siempre me brindaron día a día en el transcurso de cada año de mi Maestría Universitaria.

AGRADECIMIENTOS

- ✓ Doy gracias a Dios por permitirme tener tan excelente experiencia dentro de la Universidad.
- ✓ A la Universidad Nacional del Altiplano, Gracias por permitirme ser profesional maestro en lo que tanto me apasiona.
- ✓ Gracias a cada docente de la maestría en informática que hizo parte de mi proceso integral de formación. Así como a también a mis compañeros, colegas, estudiantes de la maestría en informática.
- ✓ Gracias por creer en mí y gracias a Dios por permitirme vivir y disfrutar cada día.
- ✓ Tomé este camino como un reto, aunque no es sencillo, gracias por su apoyo, su bondad, comprensión, aunque ha sido complicado lograr esta meta, les agradezco y hago presente mi afecto hacia ustedes mi inmensa familia.

ÍNDICE GENERAL

Pág.

CAPÍTULO III

MATERIALES Y MÉTODOS

CAPÍTULO IV

RESULTADOS Y DISCUSIÓN

ÍNDICE DE TABLAS

Pág.

ÍNDICE DE FIGURAS

Pág.

ÍNDICE DE ANEXOS

Pág.

RESUMEN

El acelerado avance de las tecnologías de la información en diversos entornos, así como en las plataformas móviles, ha generado que se puedan manejar grandes volúmenes de datos en tiempo real, por esta razón la presente tesis tuvo como objetivo general de implementar una plataforma de big data para el estudio de casos de anemia en América Latina, 2018. Se realizó la implementación de la plataforma de Big Data definiendo una metodología mediante un diseño que utiliza como principal herramienta Cloudera, una distribución de Linux, en la cual se realizó la configuración del agente Flume para iniciar con el streaming o transmisión de información disponible para la APIREST de Twitter, se utilizó HDFS(Hadoop Data File System, Sistema de Archivos para Hadoop) para el almacenamiento de información en Hadoop, para el mapeo y reducción de información se utilizó MapReduce, como bases de datos y procesamiento de la información, se cargó la información en Hive y MySql, de esa forma se demuestra que es posible utilizar tecnología de forma híbrida y para la visualización de gráficos en Excel. De esta manera se puede concluir que los términos encontrados son semejantes a los buscados, con 7,192,687 registros de tweets recolectados, se encontraron 23 veces de la palabra “anemia” que representa el 0.00032% de registros de tweets, esto respecto al 1% disponible de la data disponible en Twitter.

Palabras clave: Anemia, Big data, Hadoop, redes sociales y Twitter.

ABSTRACT

The accelerated advance of information technologies in various environments, as well as mobile platforms, have generated that can handle large volumes of data in real time, for this reason this thesis had as its general objective to implement a platform of big data for the study of cases of anemia in Latin America, 2018. The implementation of the Big Data platform was carried out by defining a methodology using a design that uses Cloudera as a main tool, a Linux distribution, in which Flume agent was configured to start streaming or transmitting information available to the user. APIREST Twitter, HDFS (Hadoop data File System, File System for Hadoop) was used for storage in Hadoop, for mapping and data reduction MapReduce was used as databases and information processing was charged the information in Hive and MySql, in this way it is demonstrated that it is possible to use technology in a hybrid way and for the visualization of graphics in Excel. In this way we can conclude that the terms found are similar to those sought, with 7,192,687 records tweets collected, they found 23 times of "anemia" word representing 0.00032% of records tweets, this about 1% available of the data available on Twitter.

Keywords: Anemia, Big data, Hadoop, social networks and Twitter.

INTRODUCCIÓN

La presente investigación se refiere al tema de big data aplicado para casos de Anemia en América Latina. La anemia es el déficit de consumo de hierro, elemento principal para la formación de la hemoglobina, mediante las redes sociales como Twitter es posible identificar a un grupo de personas que les haya interesado hablar de anemia en un momento y quedando registrado en su cuenta. Una de las causas por las que se genera la anemia es por la necesidad económica y desórdenes alimenticios, los cuales son cuantificables.

La investigación de esta problemática se realizó por el interés de saber la cantidad de personas que manifiestan interés personal o familiar al respecto de la anemia en la red social Twitter, así como utilizar tecnología referente a Big data y herramientas relacionadas al manejo de grandes volúmenes de información como hadoop, Cloudera, Flume, MapReduce y Hive.

En el Capítulo I se realiza la revisión de literatura, definición de la tecnología Big Data, y explicando los requisitos básicos para que un conjunto de información se pueda considerar Big Data.

En el Capítulo II se realiza el planteamiento del problema, en el que se plantea en la formulación del problema con la siguiente interrogante: ¿Cuáles son los resultados de la implementación de una plataforma big data para el estudio de casos de anemia en América Latina? Se tiene el siguiente objetivo general que es: Implementar una plataforma de big data para realizar el estudio de casos de anemia en América Latina, 2018 Los objetivos específicos son: 1. Definir una metodología para el desarrollo de una plataforma de big data., 2. Implementar una plataforma de big data para casos de anemia en América latina, 3. Evaluar los resultados de la plataforma de big data para casos de anemia en América Latina. En el diseño se define la metodología para la plataforma, luego se procedió a la implementación. Las pruebas se realizaron con información real para obtener resultados reales.

En el Capítulo III, se define el lugar de estudio, población y muestra, así como los métodos para la investigación.

En el Capítulo IV, la conclusión general es que la plataforma de big data ha permitido obtener resultados del análisis de los datos obtenidos para casos de anemia en América Latina.

CAPÍTULO I

REVISIÓN DE LITERATURA

1.1 Marco teórico

Según Menasalvas (2015) dice:

Un estudio realizado por la consultora International Data Corporation (IDC), de 2005 a 2020 se espera que el tamaño del universo digital se multiplique por 300, creciendo de 130 exabytes (un exabyte es un millón de gigabytes) a 40 mil, duplicándose anualmente la cantidad de datos digitales. Esto significa que se almacenarán 5.200 gigabytes por ser humano. Un nivel de complejidad alcanzado tanto en los datos como en su análisis, que impide tratar con el software tradicional. Y es ahí donde surge el Big Data.

1.2 Definición de big data

Según Rouse (2017) dice, "Big data (en español, grandes datos o grandes volúmenes de datos) es un término evolutivo que describe cualquier cantidad voluminosa de datos estructurados, semiestructurados y no estructurados que tienen el potencial de ser extraídos para obtener información".

De manera similar, Dijcks (2013) afirma:

Big Data son los datos provenientes de las diferentes fuentes de datos como los datos tradicionales de las empresas que incluye la información de los clientes de los diferentes sistemas ya sea transacciones en la web, CRM, ERP; también, datos generados por máquinas como sensores de datos, medidores inteligentes e industriales, entre otras y; por último y no menos importante, datos sociales como el Twitter, Facebook, Linkedin, entre otras.

Es así que NIST (2018) indica lo siguiente:

El principal beneficio del análisis de Big Data es la capacidad de procesar grandes cantidades y varios tipos de información. La necesidad de un mayor rendimiento o eficiencia ocurre de manera continua. Sin embargo, Big Data representa un cambio fundamental a la escalabilidad paralela en la arquitectura necesaria para manejar de forma eficiente los conjuntos de datos actuales.

Características de big data

Existen cuatro características:

- **Volumen**: Para Joyanes (2016) "La cantidad de datos en el año 2000, era aproximadamente de 800,000 petabytes de datos guardados a nivel global. Este número se espera incrementar para el año 2020 en 35 zetabytes, considerando que Twitter genera 9 terabytes y Facebook 10 terabytes cada día".

- **Velocidad**: Según Dijcks (2013) indica que los flujos de datos de las redes sociales, aunque no son tan masivos como los datos generados por una máquina, Produce una gran afluencia de opiniones y relaciones valiosas para la relación con el cliente, incluso a 140 caracteres por tweet, la alta velocidad (o frecuencia) de los datos de Twitter aseguran grandes volúmenes (más de 8 TB por día.

– **Variedad**: Según Dijcks (2013) indica que los formatos de datos tradicionales tienden a estar relativamente bien definidos por un esquema de datos y cambian despacio En contraste, los formatos de datos no tradicionales exhiben una tasa vertiginosa de cambio. A medida que se agregan nuevos servicios, se implementan nuevos sensores o nuevas campañas de marketing. ejecutado, se necesitan nuevos tipos de datos para capturar la información resultante.

– **Valor**: Según Dijcks (2013) indica que el valor económico de los diferentes datos varía significativamente. Típicamente hay buena información oculta entre un cuerpo más grande de datos no tradicionales; el reto es identificar lo que es valioso y luego transformar y extraer esos datos para su análisis.

Fases de la aplicación big data

Las empresas deben evolucionar sus infraestructuras de TI para gestionar estos nuevos retos con alta velocidad, gran volumen y fuentes de alta variedad de datos. Se desagrega en 3 fases según Dijcks (2013):

- Adquirir: Esta infraestructura debe ser capaz de entregar baja latencia predecible, tanto para la captura de datos como la ejecución de consultas simples. Las bases de datos NoSQL son utilizadas con frecuencia para adquirir y almacenar grandes volúmenes de datos de los medios sociales ya que trabaja con estructuras de datos dinámicas y son altamente escalables.

- Organizar: en esta fase, los datos son denominados integración de datos, ya que los datos son organizados en la ubicación de almacenamiento original siendo capaz de manipular y procesar gran cantidad de datos no estructurados. Hadoop es una de las nuevas tecnologías que permite la manipulación y procesamiento de datos, manteniendo la ubicación de almacenamiento de datos original.

- Analizar: en esta fase, el análisis resulta en un entorno distribuido, accediendo a los datos de la ubicación original o de forma transparente de un Data warehouse. La infraestructura debe ser capaz de soportar análisis más profundos, como análisis estadísticos o minería de datos, teniendo como resultado tiempo de respuesta eficiente y automatización de decisiones.

1.3 Arquitectura

La arquitectura del Big Data es equivalente a una arquitectura de macrodatos, está diseñada para controlar la ingesta, el procesamiento y el análisis de datos que son demasiado grandes o complejos para los sistemas de bases de datos tradicionales.

Componentes de una arquitectura de macrodatos

En la Figura 1 se muestra los componentes lógicos que contiene una arquitectura de macrodatos. Es posible que las soluciones individuales no contengan todos los elementos de este diagrama.

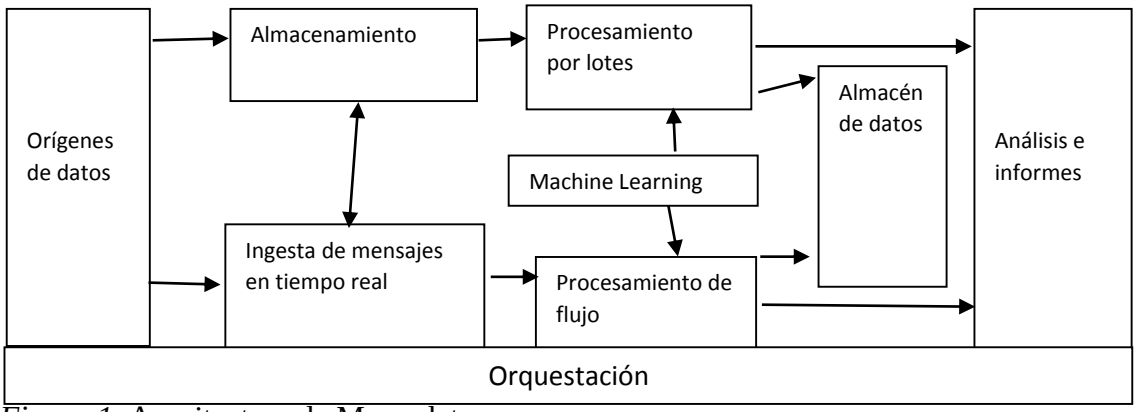

Figura 1. Arquitectura de Macrodatos

Fuente: (Tejada, Zoiner, 2017)

La mayoría de las arquitecturas de macrodatos incluyen algunos de los componentes siguientes (o todos ellos). Según detalla Tejada, Zoiner (2017), resumimos cada uno de ellos a continuación:

- **Orígenes de datos.** Se refiere a almacenes de datos de aplicación, como bases de datos relacionales, archivos estáticos generados por aplicaciones, archivos de registro de servidor web, orígenes de datos en tiempo real, como dispositivos de IoT

- **Almacenamiento de datos**. Los datos de las operaciones de procesamiento por lotes se almacenan normalmente en un almacén de archivos distribuido que puede contener importantes cantidades de archivos grandes en diferentes formatos.

- **Procesamiento por lotes**. Como los conjuntos de datos son tan grandes, a menudo una solución de macrodatos debe procesar los archivos de datos mediante trabajos por lotes de ejecución prolongada para filtrar, agregar o preparar de cualquier otra forma los datos para su análisis. Las opciones incluyen la ejecución de trabajos con el uso de Hive, o trabajos personalizados de Map/Reduce en un clúster de HDInsight Hadoop.

- **Ingesta de mensajes en tiempo real**. Si la solución incluye orígenes en tiempo real, la arquitectura debe incluir una manera de capturar y almacenar los mensajes en tiempo real para el procesamiento de flujos. Esta parte de una arquitectura de streaming a menudo se conoce como almacenamiento en búfer de flujos.

- **Procesamiento de flujos**. Una vez capturados los mensajes en tiempo real, la solución debe procesarlos filtrando, agregando o bien preparando los datos para

su análisis. Los datos de secuencias procesados se escriben entonces en un receptor de salida. que proporciona un servicio de procesamiento de secuencias administrado basado en consultas SQL de ejecución permanente que operan en secuencias sin enlazar.

- **Almacén de datos analíticos**. Es posible que los datos se presenten a través de una tecnología NoSQL de baja latencia como HBase, o una base de datos de Hive interactiva que proporciona una abstracción de metadatos sobre los archivos de datos en el almacén de datos distribuidos.

- **Análisis e informes**. los análisis y la creación de informes también pueden adoptar la forma de exploración interactiva de datos por parte de científicos o analistas de datos.

- **Orquestación**. La mayoría de las soluciones de macrodatos constan de operaciones de procesamiento de datos repetidas, encapsuladas en flujos de trabajo, que transforman los datos de origen, mueven datos entre varios orígenes y receptores, cargan los datos procesados en un almacén de datos analítico o envían los resultados directamente a un informe o panel.

1.4 Tecnologías sobre las que se apoya el big data

Las tecnologías para realizar el tratamiento de los datos Big Data son las bases de datos Nosql, hadoop y flume. (Rodrìguez Garcìa, 2018).

1.4.1 Bases de datos

Para Silberschatz (2001) tenemos la siguiente definición:

Un sistema de bases de datos es una colección de archivos interrelacionados y un conjunto de programas que permitan a los usuarios acceder y modificar estos archivos. Uno de los propósitos principales de un sistema de bases de datos es proporcionar a los usuarios una visión abstracta de los datos. Es decir, el sistema esconde ciertos detalles de cómo se almacenan y mantienen los datos.

1.4.2 NoSQL

Según Systems (2017) define de la siguiente manera:

Las bases de datos nosql adoptan un enfoque diferente para resolver los problemas de Big Data. Se han tenido en cuenta muchas consideraciones inherentes a la arquitectura y el diseño, como el diseño sin esquema, el compromiso con las propiedades Atomicidad, Consistencia, Integridad y Durabilidad (ACID), principalmente el procesamiento basado en RAM para un gran conjunto de datos, el almacenamiento basado en valores-clave / documento.cuenta en la mayoría de las bases de datos de nosql.

1.4.3 Hadoop

"Hadoop es un framework, que permite el procesamiento de grandes volúmenes de datos a través de clusters, usando un modo simple de programación". (IDS, 2018).

Para Hernández-Leal, Duque-Méndez, & Moreno-Cadavid (2017):

Hadoop cuenta con dos componentes principales, el HDFS, sistema de archivos distribuidos que permite distribuir los ficheros en distintas máquinas y MapReduce, framework que permite al desarrollador aislarse de la programación paralela, permite ejecutar programas escritos en lenguajes de programación conocidos (p.e Java) en el clúster de Haddop.

Para complementar IDC (2018) indica:

En lugar de utilizar un equipo grande para procesar y almacenar datos, se utilice una variedad de herramientas para satisfacer las necesidades de cargas de trabajo en el análisis de datos gracias a un sistema de archivos distribuidos que engloba distintos productos. Es decir: soporta distintas aplicaciones distribuidas bajo una licencia libre que permite trabajar con miles de nodos y petabytes de datos.

Entre sus ventajas, además de permitir pasar de pocos nodos a miles de nodos de forma ágil están Holmes (2012), indica lo siguiente:

- Capacidad de ejecutar procesos en paralelo en todo momento

- Permite realizar consultas

- Mejor disponibilidad y recuperación ante los desastres

- Tecnología escalable

- Almacenamiento de bajo coste

- Flexibilidad

- Velocidad

Hadoop está desarrollado como una arquitectura distribuida maestro – esclavo, como se muestra en la Figura 2, que consiste en Hadoop Distributed File System (HDFS), para almacenamiento y MapReduce para capacidades computacionales. Los rasgos intrínsecos de Hadoop son la partición de datos y la computación paralela de grandes cantidades de datos. Su almacenamiento y capacidades computacionales se escalan con el número de hosts de la rama de Hadoop, y pueden llegar a volúmenes de *petabytes* con cientos de *hosts*.

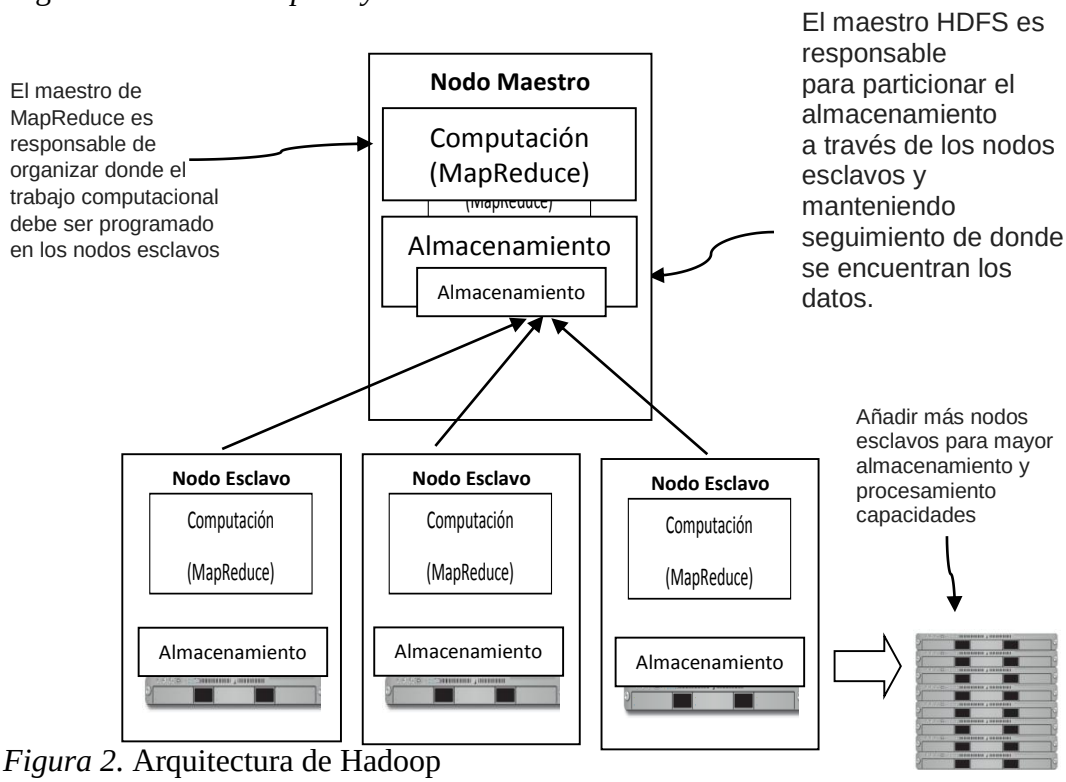

Figura 2. Arquitectura de Hadoop

Fuente: (Holmes, 2012)

A continuación, las tecnologías HDFS y MapReduce, como parte del funcionamiento de Hadoop.

1.4.3.1 Hadoop Distributed File System (HDFS)

Según Holmes (2012) afirma lo siguiente:

HDFS es el componente de almacenamiento de Hadoop. Es un sistema de ficheros distribuido, modelado posterior a Google File System (GFS) de Google. HDFS está optimizado para tener un alto rendimiento y funciona mejor al leer y escribir archivos de gran tamaño (gigabytes y más grandes). Para apoyar este rendimiento HDFS aprovecha inusualmente grande (para un sistema de archivos) tamaños de bloque y optimizaciones para reducir la red de entrada / salida (I / O).

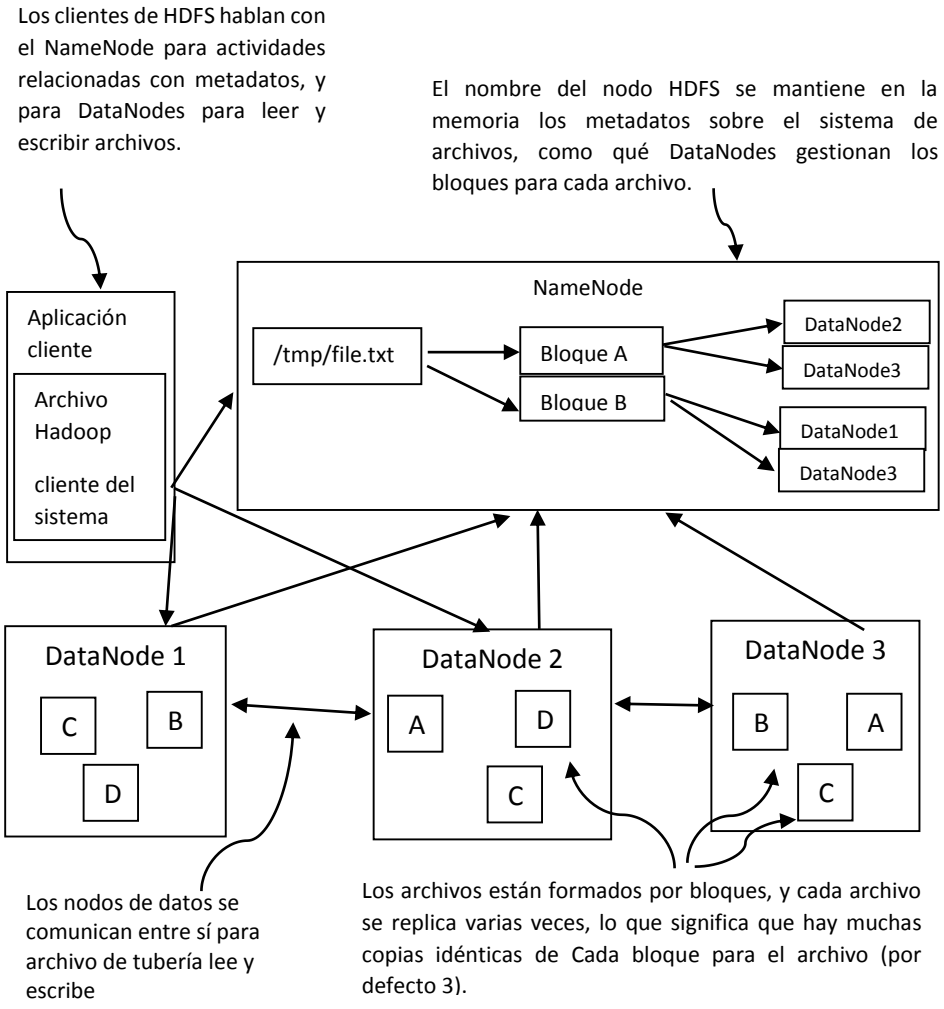

Figura 3. Arquitectura HDFS
Fuente: (Alex, 2012)

Así también Holmes (2012) indica:

La escalabilidad y la disponibilidad son también rasgos clave de HDFS, que se logra, en parte debido a la replicación de datos y tolerancia a fallos.

HDFS replica archivos para un número configurado de veces, es tolerante con el fracaso tanto de software y hardware, y automáticamente re-replica bloques de datos en los nodos que han fallado.

La Figura 3 muestra una representación lógica de los componentes en HDFS: la NameNode y la DataNode. También muestra una aplicación que está utilizando el sistema de archivos de la biblioteca de Hadoop para acceder a HDFS.

MapReduce

Los trabajos de Hadoop MapReduce se dividen en un conjunto de tareas de mapa y reducen las tareas que se ejecutan de forma distribuida en un conjunto de computadoras. Cada tarea trabaja en el pequeño subconjunto de los datos que se le han asignado para que la carga se distribuya en el clúster, como se muestra en la Figura 4.

Detalla Donald & Adam (2013)

Las tareas del mapa generalmente cargan, analizan, transforman y filtran datos. Cada tarea de reducción es responsable de manejar un subconjunto de la salida de la tarea del mapa. Los datos intermedios se copian de las tareas del asignador mediante las tareas del reductor para agrupar y agregar los datos.

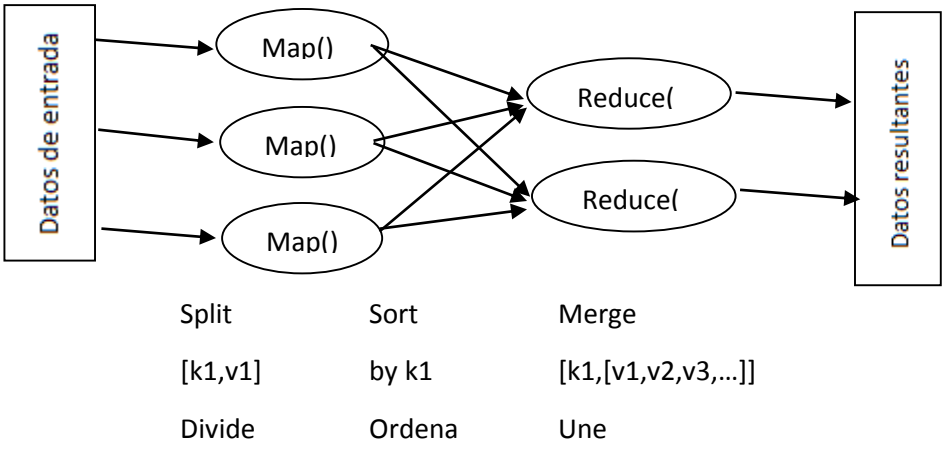

Figura 4. Funcionamiento Mapreduce

Fuente: (Niño, 2015)

1.4.3.2 Limitaciones de Hadoop

Las dos principales limitaciones que rodean Hadoop son la disponibilidad y la seguridad.

En HDFS se debe destacar la falta de disponibilidad, ya que es ineficiente manejando ficheros pequeños, y tiene una falta de compresión transparente. HDFS no está diseñado para trabajar bien con lecturas aleatorias sobre ficheros pequeños debido a su optimización para un gran rendimiento. No obstante, esto se soluciona con la nueva versión de HDFS de *HighAvailability*. (Holmes, 2012)

MapReduce es una arquitectura basada en procesos batch, que implica que no se presta a sí mismo para usar casos que necesitan acceso a datos en tiempo real. Las tareas que requieren sincronización global o compartir datos mutables no encajan bien con MapReduce, debido a que se trata de una arquitectura que no comparte, lo que produce algunos retos en ciertos algoritmos que emplea. Como alternativa mejorada a MapReduce, Apache ha desarrollado el proyecto YARN. (Holmes, 2012)

1.4.3.3 Hadoop 3.0

Hadoop con la versión 3.x. tiene una arquitectura mejorada con YARN y los bloques de construcción parecen más flexibles.

Según Intellipaat (2018) algunas características adicionales en Hadoop 3.0.0 a continuación:

YARN Timeline Service Versión 2: Este servicio está equipado con la capacidad de mejorar la escalabilidad, confiabilidad y facilidad de uso mediante flujos y agregación, contiene métricas, información específica de la aplicación, eventos de contenedores, etc.

Compatible con Java 8: Hadoop 3.0.0 funcionan con Java 8. (Intellipaat, 2018)

Tolerancia a fallos mejorada con Quorum Journal Manager: la tolerancia a fallas del clúster de big data ha sido mejorada con la ayuda de

Quoram Journal Manager, que se compone de un mínimo de tres nodos que pueden recuperar el sistema, incluso si falla un nodo. La razón detrás de esta tolerancia mejorada a fallos es que ejecuta varios NameNodes en espera, a diferencia del anterior, lo que a su vez aumenta la eficiencia de HDFS.

Intra-DataNode Balancing: Corrige los errores que ocurren mientras se agregan o eliminan más espacios de almacenamiento. Por lo general, mientras se realiza una operación de escritura en un disco, se llenará de manera uniforme, sin embargo, a veces se producen sesgos en el DataNode al agregar o eliminar lo que no fue manejado por el equilibrador HDFS. Por lo tanto, Intra-DataNode equilibra este error.

1.4.4 Flume

Según Point (2017) Apache Flume es una herramienta, servicio y mecanismo de ingesta de datos para recopilar datos agregados y transporte de grandes cantidades de datos de transmisión, tales como archivos de registro, eventos (etc.) desde varias fuentes a un almacén de datos centralizado. Flume es una herramienta altamente confiable, distribuida y configurable. Está diseñado principalmente para copiar datos de transmisión (datos de registro) de varios servidores web a HDFS como se muestra en la Figura 5.

También Big data (2017) indica que la información procedente de las redes sociales, como Twitter, se va a recopilar con Flume, que a través de una API REST permite conectarse directamente con el sistema a emplear en el almacenamiento (HDFS), y volcarlo de una forma muy sencilla.

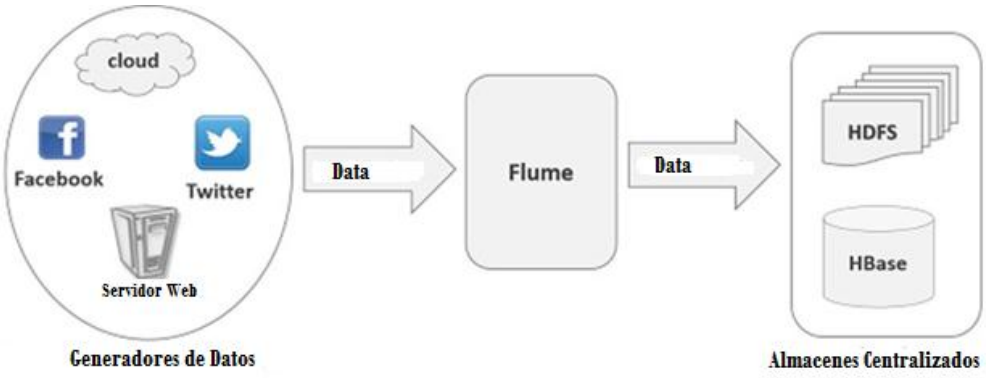

Figura 5. Funcionamiento de Flume

Fuente: (Point, 2017)

Arquitectura de Flume

Resumimos a continuación los conceptos de Flume según Flume (2019).

- Evento: una carga de bytes, con encabezados opcionales, que representan la unidad de datos que Flume para transportar desde el origen hasta el destino.
- Flujo de datos: es el movimiento de eventos desde el origen al destino.
- Cliente: es la implementación de una interfaz que recoge los eventos y se los entrega a un agente de Flume. Suele operar en el espacio de proceso de la aplicación de la que se están consumiendo los datos.

Se muestra en la Figura 6 la arquitectura de Flume, y a continuación se describe sus elementos:

- Agente: es un proceso independiente que aloja otros componentes como *sources*, *sinks* o *channels*. Se encarga de recibir, guardar y enviar eventos.
- Source: es la implementación de una interfaz que consume eventos que le son enviados con un mecanismo específico. Por ejemplo, la interfaz de Avro es una implementación que permite recibir eventos Avro desde clientes u otros agentes y ponerlos en un canal.
- Channel: es un almacén transitorio de eventos, estos eventos son entregados al canal a través de las fuentes que operan en el agente.

- Sink: es la implementación de una interfaz que permite eliminar eventos de un canal y transmitirlos al siguiente agente en el flow o a su destino final. Estos últimos son conocidos como terminal sink, como HDFS.

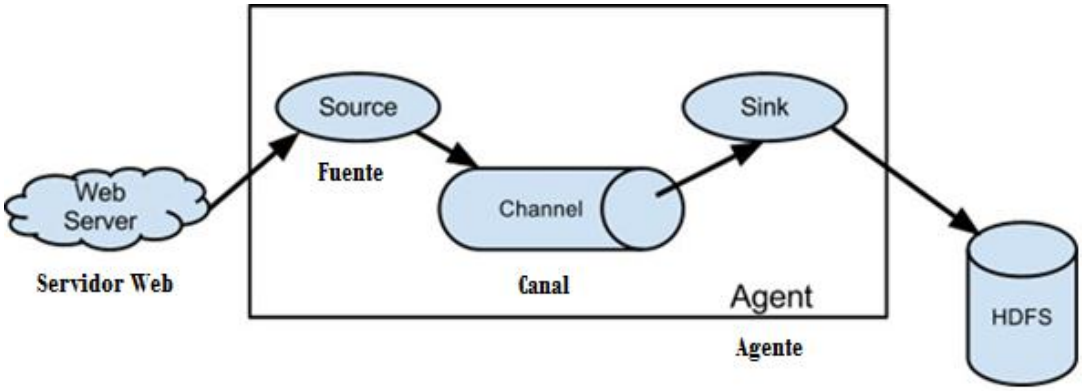

Figura 6. Arquitectura de Flume

Fuente: (Flume, 2019)

Flujo de datos

Según Big data (2017), *source* consume eventos dirigidos hacia él desde una fuente externa, por ejemplo, un servidor de aplicaciones. La fuente externa envía eventos a Flume en un formato que es reconocible para el *source* al que va dirigido. Cuando un *source* recibe un evento, lo almacena en uno o varios canales. El canal es un almacenamiento transitivo que guarda los eventos hasta que son consumidos por un *sink*. Por ejemplo, el canal de ficheros está respaldado por el sistema de ficheros local. S*ink* elimina el evento del canal y lo pone en un repositorio externo como HDFS o lo manda al siguiente *source* del siguiente agente en el flujo de datos.

La anemia

Según Figueroa Chire (2014) indica que la anemia es un síndrome agudo o crónico, caracterizado por una disminución en la capacidad de transporte de oxígeno por la sangre, en asociación con una reducción en el recuento eritrocitario total y/ o disminución en la concentración de hemoglobina (Hb) circulante, en relación con valores límites definidos como normales para la edad, raza, género, cambios fisiológicos (gestación, tabaquismo) y condiciones medioambientales (altitud).

Causas

Según Figueroa Chire (2014) la anemia tiene tres causas principales:

- Pérdida de sangre
- Falta de producción de glóbulos rojos
- Aumento en la velocidad de destrucción de los glóbulos rojos

Efectos

Según Figueroa Chire (2014):

- Palidez anormal o pérdida de color en la piel
- Aceleración de la frecuencia cardíaca (taquicardia)
- Dificultad respiratoria (disnea)
- Falta de energía, o cansancio injustificado (fatiga)
- Mareos o vértigo, especialmente cuando se está de pie
- Dolores de cabeza - Irritabilidad
- Ciclos menstruales irregulares
- Ausencia o retraso de la menstruación (amenorrea)
- Llagas o inflamación en la lengua (glositis)
- Ictericia o color amarillento de la piel, los ojos y la boca
- Aumento del tamaño del bazo o del hígado. (esplenomegalia, hepatomegalia)
- Retraso o retardo del crecimiento y el desarrollo
- Cicatrización lenta de heridas y tejidos (18).

Según (CMP, Mayo, 2018) indica que la anemia representa el más extendido problema de salud y nutrición publica en el mundo. Se estima que más de 2 000 millones de personas (30 % de la población mundial) registran algún grado de anemia. Si bien es cierto que los niveles de anemia son mayores en los países, regiones y grupos poblacionales con mayor nivel de pobreza, afecta a casi todos los países y todos los grupos poblacionales, incluidos los no pobres. La principal causa de anemia es el déficit en el consumo de hierro, elemento principal para la

formación de hemoglobina, que puede ser exacerbado por las enfermedades infecciosas.

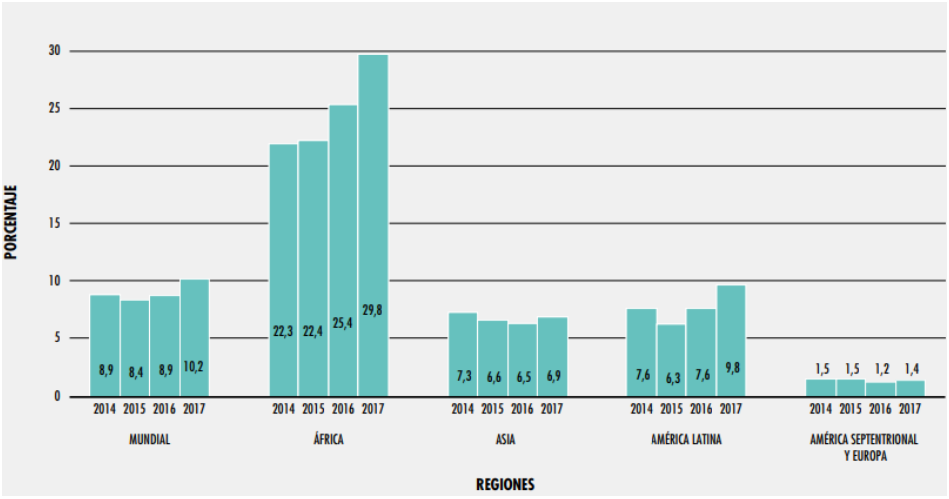

Figura 7. La Inseguridad Alimentaria Grave en 2017 es más alta que en 2014 en todas las regiones excepto América Septentrional y Europa, con aumentos notables en África y América Latina

Fuente: (F.A.O., 2018)

Según las estimaciones más recientes de la FAO en 2017 (FAO, 2018):

Aproximadamente 10% de la población mundial estuvo expuesta a una inseguridad alimentaria grave, lo que corresponde a alrededor de 770 millones de personas. A nivel regional, los valores oscilan entre el 1,4% en América Septentrional y Europa y casi el 30% en África. Al igual que en el caso de la prevalencia de la subalimentación, la inseguridad alimentaria grave ha ido en aumento a nivel mundial, impulsado por las tendencias observadas en África y América Latina tal como se muestra en la Figura 7.

1.5 Antecedentes

1.5.1 Antecedentes internacionales

Ballestar de las Heras (2018) concluye sobre el uso de metodologías vinculadas a la conocida como ciencia de los datos para el análisis masivo de información (Big-Data), ha permitido contrastar la relevancia de las redes sociales en el contexto del e-commerce. Así mismo, se ha contrastado la conveniencia del uso de estrategias de marketing tradicionales en combinación con otras más innovadoras como el cash back y el word of-mouth. En ambos casos el principal objetivo de las estrategias de

marketing en e-commerce consiste en incrementar la captación y fidelización de clientes, para de este modo, generar beneficio económico tanto para el consumidor como para el e-commerce. Por lo tanto, esta información constituye una herramienta valiosa que abre la posibilidad de un diseño de estrategias de marketing personalizadas basándose en el perfil del consumidor, lo que permitiría optimizar y maximizar el retorno de la inversión(ROI).

Figueres Cañadas (2017) concluye en este trabajo de investigación se ha pretendido contribuir a ampliar nuestra comprensión acerca del fenómeno Big Data y, sobre todo, de sus potenciales efectos sobre el desarrollo económico, pues se considera que cualquier hallazgo en este sentido podría resultar de gran utilidad para ayudarnos a identificar las estrategias más efectivas para combatir la pobreza y la injusticia social. Luego de proponer una nueva metodología de análisis, con el fin de elaborar una base teórica sobre la que fundamentar el análisis del fenómeno Big Data y de sus efectos sobre el desarrollo económico se aplica a la actual era del Big Data (Análisis Histórico), tras lo cual identifica una serie de procesos macroeconómicos de fondo que considera que sería importante tener en cuenta a la hora de elaborar planes empresariales. Finalmente, utiliza la base teórica histórica completa elaborada (Análisis Generativo) para valorar diferentes planes de acción contra la pobreza (Análisis Discriminativo), sin pretender con ello alcanzar ninguna respuesta concluyente cuyo éxito esté a priori garantizado.

González-Benito (2016) concluye que a lo largo de todo el trabajo se ha diferenciado entre dos tipos de usuarios con formación en nuevas tecnologías. De un lado están aquellos que dominan ciertas herramientas o redes sociales desde un punto de vista funcionalmente avanzado. La mayoría de periodistas recién titulados forman parte de este grupo. Del otro están usuarios que bien por su formación académica, bien por su inquietud e interés en la materia han ahondado en la parte más técnica de la informática, siendo capaces de desarrollar herramientas propias y de comprender la realidad digital que les rodea de un modo más preciso. Solo el segundo grupo está capacitado para exprimir la totalidad de oportunidades que Internet brinda a la sociedad red.

Guerrero López & Rodríguez Pinilla (2013) concluye en la estructura de un ambiente Big Data que ayuda a mejorar la manipulación de los datos, optimizando

la gestión de la información respecto a tiempo y costo, logrando obtener mejores resultados en las estadísticas para una buena toma de decisiones. La creación de un ambiente Big Data se debe realizar dentro de un clúster, el cual permita integrar todas las aplicaciones que se van a utilizar, como en este caso Hadoop, en el cual se almacena la información y las aplicaciones corren dentro del mismo nodo, evitando conflictos durante la ejecución. Es importante resaltar que existen muchas maneras para transformar el mismo modelo relacional al modelo basado en columnas, ya que se pueden tomar distintos caminos para la unión de los datos, esto depende de la información que se desee encontrar o saber. Para obtener una adecuada transformación se deben tener en cuenta las llaves primarias, las cuales se convertirán en las row key, que permitirá integrar toda la información dentro de una misma columna, mejorando la manipulación que se darán a los datos.

Hernández-Leal E. J. (2016) concluye, en la investigación se buscó hacer frente a algunas limitaciones encontradas respecto a la administración de grandes volúmenes de datos ambientales (hidrometeorológicos). Se logró como resultado la definición de un modelo por capas para la gestión y tratamiento de datos en el dominio ambiental. Cada una de las capas del modelo propuesto tiene elementos modulares que se pueden incluir, cambiar o ampliar. La aplicación del modelo por capas permite presentar información relevante para el análisis de datos de este campo de dominio acordes a la región y que podrán ser utilizados en la toma de decisiones o para el entendimiento de los fenómenos que están detrás de este tipo de datos.

Hernández-Leal, Duque-Méndez, & Moreno-Cadavid (2017) concluye en que Big Data no trata solo de grandes volúmenes de datos, sino que incluye otras dimensiones significativas en el tratamiento de datos, como son la variedad, velocidad y veracidad. No obstante, una implementación de Big Data requiere altos costos en expertos, mayor tiempo de adaptación tecnológica, dificultad para implementar nuevos análisis y percepción limitada. Big Data no busca sustituir a los sistemas tradicionales, sino construir una nueva tendencia donde se construyan arquitecturas de sistemas que permitan manejar todas las peticiones. Y ya ha logrado incentivar en la comunidad académica y comercial el desarrollo de tecnologías de apoyo que toman los paradigmas base y los emplean en la construcción de soluciones particularizadas a problemas de entornos de

investigación y producción reales.

Manso (2015) concluye en que mediante soluciones de Big Data (BDL), que son básicamente repositorios para manejar grandes volúmenes de datos que, a diferencia de un EDW tradicional, pueden almacenar y manejar una cantidad masiva de datos estructurados, semi-estructurados y no estructurados en su forma cruda en sistemas de almacenamiento de bajo costo por ser un commodity. Para alcanzar el éxito en la incorporación de una estrategia de la explotación de los datos mediante BDL en las empresas de telecomunicaciones, se deben realizar un cambio de diseño organizacional y cultural que permita integrar en forma centralizada mediante un Chief Data Officer (CDO) las áreas de ingeniería, IT y servicio al cliente que actualmente trabajan en forma de silos. Esto se debe a que una plataforma de estas características permite agregar valor al negocio mediante la integración de información complementaria.

Moclan Soria (2016) concluye en que propuso un prototipo de infraestructura diseñada en la nube para los distintos tipos de usuarios, soportada por todos los nuevos desarrollos tecnológicos que ha permitido el big data. A partir de la infraestructura en la nube propuesta se ha presentado una aplicación para bosques tropicales de forma que el usuario pueda acceder remotamente desde cualquier parte del mundo a ella, analizar y descargar los datos que necesite. a partir de la infraestructura en la nube propuesta se ha presentado una aplicación para la vid de forma que el usuario pueda gestionar y tomar decisiones en su viñedo a partir de los datos de satélite y de los datos de los sensores instalados en sus parcelas. También existe la posibilidad de añadir cualquier tipo de fuente de dato externa que el usuario considere necesaria para su gestión.

Perreau de Pinninck (2015) es de gran utilidad en la sociedad actual, que cada vez posee más datos y mayor necesidad de extraer un valor de ellos. Es por ello, que la plataforma con el caso de uso particular que se ha desarrollado se considera de gran utilidad para realizar dichos análisis estadísticos que permitan conocer mejor el comportamiento de los usuarios en la red social Twitter. Además, la plataforma de big data permite realizar en un futuro un análisis más profundo, introduciendo el análisis de sentimiento, que conlleve a unos resultados más detallados sobre la información.

Teodoro Rodríguez (2014) concluye en que se propuso analizar y mejorar los mecanismos utilizados para la obtención de usuarios influenciadores en redes sociales. Luego de estudiar en profundidad el estado del arte, diversos puntos críticos y falencias del método fueron detectados, para lo cual planteó una serie de propuestas de mejora enfocadas en resolver las limitaciones encontradas, optimizar la performance y mejorar la calidad de los resultados. Se propuso idear un sistema que permita involucrar más activamente al usuario final, permitiéndole definir de forma dinámica sus patrones de búsqueda. Por último, desarrollamos una aplicación que implemente las propuestas realizadas y nos permita validar las mismas, a partir de los resultados obtenidos mediante la utilización de un set de datos testigo.

Sirera Martínez (2015) concluye en que se ha logrado desarrollar un catálogo de servicios Big Data aptos para PYMES con los que adquirir conocimiento de datos internos y externos a la empresa para mejorar tanto sus productos como la relación con sus clientes. Algunos servicios como Metrikea, Movintracks o TC Store están pensados casi exclusivamente para comercios, pero el resto son de propósito más general y se usan en empresas de cualquier tipo.

1.5.2 Antecedentes nacionales

Garvich San Martín (2017) concluye en que se desarrolló una propuesta de arquitectura de análisis de datos no estructurados con las herramientas de la plataforma de Big Data de IBM, las cuales impactan positivamente en la generación de decisiones oportunas al reducir los tiempos de extracción, procesamiento, análisis y visualización de datos. Asimismo, impactan positivamente en la reducción de costos al brindar un análisis de datos en tiempo real. De acuerdo con los resultados obtenidos, más del 50% de los gestores de proyectos indican que actualmente los procesos de extracción, procesamiento, análisis y visualización de datos toman aproximadamente 1 día, llegando incluso a tardar semanas o meses dependiendo de la complejidad y/o volumen de los datos. Por lo tanto, la propuesta de análisis de datos no estructurados, mediante las herramientas de la plataforma Big Data de IBM, permitirán automatizar estos procesos causando una reducción en los tiempos de respuesta a sólo segundos o minutos. En Conclusión, se acepta la hipótesis general donde se determina que la propuesta de análisis de datos no estructurados favorecerá la generación decisiones oportunas en la fase de implementación de los proyectos de GMD, mediante el uso de las herramientas

IBM InfoSphere BigInsights, Streams, Information server y Cognos BI.

Mérida Fonseca & Rios Alvarado (2014) concluye en que se ha propuesto una plataforma de big data orientada al sector turístico para cubrir las deficiencias que presentan plataformas de big data como Oracle e IBM, en cubrir aspectos importantes en un sector que está creciendo a una velocidad muy rápida debido al consumo masivo de nuevas tecnologías móviles. Esta propuesta tiene como finalidad el uso de los procesos internos del sector turístico para ser aprovechados como fuentes de información y, considerando tecnologías abiertas como Hadoop, permite hacer un análisis más profundo de las características y comportamientos de los consumidores y/o potenciales clientes. El diseño, tiene como finalidad sentar las bases para una futura implementación de la plataforma.

Figueroa Chire (2014) concluye en lo siguiente: 1. La prevalencia de anemia en gestantes atendidas en el Hospital Hipólito Unanue de Tacna en el año 2013 fue de 20,7%. 2. La prevalencia de gestantes atendidas con anemia leve fue de 17,78%. 3. La prevalencia de gestantes atendidas con anemia moderada fue de 2,92%. 4. La prevalencia de gestantes atendidas con anemia severa fue de 0,00%. 5. Los factores sociodemográficos más frecuentes en las gestantes con anemia fueron: las gestantes con edades entre 25 – 29 (26,94%), las gestantes con estado civil de convivientes (78.88%). Los principales antecedentes obstétricos de las gestantes con anemia, fueron: Segundíparas (33,81%). El estado nutricional que prevaleció en las gestantes con anemia fue el BUEN ESTADO NUTRICIONAL (47,89%).

Higa Martinez (2017) concluye en que se determinó el efecto del Desarrollo de la Operación del Servicio en el Registro de una incidencia en la división de Aplicaciones de la empresa Viettel Perú, Debido a que, se redujo la Media antes 10.588 después 5.059 lo que quiere decir el promedio de Número de incidencias asignadas de manera incorrecta se redujo casi a la mitad. Con respecto a la Mediana antes 10 después 4 lo que quiere decir que el número de incidencias se redujo a la mitad. Con esto se determinó que el Desarrollo en la Operación del Servicio bajo el enfoque de ITIL v3 obtuvo un efecto positivo en la Gestión de Incidencias en la división de Aplicaciones dentro de la empresa Viettel Perú.

Milla Caballero (2017) en resumen se diseña en UML parte del proceso propuesto (modelamiento y despliegue), y se implementa en MATLAB utilizando como

herramienta de Data Mining al software libre WEKA (Waikato Environment for Knowledge Analysis) Finalmente se valida la metodología propuesta, con datos reales disponibles de un ISP típico (TdP), aplicando árboles de decisión como técnica de modelamiento, y se logra mostrar que la efectividad de las campañas mejora considerablemente.

Ocsa Mamani (2015), concluye en que el estudio de los algoritmos de búsqueda por similitud, enfocado en los algoritmos aproximados. La adaptación de algoritmos para arquitecturas paralelas en CPU y GPU. Estudio de la importancia de los algoritmos aproximados asociada a la computación GPGPU en la identificación de motifs. Los resultados experimentales mostraron que para el método de identificación de motifs, implementado por el algoritmo CUDA-TopKMotifs, al utilizar soluciones aproximadas de búsqueda y las capacidades de computación de uso general de las GPUs se consigue mejorar el desempeño de los algoritmos. Estos resultados mostraron también el equilibrio entre desempeño y precisión de los algoritmos, que utilizan la búsqueda kNN como un procedimiento importante, es garantizado debido a la utilización de técnicas aproximadas de búsqueda lo que permite un costo sublinear para datos en altas dimensiones. De modo general, los resultados de estos estudios mostraron que una utilización adecuada de las técnicas aproximadas y programación multi-thread en problemas complejos de recuperación y minería de datos garantiza un aumento significativo en el desempeño de estos, especialmente para algoritmos que demanden grandes recursos de computación.

Ortega Arana (2018) concluye sobre las empresas que necesitan un modelo de negocio para mejorar en la toma de decisiones de las PYMES del sector Retail de Lima Metropolitana. Considerando que este modelo se ajustó según la necesidad de cada PYME, el tiempo que se invierte en generar reportes importantes, la calidad de información, los procesos de información, el presupuesto para invertir en tecnología, una buena gestión de proyectos, implementando un modelo que contribuya a la eficiencia, eficacia y efectividad en la toma de decisiones; todo esto planteado en las 12 encuestas que se realizaron para este estudio. También podemos afirmar que según lo que se contrastó en la prueba RHO de SPEARMAN existe correlación de 0,932 y de acuerdo al baremo de estimación de la correlación de SPEARMAN existe una correlación positiva perfecta y con un nivel de significancia de 0.01.

1.5.3 Antecedentes locales

Carpio (2012) concluye sobre la anemia que, es un factor determinante según el riesgo relativo, ya que la presencia de anemia hace 7 veces más susceptible de que los niños tengan bajo peso al nacer, de las 56 madres que fueron evaluadas en el Hospital "Antonio Barrionuevo" – Lampa en su periodo de gestación por el Servicio de Laboratorio y Consultorio obstétrico. El bajo peso al nacer es el índice predictivo más importante de mortalidad infantil y el principal factor desencadenante de las más de 5 millones de muertes neonatales que ocurren anualmente en el mundo.

Coyla Idme (2016) concluye sobre las características y patrones de comportamiento en el desempeño académico de los ingresantes a la Universidad Nacional del Altiplano fueron identificados utilizando Rattle. Los paquetes como Rattle diseñados en el Lenguaje R fueron útiles para explorar, analizar y manipular base de datos gigantes. El 51 % de ingresantes conocen problemas matemáticos I. El 62% de ingresantes no resuelven problemas de matemática II. El 68% de ingresantes marcaron erradamente las alternativas de las preguntas referidos a Física, el 67 % de ingresantes marcaron erradamente las alternativas de las preguntas referidas a Química, el 53 % de ingresantes conocen problemas de razonamiento matemático y el 64 % de ingresantes conocen problemas de razonamiento verbal.

Holguín Holguín (2014) concluye de la siguiente forma: Primera: Al recuperar información semántica se optimiza al aplicar métodos que evalúan los vectores de coocurrencia de dos palabras que tienen cierto grado de similitud, calculando la frecuencia de ocurrencia de los mismos en un contexto determinado. Segunda. El modelo de espacio de palabras permite determinar la lejanía o cercanía de un par de términos, usando un espacio multidimensional, tomando en cuenta su distribución con el resto de términos del lenguaje y cuyo número de dimensiones, depende del número de vocablos diferentes encontrados en el corpus de Google utilizado. Tercera. El algoritmo basado en la medida de similitud por coseno del ángulo que forman entre el vector de las palabras y el vector consulta permiten ponderar y obtener términos con cierta similaridad semántica. Cuarta. Los términos encontrados por la métrica de semejanza del coseno del ángulo varían considerablemente entre un 0, 1 a un 0,9 dependiendo de los bigramas analizados y proporcionados por el Corpus de Google.

Pacompia Lara (2017) concluye sobre el algoritmo Subtractive Clustering que es una técnica rápida para estimar el número de clústeres y los centros de clúster en un conjunto de datos. Las estimaciones de clúster obtenidas a partir de esta se utilizan para inicializar métodos de agrupación basados en la optimización iterativa, como lo es el algoritmo PSO.

CAPÍTULO II

PLANTEAMIENTO DEL PROBLEMA

2.1 Descripción del problema de investigación

El acelerado avance de la tecnología y las plataformas móviles ha generado cuantiosa información. Estos grandes volúmenes en petabytes y terabytes, aún tienen dificultades en su estructuración y posterior procesamiento para obtener resultados que puedan ser en beneficio de la comunidad.

Los datos que los usuarios de las redes sociales como Twitter no se basa solamente en datos personales como nombres, dirección, sino que se ha ampliado teniendo en consideración la posición geográfica, lugares de visita, gustos, enfermedades y otros. De esta manera al tener información desestructurada no permite identificar tendencias de consumo, gustos o enfermedades como la anemia de las cuales se pueda cuantificar un ítem para beneficio de ofertas, estudios o investigación para tomar decisiones y dar mayor utilidad a la información. Es una oportunidad tener acceso al 1% de datos de Twitter para analizar el registro de tweets registrados respecto a la anemia. Pero el gran volumen de datos que se obtiene da lugar al uso de herramientas de procesamiento informáticas adecuadas, y una metodología para un adecuado tratamiento de la información.

Respecto a la anemia, representa el más extendido problema de salud y nutrición publica en el mundo. Se estima que más de 2 000 millones de personas (30 % de la población mundial) registran algún grado de anemia. La principal causa de la anemia es el déficit en el consumo de hierro, elemento principal para la formación de hemoglobina, que puede

ser exacerbado por las enfermedades infecciosas. (CMP, Mayo, 2018).

En consecuencia, el tema de big data, nos lleva a pensar cómo aprovechar esta cantidad de datos para generar un valor agregado sobre la información analizada. Si planteamos la posibilidad de recabar información en las redes sociales como Twitter respecto a anemia en cantidades voluminosas de información, ya que es una enfermedad extendida hasta en una 30% de la población, se hace posible plantear la investigación de big data respecto a casos de anemia, y se convierte en un problema y en una oportunidad que requiere la definición de una metodología que permita aprovechar los datos disponibles para extraer conocimiento.

Es así que al estar disponible data de Twitter y no realizarse análisis de la información generada, se pierde posibilidades de realizar aportes mediante investigaciones como la planteada en la presente tesis.

Por tanto, se plantea realizar es definir una metodología para el desarrollo e implementación de una plataforma big data para casos de anemia y evaluar resultados respecto a la implementación.

2.2 Enunciado del problema de investigación

2.2.1 Problema general

¿Cuál es la metodología adecuada para implementar una plataforma big data para realizar estudios de casos de anemia en América Latina, 2018?

2.3 Objetivos

2.3.1 Objetivo general

Implementar una plataforma big data para realizar el estudio de casos de anemia en América Latina, 2018.

2.3.2 Objetivos específicos

1. Definir una metodología para el desarrollo de una plataforma de big data.

2. Evaluar los resultados de la plataforma de big data para casos de anemia en América Latina.

2.4 Justificación de la investigación

En análisis de la información de la Web, es estructurada, semiestructurada o desestructurada, en el que se aplican técnicas de procesamiento o análisis de big data, para interpretarla o decir algo con datos precisos, cuantificados y con altos grados de certeza.

Las técnicas de big data son diversas, para implementar un diseño para nuestra investigación nos centramos en las herramientas a utilizar. Si precisamos que la búsqueda será respecto a la anemia, para países de América Latina, se cuantifica sólo lo indicado, por tanto, obtendremos resultados válidos para nuestra investigación.

2.5 Hipótesis

2.5.1 Hipótesis general

La metodología desarrollada permite implementar adecuadamente una plataforma big data para realizar estudios de casos de anemia en América Latina.

CAPÍTULO III

MATERIALES Y MÉTODOS

3.1 Lugar de estudio

La presente investigación se llevará a cabo en la ciudad de Puno, utilizando la plataforma big data ubicada, que abarcará el ámbito geográfico de América Latina.

3.2 Población

La población para esta investigación para efectos de validar la plataforma está constituida por la totalidad de tweets registrados en el API REST disponible para desarrolladores.

3.3 Muestra

La muestra se obtuvo mediante muestreo no probabilístico y está constituido por 7´192,687 tweets registrados a través del API REST

3.4 Métodos de investigación

Metodología para el desarrollo de una plataforma de big data

Para realizar la transmisión o streamming de información de Twitter se ha configurado Java 8, sus variables de entorno en Cloudera, también las variables de entorno de Flume, así como la parametrización de las variables de Flume.

Para realizar el almacenamiento de la información en hadoop se ha definido el tamaño de los paquetes en flume mediante las propiedades hdfs.rollcount, hdfs.rollinterval, MemChannel.capacity y MemChannel.transactionCapacity.

Respecto al Mapeo y Reducción de información se ha utilizado el framework MapReduce para la selección de la información que contiene la palabra “anemia”

Para las bases de datos, en NoSql se ha utilizado Hive debido a su capacidad para análisis de bases de datos semiestructurados y Mysql para procesar paralelamente a Hive.

Finalmente se ha utilizador la hoja de cálculo Excel, para cuantificar y visualizar los resultados de la información relevante y de interés de la investigación.

Las técnicas conceptuales utilizadas fueron la abstracción, análisis, sistematización y síntesis para el planteamiento de la metodología de la plataforma de big data.

Para la implementación de la plataforma, la técnica utilizada para la recolección de información fue la observación heurística y los instrumentos de recolección de datos fue una guía de observación.

CAPÍTULO IV

RESULTADOS Y DISCUSIÓN

4.1 Resultados de la metodología para el desarrollo de una plataforma de Big Data.

Se procedió a definir una metodología para el desarrollo de la plataforma que permita obtener, almacenar y hacer uso de los datos en cuestión.

Para ello, se definió las herramientas a emplear para realizar el tratamiento completo de la información. Estas herramientas, son compatibles con Cloudera y óptimas para el caso a desarrollar.

Las herramientas que se han seleccionado han sido las siguientes:

Flume. Apache Flume es un sistema confiable y disponible que se ha utilizado para recopilar, agregar y mover de manera eficiente grandes cantidades de datos de registro de fuentes diferentes como HDFS a un almacén de datos centralizado como Hive.

El uso de Apache Flume no solo está restringido a la agregación de datos de registro. Dado que las fuentes de datos son personalizables, Flume se usa para transportar cantidades masivas de datos de eventos, incluidos, entre otros, datos de tráfico de red, datos generados en las redes sociales, mensajes de correo electrónico y prácticamente cualquier fuente de datos posible, según Flume (2019). Es la herramienta que se utilizará para la recolección de los datos de HDFS a Hive.

HDFS. Es el sistema de almacenamiento propio de Hadoop, y se trata de un sistema de almacenamiento distribuido en ficheros que trata de optimizar el posterior tratamiento

que se quiera realizar de la información (Holmes, 2012). En nuestro caso se ha almacenado la data obtenida de los tweets recolectados mediante el API_REST.

MapReduce. Es un *job tracker* que permite reducir las grandes masas de información a un volumen muy inferior para realizar consultas mucho más rápidas y eficientes. Esta herramienta permitirá que se realicen los análisis a mayor velocidad (Holmes, 2012). Nos ha permitido realizar la reducción de la data obtenida en HDFS, antes de enviarlo a Hive, utilizamos MapReduce.

Hive. Apache Hive es una infraestructura de almacenamiento de datos construida sobre Hadoop para proporcionar agrupación, consulta y análisis de datos. (Venner, 2009). Se ha almacenado la data para realizar consultas, de manera similar al lenguaje sql.

MySQL. es un sistema de gestión de base de datos relacional (RDBMS) de código abierto, basado en lenguaje de consulta estructurado (SQL). (Rouse, 2018). Se realizó el almacenamiento de la información paralela a Hive, para realizar consultas en lenguaje sql.

Finalmente, para mostrar la **visualización** de datos se hará uso de **Excel**.

Por lo tanto, con las herramientas seleccionadas, se realiza un análisis completo de la información.

En la Figura 8 se muestra cómo se relacionan las diferentes herramientas que conforman el diseño de la plataforma big data que representa al sistema, indicando así el flujo de los datos.

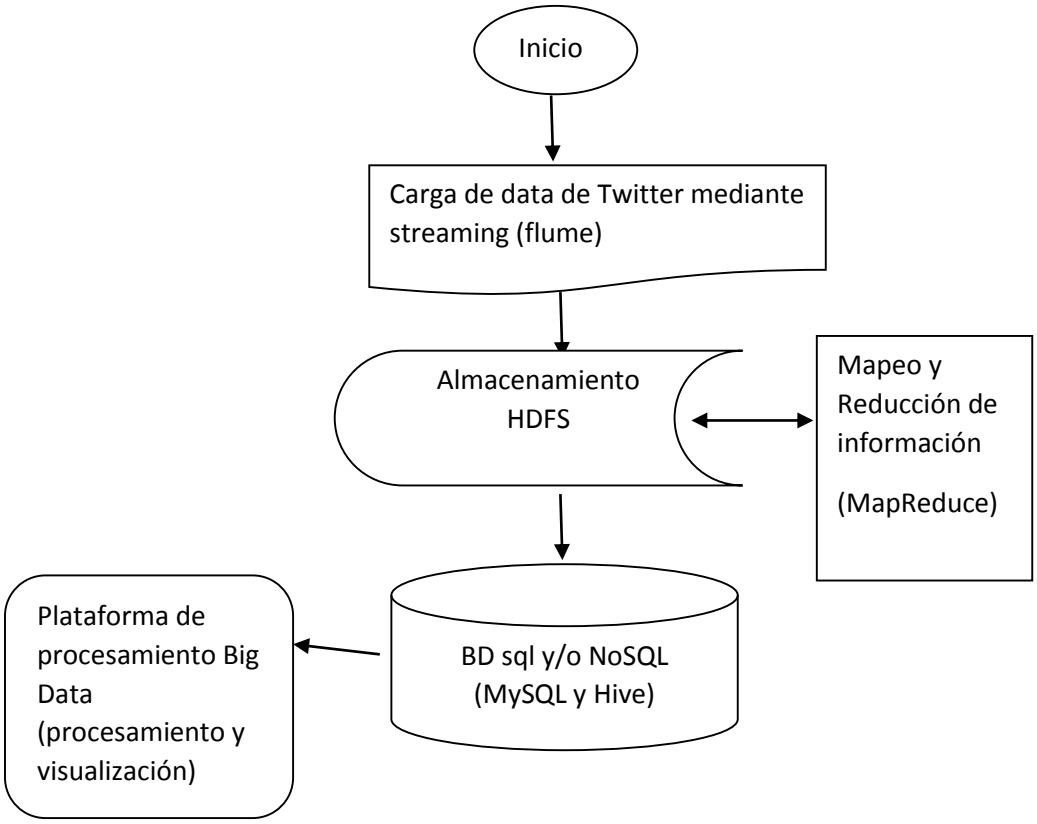

Figura 8. Diseño de la Plataforma Big Data

Como se observa, los datos se extrajeron de Twitter, con la API REST, para acceso a sus datos, de forma que estos se descargaron y trabajar con ellos. Estos tweets se descargan a través de la herramienta Flume, que los almacena en el sistema de ficheros HDFS.

En el siguiente paso, los datos se procesaron con MapReduce, que permite obtener una información mucho más simplificada y reducida para realizar consultas lo más óptimas posibles con la herramienta Mysql, que posee su lenguaje de consultas, que envía a la herramienta de visualización Excel los datos a representar.

Finalmente, con Excel se representa una serie de tablas y estadísticas sobre la información descargada inicialmente. Estos gráficos permiten obtener conclusiones para el análisis de la información.

Infraestructura necesaria para la implementación de la plataforma

En el servidor se instala una plataforma de virtualización, que permite hospedar los nodos del sistema. Esta plataforma es la propia de VirtualBox.

En esta virtualización se hospedan los nodos, que son máquinas virtuales con el sistema operativo Windows, estos nodos son clones de la máquina virtual que se tiene en una computadora de escritorio.

Como Hadoop se basa en una arquitectura maestro-esclavo, en la que existe un nodo maestro que se encarga de distribuir la carga de trabajo entre los nodos esclavos, será necesario que haya un nodo que sea maestro, realizando los servicios de *NameNode* de HDFS y *Job Tracker* en MapReduce. El resto de nodos son esclavos o nodos de trabajo, por lo que se ejecutaron en modo esclavo, *DataNode* y *TaskTracker*. El número de estos nodos es tres, para comprobar el funcionamiento de esta arquitectura. Las máquinas virtuales mencionadas anteriormente son las que hacen las funciones de los distintos nodos. El número de nodos esclavos dependerá de las necesidades del cliente, buscando siempre el punto óptimo en cuanto a rendimiento.

Para gestionar y administrar los nodos que se hospedan en el servidor, se utiliza VirtualBox con una interfaz gráfica que facilita el proceso. Permite conectarse al servidor, administrar el estado de las máquinas virtuales y acceder a la consola.

A continuación, se realiza la implementación de la plataforma propuesta. Una vez instalado todo el *software* en el equipo, se procede a recolectar los *tweets* para realizar el análisis de la información, siguiendo el esquema de la arquitectura propuesta para Big Data, y haciendo uso de las herramientas especificadas.

4.1.1 Ejecuciones de las herramientas

Al ejecutar los diferentes comandos que inician el funcionamiento de las diferentes herramientas se obtienen diversos resultados.

4.1.2 Resultados de flume

Una vez se ha lanzado el agente de Flume, en la consola aparece información acerca del proceso que se está ejecutando, de la cual se ha extraído la que se consideraba importante, mostrada a continuación, primero en la Figura 9 y luego el código ejecutado en la Figura 10.

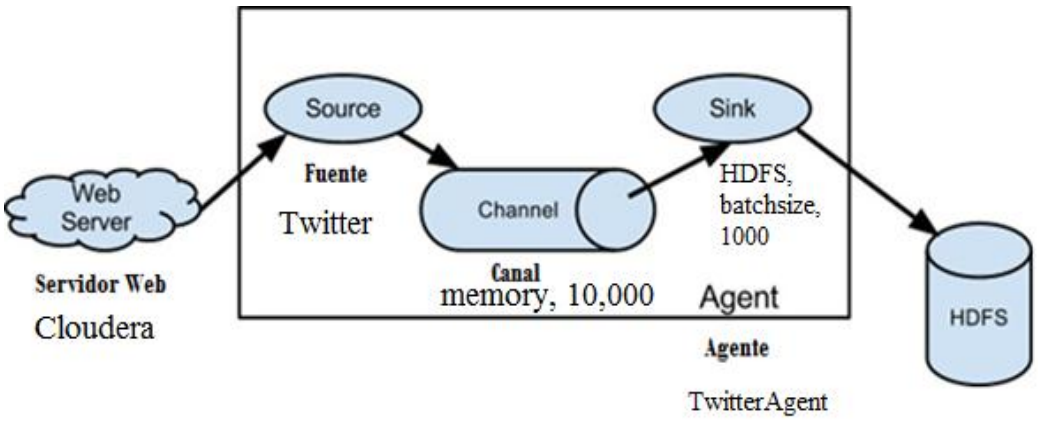

Figura 9. Aplicación de la Arquitectura de Flume

Fuente: (Flume, 2019)

```
[cloudera@quickstart -]$ hdfs dfs -rmr /user/cloudera/flume/output/
rmr: DEPRECATED: Please use 'rm -r' instead.
Deleted /user/cloudera/flume/output
[cloudera@quickstart ~]$ hadoop jar /usr/lib/hadoop-mapreduce/hadoop-streaming.jar -input /user/cloudera/flume -output /user
[cloudera/flume/output -mapper /home/cloudera/mapper_cuentapalabras.py -reducer /home/cloodera/reducer cuentapalabras.py
packageJobJar: [] [/usr/liD/hadoop-mapreduce/hadoop-streaming-2.6.0-cdh5.12.0.jar] tmp/streasjob8018278537687197628.jar tmp
Dir=null
07/01/19 13:30:50 INFO client.RMProxy: Connecting to ResourceManager at /0.0.0.0:8032
07/01/19 13:30:50 INFO client.RMProxy: Connecting to ResourceManager at /0.0.0.0:8032
07/01/19 13:30:51 INFO mapred.Fileinputforsat: Total input paths to process : 42
07/01/19 13:30:51 WARN hdfs.DFSClient: Caught exception
java.lang.InterruptedException
at java.lang.Objectorait(Native Method)
at java.lang.Thread.join(Thread.java:1281)
at java.lang.Thread.join(Tbread.java:1355)
at org.apache.hadoop.hdfs.OFSOutputStreanSbataStreaser.closeResponderIDFSOutputStreae.yava:9521
at org.apache.hadoop.hdfs.00SOutputStreae5DataStreamer.endelock(DFSOutputStrea.java:690)
at org.apache.hadoop.hdfs.OFSOutputStreasSoataStreaser.runIDESOutputStreakjava:8791
07/01/19 13:30:51 INFO mapreduce.JobSubmitter: number of splits:42
07/01/19 13:30:51 INFO mapreduce.340Submitter: Submitting tokens for Job: Job_1526772272287 0007
07/01/19 13:30:51 INFO impl.YarnClientImpl: Submitted application application 1526772272287 0007
07/01/19 13:30:51 INFO mapreduce.Job: The url to track the Job:
http://quickstart.cloudera:8088/proxy/application_1526772272 287 0007/
```

Figura 10. Ejecución del Agente

En la Figura 10 se observa cómo se establece correctamente la conexión con la fuente de la que se extraen los datos, en este caso Twitter, que conlleva a la descarga de la información almacenada en la ruta /user/cloudera/flume, en particular en el fichero tweets.1436467134063.

En este momento de la descarga, como aún se está llenando el fichero, se observa una extensión .tmp, que indica que está en proceso y aún no se accede a él.

El tiempo de descarga varía en función de la configuración que se haya establecido para el agente de Flume, puesto que se define el tamaño del fichero, la capacidad del canal de memoria, así como otros muchos aspectos. Como ya se mencionó, el tamaño de fichero buscado es de gran tamaño, por lo que el tiempo que tardará en llenarse cada uno de los ficheros será mayor que si se tratase de ficheros pequeños.

4.1.3 Resultados de HDFS

Una vez se haya llenado el fichero que se creó con el agente de Flume, éste quedará almacenado en el sistema de ficheros HDFS de la ruta especificada, de donde se podrá descargar y continuar con el trabajo.

Como ya se mostró anteriormente, se accede a los datos que se van descargando a través del navegador, gracias al buscador que se presenta a continuación, se muestra en forma gráfica en la Figura 11 y el resultado en Cloudera en la Figura 12.

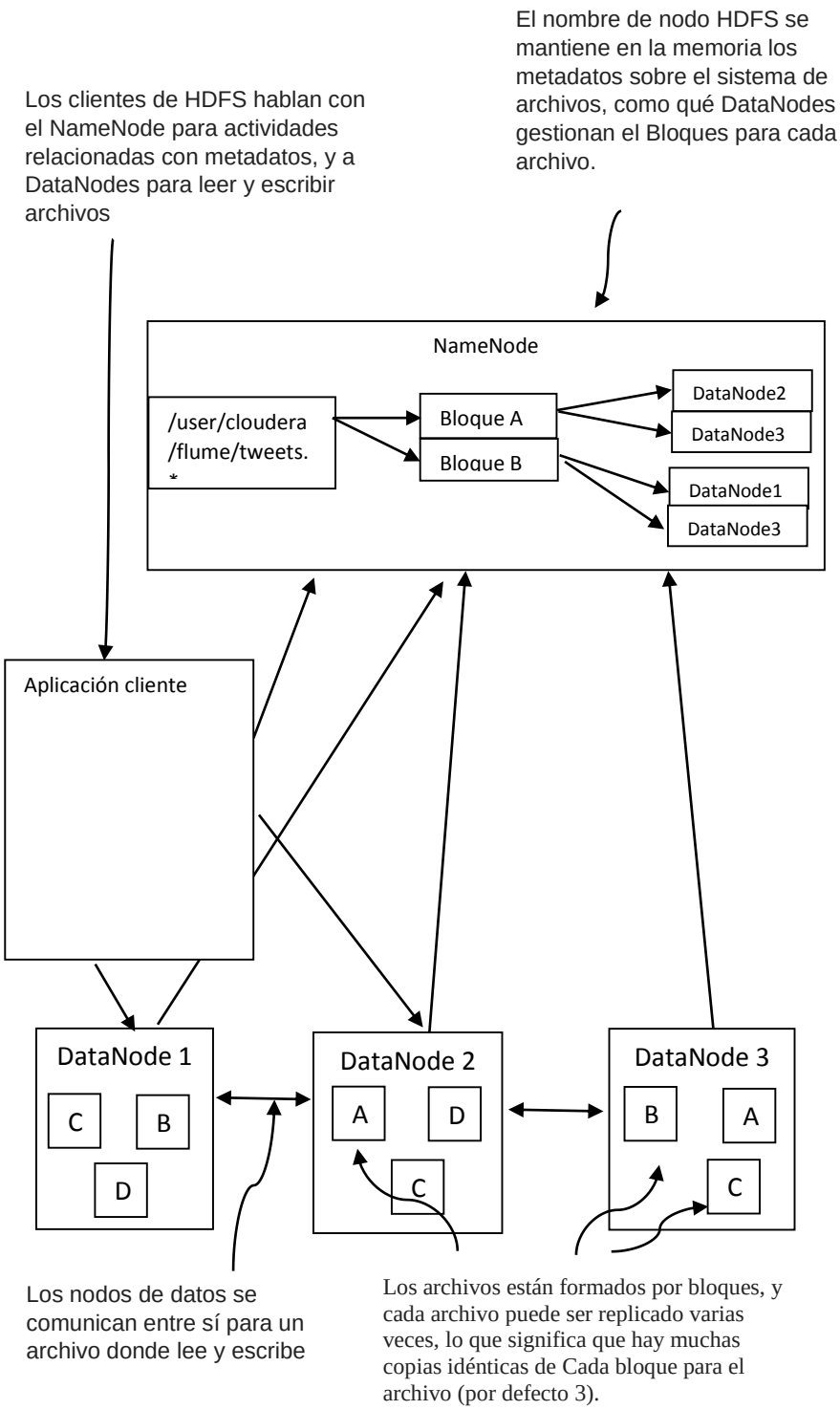

Figura 11. Arquitectura Hdfs

Fuente: Adecuado de (Alex, 2012)

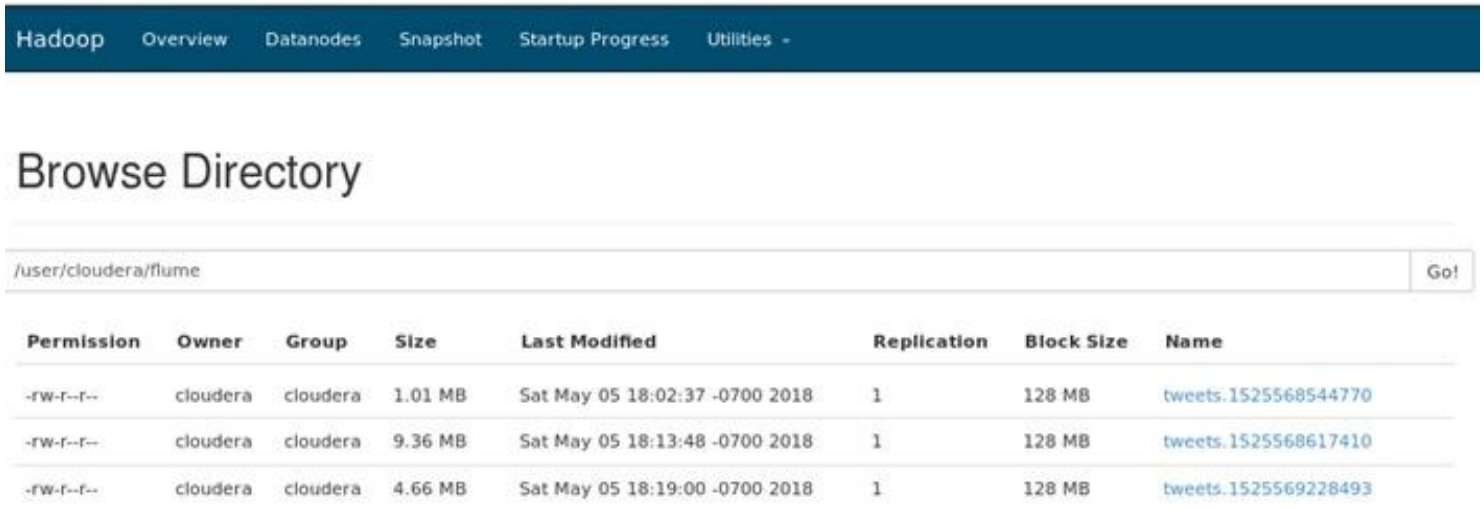

Figura 12. Sistema de Ficheros Hdfs conteniendo datos descargados

Se observa que cada uno de los ficheros de *tweets* tiene un tamaño diferente. Esto se debe a que cada uno se ejecutó realizando diversas pruebas en las que se variaba el tamaño del fichero para comprobar que efectivamente se reducía el tiempo de llenado cuando se disminuía el tamaño del fichero.

En cuanto a los permisos que se tienen sobre los ficheros, se observa que únicamente tiene permiso para escribir el sistema, y que el usuario únicamente tiene permiso para leer de ellos, lo que permite trabajar con ellos en otros ficheros, pero no modificar los existentes.

4.1.4 Resultados de Map Reduce

Una vez se han descargado los ficheros que contienen los tweets a analizar, se inicia el proceso de MapReduce. Esto, tal y como se definió en su configuración, iniciando dos procesos que se encargaron de esta tarea.

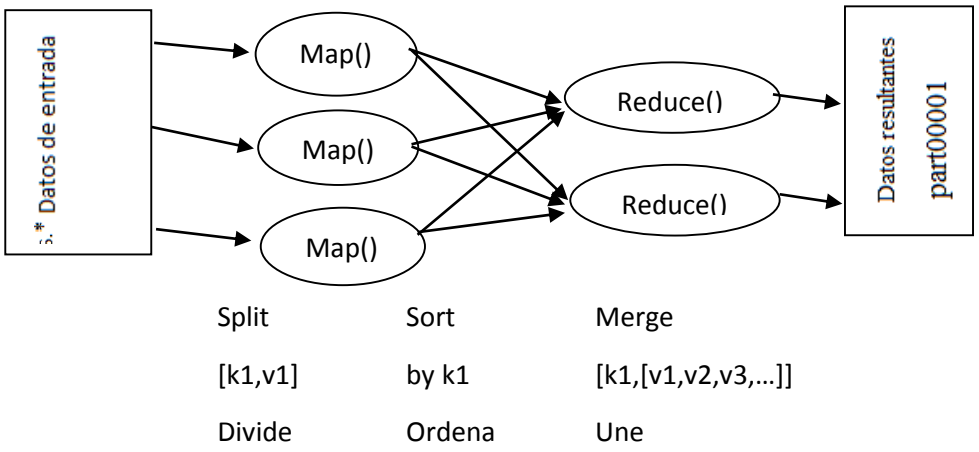

Figura 13. Mapreduce Aplicado

Fuente: (Niño, 2015)

En las imágenes que se muestran a continuación, se observa la información que se muestra por consola al ejecutar el proceso.

```
//Mapea el 100% del trabajo y llega a reducir el 100%
07/01/19 14:16:30 INFO mapreduce.job: map 100% reduce 98%
07/01/19 14:16:36 INFO mapreduce.job: map 100% reduce 99%
07/01/19 14:16:42 INFO mapreduce.job: map 100% reduce 100%

//indica que el trabajo(job) de nombre job_1528404483268_0001 se ha completado satisfactoriamente
07/01/19 14:16:43 INFO mapreduce.job: Job job_1528404483268_0001 completed successfully
//Los Contadores del trabajo son 51
07/01/19 14:16:43 INFO mapreduce.job: Counters: 51
File System Counters
//Archivo: numero de bytes leidos, escritos, con operaciones leidas, con operaciones escritas
FILE: Number of bytes read=439830765
FILE: Number of bytes written=885128724
FILE: Number of read operations=0
FILE: Number of large read operations=0
FILE: Number of write operations=0
//HDFS: numero de bytes leidos, escritos, con operaciones leidas, con operaciones escritas
 HDFS: Number of bytes read=364473698
 HDFS: Number of bytes written=240268323
 HDFS: Number of read operations=129
 HDFS: Number of large read operations=0
 HDFS: Number of write operations=2
Job Counters
//mapas eliminados=2
Killed map tasks=2
//mapas lanzados=42
Launched map tasks=42
//reducciones lanzadas
Launched reduce tasks=1
Other local map tasks=1
Data-local map tasks=41
//Tiempo total empleado por todos los mapas en ranuras ocupadas (ms) = 2287036
Total time spent by all maps in occupied slots (ms)=2287036
//Tiempo total empleado por todas las reducciones de las ranuras ocupadas (ms) = 520160
Total time spent by all reduces in occupied slots (ms)=520160
//Tiempo total empleado por todas las tareas del mapa (ms) = 2287036
Total time spent by all map tasks (ms)=2287036
//Tiempo total empleado por todas las tareas de reducción (ms) = 520160
Total time spent by all reduce tasks (ms)=520160
//Vcore-milisegundos totales tomados por todas las tareas de SAP = 2287036
Total vcore-milliseconds taken by all sap tasks=2287036
//Total de vcore.milliseconds tomadas por todas las tareas reducidas = 520160
Total vcore.milliseconds taken by all reduce tasks=520160
//Megabyte-milisegundos totales tomados por todas las tareas del mapa = 2341924864
Total megabyte-milliseconds taken by all map tasks=2341924864
//Total megabyte-milliseconds taken by all reduce tasks=532643840
Total de megabyte-milisegundos tomados por todas las tareas reducidas = 532643840
//Mapa-Reducir Marco
Map-Reduce Framework
//Mapa registros ingresados = 1085167
Map input records=1085167
```

Figura 14. Ejecución del Proceso Mapreduce (I)

Map-Reduce Framework//Marco Mapear-Reducir
Map input records=1085167
Map output records=18893415
Map output bytes=481619830
Map output materialized bytes=439831805
Input split bytes=5299
Combine input records=0
Combine output records=0
Reduce input groups=5353657
Reduce shuffle bytes=43983100S
Reduce input records=18893415
Reduce output records=5353657
Spilled Records=37786830
Shuffled Maps =42
Failed Shuffles=0
Merged Hap outputs=42
GC time elapsed (ms)=37832
CPU time spent (ms)=126100
Physical memory (bytes) snapshot=12105191424
Virtual memory (bytes) snapshot=64782696448
Total committed heap usage (bytes)=8556658688
Shuffle Errors //Errores del bloque
BAD ID=0
CONNECTION=0
IO_ERROR=0
WRONG_LENGTH=0
WRONG_MAP-0
WRONG_REDUCE=0
//Contadores de formato de entrada de archivo
File Input Format Counters
Bytes Read=364468399
//Contadores de formato de salida de archivo
File Output Format Counters
Bytes Written=240268323

Figura 15. Ejecución del Proceso Mapreduce (II)

En la Figura 13 se muestra el proceso en forma gráfica, en la Figura 14 y Figura 15 se observa cómo se muestra el progreso del proceso lanzado, de forma que se realiza primero la tarea de mapear y en el siguiente lugar se observa cómo se reducen los datos. También se incluye información del tamaño del fichero que se ha procesado, así como de los errores producidos, que en este caso son nulos.

Por otra parte, Cloudera ofrece un servicio a través del cual se visualiza en el navegador el progreso de los sistemas de MapReduce de forma gráfica, en http://quickstart.cloudera:8088/cluster.

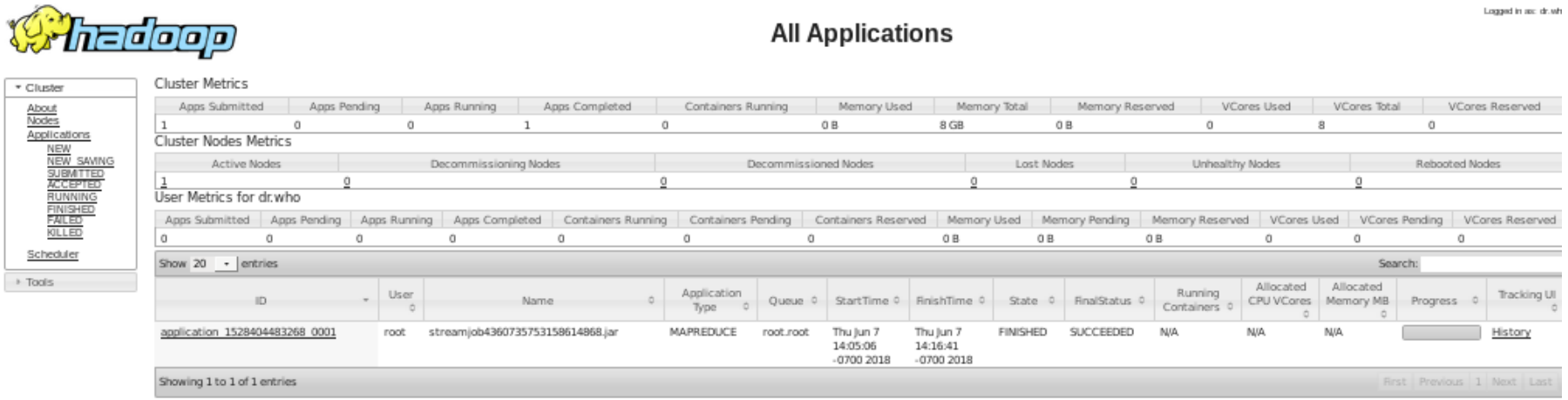

Figura 16. Interfaz Gráfica que Muestra el Progreso de Mapreduce

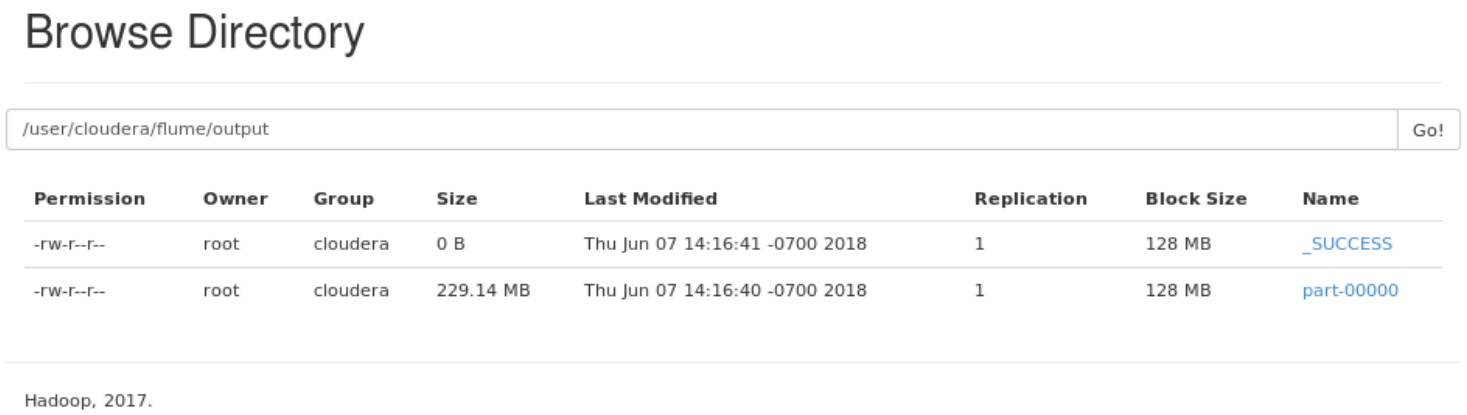

Figura 17. Hdfs conteniendo datos de la salida de Mapreduce

Al finalizar el proceso MapReduce que se lanzó, se observa dentro del sistema de ficheros HDFS, se muestra en la Figura 16 cómo se ha creado una carpeta nueva que contiene la información obtenida, tal como se muestra a continuación en la Figura 17.

Como se observa, se han creado dos ficheros. El primero, vacío, indicando que el proceso se ha completado con éxito. El siguiente es el del proceso paralelo que ejecuta MapReduce. En esta prueba en particular se hizo con un fichero muy pequeño, por lo que no aparecen datos en el primero y el segundo está prácticamente vacío, para ficheros de mayor tamaño se observa que ambos ficheros tienen aproximadamente el mismo tamaño.

4.1.5 Resultados de Hive y MySQL

Al ejecutar las sentencias que crean tablas en Hive, el sistema aporta unos mensajes indicando el tiempo de ejecución que ha tardado en completar la sentencia, así como si se ha completado correctamente o no.

```
//crea la table tweets1
hive(twitter)>CREATE TABLE tweets1(
        //con el campo ”palabra” de tipo String
        >palabra String,
        // con el campo ”cuenta” de tipo Int
        >cuenta Int)
        //FORMATO DE LA FILA DELIMITADO
        >ROW FORMAT DELIMITED
        //CAMPOS TERMINADOS POR ´ \ 011 ’
        >FIELDS TERMINATED BY ´\011’
        //ALMACENADO COMO TEXTFILE;
        >STORED AS TEXTFILE;
OK
Time taken: 0.094 seconds
```

Figura 18. Tiempo de Creación de una tabla en Hive

En la Figura 18 se muestra el tiempo que tarda el sistema en cargar los datos en la propia tabla. Como se observa, el tiempo es claramente superior al tiempo de creación, pero sigue siendo muy pequeño. Esto se debe a que el contenido que se

ha cargado no ocupa mucho espacio y por lo tanto, se procesa rápidamente. Sin embargo, si se tratase de una carga mucho mayor, el rendimiento se vería afectado.

```
//recupera data en 'user/cloudera/proyecto/part-r-00000'
hive (twitter)> LOAD DATA INPATH 'user/cloudera/proyecto/part-r-00000'
//en la tabla tweets1
> INTO TABLE tweetsl:
//Cargando datos a la tabla twitter.tweets1
Loading data to table twitter.tweets1
chgrp: changing ownership of 'user/hive/warehouse/twitter.db/tweetslipart-r-00000':
//El usuario no pertenece a hive
User does not belong to hive
Table twitter.tweetsl stats: [num_ partitions: 0, num_files: 2, num .rows: 0, total
size: 6320, raw_datasize: 0]
OK
Tine taken: 0.391 seconds
```

Figura 19. Tiempo de carga de datos en una tabla en Hive

En la Figura 19, al ejecutar las sentencias que crean tablas en Mysql, el sistema nos envía mensajes indicando el tiempo de ejecución que ha tardado en ejecutar la sentencia, y si se ha ejecutado correctamente o no. Se crea la tabla "tweets" con la siguiente sentencia:

```
mysql>create table tweets(palabra varchar(250), cuenta Int);
```

La siguiente sentencia para cargar datos en la tabla tweets:

```
mysql> LOAD DATA LOCAL INFILE
'/home/cloudera/Downloads/part-00000' INTO TABLE tweets
COLUMNS TERMINATED BY '\011'
```

4.2 Evaluación de resultados de la plataforma de Big Data para casos de anemia

A continuación, se va a mostrar los gráficos obtenidos de los datos de las tablas creadas en Mysql y hive. La herramienta Excel es la encargada de representar los datos, y ofrece una amplia variedad de opciones en cuanto a gráficos y tablas, para satisfacer los requisitos que se necesitan para realizar los análisis oportunos.

En primer lugar, se representaron los gráficos obtenidos de los datos que se han procesado con MapReduce. Estos datos corresponden a una sucesión temporal, por lo que se estudió el progreso existente mediante gráficos de diagrama de barras.

En los diferentes análisis, se ha estudiado el comportamiento del número de tweets en función del minuto en el que se han creado y en función de la hora.

Primero se estudió la ocurrencia cada 10 minutos. Sólo se obtiene una muestra, el 1% de los datos, que así lo permite Twitter, y realiza un análisis aproximado del comportamiento general de los datos que se generan en la red social.

En la Figura 20 que se muestra a continuación se observa cómo el número de tweets generados cada 10 minutos no sigue ningún tipo particular de sucesión, por lo que no se ha mostrado todo el espacio temporal que se tenía descargado, sino una pequeña parte para mostrar el resultado.

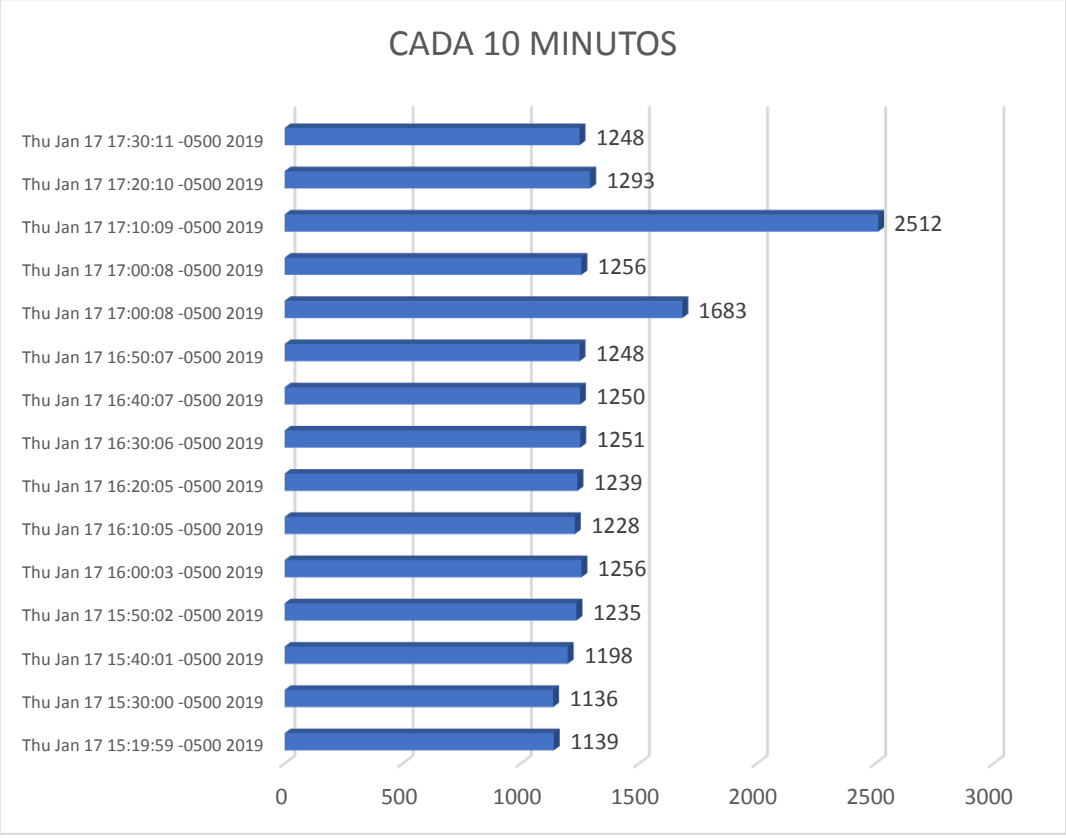

Figura 20. Sucesión Temporal por minutos de los Tweets

Sin embargo, de esta misma muestra se extraen la Figura 21 y la Figura 22, mucho más interesantes acerca del número de tweets generados. Se trata de un diagrama de caja, que muestra mucha información acerca de la muestra obtenida.

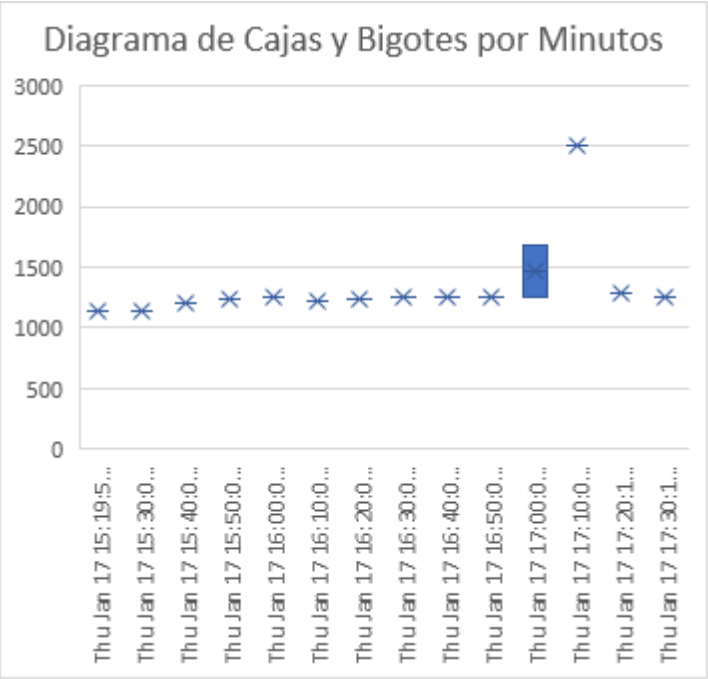

Figura 21. Diagrama de Caja de la Evolución por Minutos

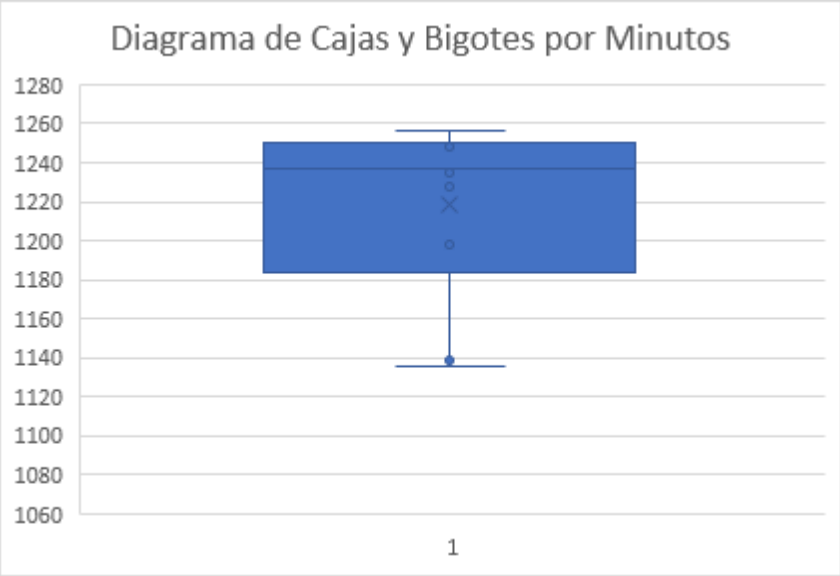

Figura 22. Diagrama de Caja por Rango

Otra forma para analizar los datos es con la Tabla 1, que muestra el número de datos que se genera por minuto, de forma que cada fila se colorea en función de su valor, Total de tweets obtenidos = 7´192,687. También se adjunta la escala de color con su significado, es decir, los minutos en los que se generaron menos tweets se presentan en blanco y cuanto mayor sea dicho número, más gris se coloreará el minuto correspondiente.

Recordemos que la anemia es el déficit en el consumo de hierro, elemento principal para la formación de la hemoglobina que puede ser exacerbado por las enfermedades infecciosas. (CMP, Mayo, 2018)

Como se observa, la mayor parte de las filas tiene un valor pequeño, sólo se superan los cuatro *tweets* por minuto que contenga la palabra anemia.

Tabla 1

Registro de evolución por minutos.

Fecha	Archivo	N° Tweets	N° Casos de Anemia
Wed Jan 16 12:08:39 -0500 2019	tweets.1.547657919860	15450	1
Wed Jan 16 13:08:45 -0500 2019	tweets.l.547661525858	11958	0
Wed Jan 16 14:08:52 -0500 2019	tweets.1.547665131898	12060	l
Wed Jan 16 15:08:58 -0500 2019	tweets.1547668737922	11811	0
Wed Jan 16 16:09:04 -0500 2019	tweets.1547672344013	11737	0
Thu Jan 17 09:05:12 -0500 2019	tweets.1547733309531	8976	0
Thu Jan 17 10:05:16 -0500 2019	tweets.1547736916864	9945	1
Thu Jan 17 10:35:19 -0500 2019	tweets.1547738719861	447	0
Thu Jan 17 11:05:22 -0500 2019	tweets.1547740522915	8985	1
Thu Jan 17 12:09:41 -0500 2019	tweets.1547744381816	14475	1
Thu Jan 17 13:09:46 -0500 2019	tweets.1547747986840	14105	1
Thu Jan 17 14:09:52 -0500 2019	tweets.l547751592890	13107	1
Thu Jan 17 15:09:59 -0500 2019	tweets.1547755198843	13060	0
Thu Jan 17 15:19:59 -0500 2019	tweets.1547755799868	1139	0
Thu Jan 17 15:30:00 -0500 2019	tweets.1547756400837	1136	1
Thu Jan 17 15:40:01 -0500 2019	tweets.1547757001877	1198	0
Thu Jan 17 15:50:02 -0500 2019	tweets.1547757602874	1235	0
Thu Jan 17 16:00:03 -0500 2019	tweets.1547153203864	1256	2
Thu Jan 17 16:10:05 -0500 2019	tweets.1541158804978	1228	2
Thu Jan 17 16:20:05 -0500 2019	tweets.1547159405878	1239	0
Thu Jan 17 16:30:06 -0500 2019	tweets.1547760006863	1251	1
Thu Jan 17 16:40:07 -0500 2019	tweets.1547760606964	1250	4
Thu Jan 17 16:50:07 -0500 2019	tweets.1547161207824	1248	2
Thu Jan 17 17:00:08 -0500 2019	tweets.1547161808833	1683	1
Thu Jan 17 17:00:08 -0500 2019	tweets.1547161808833	1256	1
Thu Jan 17 17:10:09 -0500 2019	tweets.1547762409840	2512	0
Thu Jan 17 17:20:10 -0500 2019	tweets.1547763010824	1293	1
Thu Jan 17 17:30:11 -05002019	tweets.1547163611843	1248	1
Total de tweets de la muestra		166288	23

En la Figura 23, se muestra la cantidad de veces que se ha encontrado la palabra anemia por tweets encontrados en las consultas realizadas.

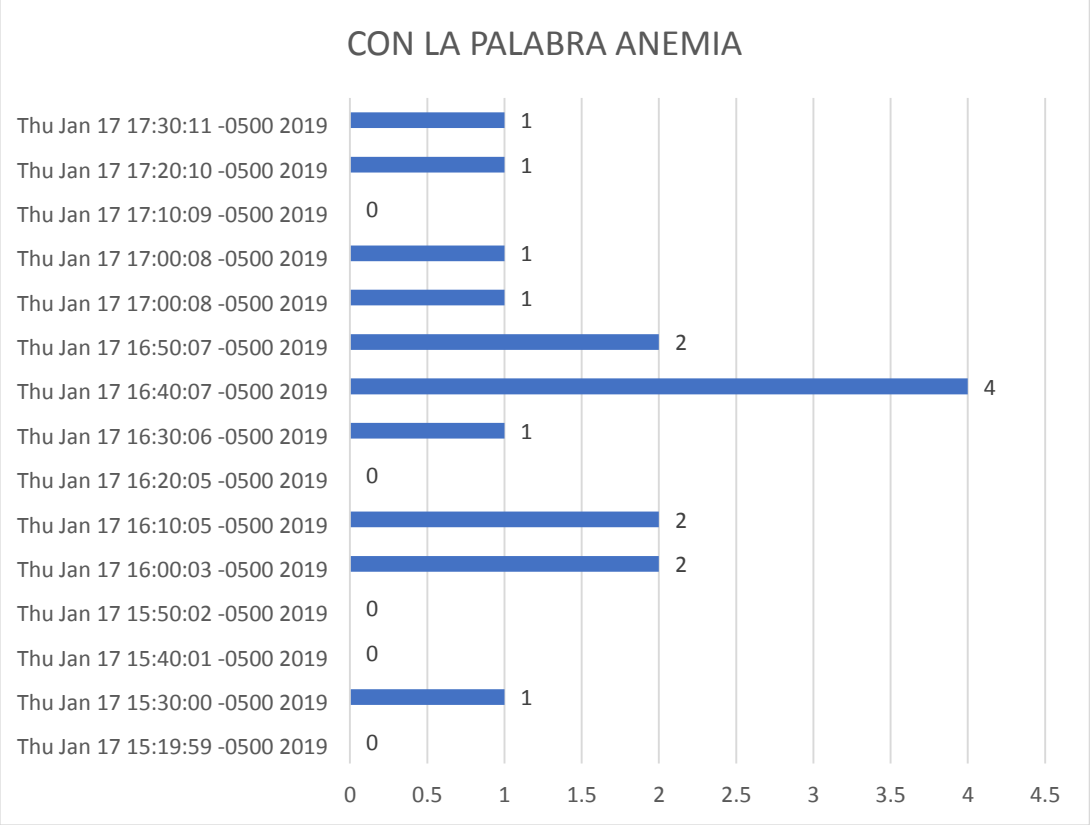

Figura 23. Repeticiones de la palabra Anemia en los Tweets

A continuación, se analiza la progresión por horas. En este caso, se observa un ligero patrón en el número de tweets que se han generado. Se debe destacar que en las horas más tempranas se generan muchos menos tweets, mientras que a medida que avanza la mañana se generan muchos más.

Figura 24. Sucesión Temporal por Horas de los Tweets

También se hace una representación de los datos con un diagrama de barras horizontal, como el que se muestra a continuación.

Finalmente se van a representar los datos que se obtuvieron al procesarlos exclusivamente con Mysql.

CONCLUSIONES

- La metodología para desarrollar una plataforma big data debe seguir los siguientes pasos 1. Carga de data de twitter de su APIREST a hadoop mediante streaming de Flume, 2. Almacenamiento en HDFS, de hadoop 3. Mapeo y reducción de información, 4. Almacenamiento en una base de datos NoSql y Sql y Procesamiento y visualización de la información, como se muestra en la Figura 8, y utilizar las siguientes herramientas informáticas Flume para el streaming o transmisión, Hadoop en el almacenamiento, MapReduce, Hive, Mysql y Excel.

- El uso de la plataforma realizando streaming de redes sociales en Twitter, carga en base de datos Nosql y análisis de la información procesada ha permitido encontrar que la palabra anemia representa el 0.00032% del total de 7´192,687 de tweets recolectados.

- Los términos encontrados son semejantes a los buscados, de la palabra anemia, en 7´192,687 registros, se encontraron 23 veces, esto respecto al 1% de la data disponible en Twitter según la Tabla 2. Registro de evolución por minutos, que representaría el 0.00032% del total de la muestra.

RECOMENDACIONES

- Se recomienda realizar una comparación con más distribuciones de big data, así como acceso a redes sociales, para estudios en diversos temas de interés.
- Se recomienda utilizar más herramientas de streaming y redes sociales, que podrán enriquecer su conocimiento y acceso a bases de datos de redes sociales.
- Se recomienda utilizar diversas distribuciones para big data locales y en la nube, ya que así se obtiene resultados con otras herramientas.
- Se recomienda utilizar otras técnicas adicionales de procesamiento de big data para investigaciones posteriores, así como redes sociales que permitan utilizar streaming.

BIBLIOGRAFÍA

Ballestar de las Heras, M. T. (2018). *Análisis del comportamiento del consumidor en comercio electrónico mediante técnicas y metodologías Big Data.* Universidad Rey Juan Carlos, Madrid, España.

Big data, D. (07 de febrero de 2017). *NoSQL and Analytics*. Obtenido de https://bigdatadummy.com/2017/02/07/apache-flume/

Carpio, M. (2012). *Prevalencia de anemia en gestantes relacionado al recién nacido con bajo peso al nacer en el hospital Barrionuevo Provincia de Lampa, departamento de Puno(Tesis de grado).* Universidad Andina Néstor Cáceres Velasquez. Lampa, Puno, Perú.

CMP, C. m. (Mayo, 2018). La anemia en el Perú ¿qué hacer? *Reporte de políticas de salud*, 3.

Coyla Idme, E. (2016). *Análisis de datos con Bigdata en procesos de admisión de la Universidad Nacional del Altiplano de Puno, 2016 (Tesis de doctorado).* Universidad Nacional del Altiplano. Puno, Perú.

Dijcks, J. P. (Junio de 2013). *Oracle: Big Data for the Enterprise.* Obtenido de http://www.oracle.com/us/products/database/big-data-for-enterprise-519135.pdf

Donald, M., & Adam, S. (2013). *MapReduce Design Patterns.* United States of America: O´reilly.

F.A.O., O. d. (2018). *FAO.* Obtenido de http://www.fao.org/3/I9553ES/i9553es.pdf

Figueres Cañadas, J. (2017). *Big Data, Ampliación Cognitiva, Procesos de Autoorganización y Desarrollo Económico.* Universidad Autónoma de Madrid. Madrid, España.

Figueroa Chire, Z. B. (2014). *Prevalencia de amenia en gestantes atendidas en el hospital Hipólito Unanue de Tacna, en el año 2013 (Tesis de grado).* Universidad Nacional Jorge Basadre Grohman. Tacna, Perú.

Flimper. (2018). *Estadísticas globales de Twitter 2018*. Obtenido de https://www.flimper.com/blog/es/estadisticas-globales-de-twitter-2018-

Flume, A. O. (08 de Enero de 2019). *Flume 1.9.0 user guide*. Obtenido de https://flume.apache.org/FlumeUserGuide.html#twitter-%C2%AD%E2%80%901-%C2%AD%E2%80%90firehose-%C2%AD%E2%80%90%20source-%C2%AD%E2%80%90experimental

Galeano Cruz, L., & Domínguez Rivera, D. A. (2017). *Prototipo de laboratorio Hadoop para Análisis Big Data en la Instituciòn Universitaria Politècnico Grancolombiano.* Institución Universitaria Politécnico Grancolombiano, Bogotá, Colombia.

Garvich San Martín, K. E. (2017). *Propuesta de Análisis de Datos no estructurados para generar decisiones oportunas en la empresa GMD (Tesis de grado).* Universidad San Ignacio de Loyola. Lima, Perú.

González-Benito, G. (2016). *Internet, comunicación y sociedad red:Algoritmos para un periodismo multiconectado.* Universidad Carlos III de Madrid. Madrid, España.

Guerrero López, F. A., & Rodríguez Pinilla, J. E. (2013). *Diseño y desarrollo de una guía para la implementación de un ambiente big data en la Universidad Católica de Colombia (Tesis de grado).* Univesidad Católica de Colombia. Bogotá, Colombia.

Hernández-Leal, E. J. (2016). *Aplicación de técnicas de análisis de datos y administración de Big Data ambientales (Tesis de maestría).* Universidad Nacional de Colombia. Medellin, Colombia.

Hernández-Leal, E., Duque-Méndez, N., & Moreno-Cadavid, J. (2017). *Big Data: una exploración de investigaciones, tecnologías y casos de aplicación.* Universidad Nacional de Colombia. Medellín, Colombia.

Higa Martinez, M. E. (2017). *Desarrollo de la Operación del Servicio para la Gestión de Incidencias en la división de Aplicaciones de la empresa Viettel Perú S.A.C.(Tesis de grado).* Universidad César Vallejo. Lima, Perú.

Hive, A. (2019). *Apache hive TM*. Obtenido de https://hive.apache.org/

Holguín Holguín, E. (2014). *Recuperación Semántica de la Información usando la Similitud Distribucional (Tesis de maestría).* Universidad Nacional de Altiplano. Puno, Perú.

Holmes, A. (2012). *Hadoop in practice.* Manning Publications C. Estados Unidos de América.

IDS, P. (28 de noviembre de 2018). *Prometeus global solutions*. Obtenido de https://prometeusgs.com/el-ecosistema-hadoop-y-su-impacto-en-la-eficiencia-en-la-gestion-de-datos-masivos/

Intellipaat. (04 de marzo de 2018). *intellipaat*. Obtenido de https://intellipaat.com/blog/special-features-new-hadoop-3-0/

Manso, F. (2015). *Análisis de modelos de negocios basados en big data para operadores móviles.* Universidad de San Andrés. Buenos Aires, Argentina.

Menasalvas, E. (2015). Big Data: el futuro a través de los datos. *Revista Universidad Politécnica de Madrid*.

Mérida Fonseca, C. M., & Rios Alvarado, R. P. (2014). *Impacto de la data warehouse e inteligencia de negocios en el desempeño de las empresas: investigación empírica en Perú, como país en vías de desarrollo (Proyecto).* Universidad Peruana de Ciencias Aplicadas. Lima, Perú.

Milla Caballero, H. H. (2017). *Propuesta de metodología para mejorar la efectividad de las campañas comerciales de un ISP utilizando Data Mining (Tesis de maestría).* Universidad Nacional de Ingeniería. Lima, Perú.

Moclan Soria, C. (2016). *Teledetección espacial, de los métodos clásicos al big data (Tesis de doctorado).* Valladolid, España: Universidad de Valladolid.

Niño, M. (10 de febrero de 2015). *MapReduce: el origen de la era Big Data*. Obtenido de Blog Mikel Niño: http://www.mikelnino.com/2015/02/map-reduce-origen-era-big-data.html

NIST, B. D. (2018). *Open Data Center Alliance.* Obtenido de https://bigdatawg.nist.gov/_uploadfiles/NIST.SP.1500-1r1.pdf

Ocsa Mamani, A. V. (2015). *Soluciones aproximadas para algoritmos escalables de minoración de datos en Dominios Complejos (Tesis de grado).* Universidad Nacional de San Agustín de Arequipa. Arequipa, Perú.

Ortega Arana, N. B. (2018). *Modelo de inteligencia de negocios para mejorar la toma de decisiones en las pymes del sector retail de Lima metropolitana (Tesis de grado).* Universidad Nacional Federico Villarreal. Lima, Perú.

Pacompia Lara, A. J. (2017). *Organización de datos multidimensionales en un sistema de recomendaciones basado en data clustering e inteligencia de enjambres (Tesis de grado).* Universidad Nacional del Altiplano. Puno, Perú.

Perreau de Pinninck, L. d. (2015). *Análisis y Desarrollo de una Plataforma Big Data (Tesis de grado).* Universidad Pontifica Comillas. Madrid, España.

Point, t. (2017). *Apache flume, introduction*. Obtenido de https://www.tutorialspoint.com/apache_flume/apache_flume_introduction.htm

Rodrìguez García, E. (15 de 06 de 2018). *ICEX Next*. Obtenido de https://www.slideshare.net/emirodgar/bases-de-datos-nosql-en-entornos-big-data

Rouse, M. (Abril de 2017). *Big Data.* Obtenido de https://searchdatacenter.techtarget.com/es/definicion/Big-data

Rouse, M. (08 de Mayo de 2018). *Guía Esencial: Las bases de datos dan soporte a las tendencias de TI*. Obtenido de https://searchdatacenter.techtarget.com/es/definicion/MySQL

Silberschatz, A. (2001). *Fundamentos de bases de datos*. Obtenido de https://es.wikipedia.org/wiki/Base_de_datos

Sirera Martínez, A. (2015). *Estudio sobre uso de Big Data en pymes.* España: Universidad Oberta de Catalunya.

Systems, A. (28 de noviembre de 2017). *Big data with NoSql*. Obtenido de https://www.aspiresys.com/casestudies/BigData_with_NoSQL_Whitepaper.pdf

Tejada, Zoiner. (27 de noviembre de 2017). *Microsoft Azure*. Obtenido de https://docs.microsoft.com/es-es/azure/architecture/data-guide/big-data/

Teodoro Rodríguez, F. (2014). *Data Mining en el cálculo de Influencia en redes sociales (Tesis de grado).* Universidad de Buenos Aires. Buenos Aires, Argentina.

Venner, J. (2009). *Pro Hadoop.* Apress.

ANEXOS

Anexo 1. Matriz de Consistencia

Tabla 2

Matriz de Consistencia

PROBLEMAS	OBJETIVOS	HIPÓTESIS	VARIABLE
Problema general	**Objetivo general**	**Hipótesis general**	**V. independiente**
¿Cuál es la metodología adecuada para implementar una plataforma big data para el estudio de casos de anemia en América Latina?	Implementar una plataforma de big data para realizar el estudio de casos de anemia en América Latina, 2018.	La implementación de la plataforma de big data ha permitido obtener resultados válidos para el estudio de casos de anemia en América Latina, 2018.	Plataforma big data
	Objetivos específicos	**Hipótesis específicas**	**V. dependiente**
	1. Definir una metodología para el desarrollo de una plataforma de big data. 2. Evaluar los resultados de la plataforma de big data para casos de anemia en América Latina	1. Se ha definido una metodología para el desarrollo y diseño de una plataforma de big data. 2. Se ha evaluado los resultados obtenidos en la plataforma de big data para casos de anemia en América Latina.	Casos de anemia en américa Latina

Anexo 2. Configuración, ejecución y almacenamiento de datos con flume

Configuración

La fuente de los datos proviene de Twitter directamente de un servidor.

```
TwitterAgent.sources.Twitter.type = org.apache.flume.source.twitter.TwitterSource
TwitterAgent.channels= MemChannel
```

Para tener acceso a datos de Twitter, es necesario crear una aplicación de desarrollador a partir de un usuario de la red social. Esta aplicación aporta al usuario 4 claves secretas que serán necesarias en la propia configuración del agente para permitir dicho acceso (*Consumer key*, *Consumer secret*, *Access token* y *Access tokensecret*).

Al configurar la fuente se establecen los criterios de búsqueda que se han definido para la presente investigación. Se deberán definir unas coordenadas geográficas que limiten el territorio, así como una palabra clave, que será el *hashtag* de búsqueda de los *tweets*. También se definen otras características de menor relevancia, como el tamaño de los ficheros de descarga.

```
TwitterAgent.sources.Twitter.consumerKey=ih3Nbi0GkDS8GPB8yWJn8x3Hx

TwitterAgent.sources.Twitter.consumerSecret=HnG7W8oigZOTVadtjdU9FxLcFYN9FR6b80nj9
bsDkZEiM2t5xB

TwitterAgent.sources.Twitter.accessToken=870321510753480704-
NtCl5AFmNu0SfHa4lAryySi8j3BuEde

TwitterAgent.sources.Twitter.accessTokenSecret=oTlsytx3HCsFsI77Q0z42W0uR1rSWlCKLfst6
KLg8ZEr4

TwitterAgent.sources.Twitter.channels=MemChannel

TwitterAgent.sources.Twitter.keywords= #anemia

TwitterAgent.sources.Twitter.swLngLat= -55.043180,10.043100

TwitterAgent.sources.Twitter.neLngLat= -77.028240,-35.320000
```

Aquí, se define del tipo memoria, indicando la capacidad que tendrá este canal.

```
twitterAgent.channels.c1.type=memory

twitterAgent.channels.c1.capacity = 10000

twitterAgent.channels.c1.transactionCapacity = 100
```

Finalmente se definirán las características de HDFS, por lo que se deberá especificar la ruta de destino y algunos otros atributos, como el tipo de fichero del que se trate o el formato en el que se guardará la información.

twitterAgent.sinks.k1.type=hdfs twitterAgent.sinks.k1.channel=c1

twitterAgent.sinks.k1.hdfs.path=/user/cloudera/flume/tweets/%{file}

twitterAgent.sinks.k1.hdfs.filePrefix=tweets

twitterAgent.sinks.k1.hdfs.fileType=DataStream

twitterAgent.sinks.k1.hdfs.writeFormat=Text

Resultaría de la siguiente forma:

TwitterAgent.sources.Twitter.consumerKey=ih3Nbi0GkDS8GPB8yWJn8x3Hx

TwitterAgent.sources.Twitter.consumerSecret=HnG7W8oigZOTVadtjdU9FxLcFYN9FR6b80nj9bsDkZEiM2t5xB

TwitterAgent.sources.Twitter.accessToken=870321510753480704-NtCl5AFmNu0SfHa4lAryySi8j3BuEde

TwitterAgent.sources.Twitter.accessTokenSecret=oTlsytx3HCsFsI77Q0z42W0uR1rSWlCKLfst6KLg8ZEr4

TwitterAgent.sources.Twitter.channels=MemChannel

TwitterAgent.sources.Twitter.keywords= anemia

TwitterAgent.sources.Twitter.swLngLat= -55.043180,10.043100

TwitterAgent.sources.Twitter.neLngLat= -77.028240,-35.320000

TwitterAgent.sinks.HDFS.channel=MemChannel

TwitterAgent.sinks.HDFS.type=hdfs

TwitterAgent.sinks.HDFS.hdfs.path=/user/cloudera/flume

TwitterAgent.sinks.HDFS.hdfs.filePrefix=tweets

TwitterAgent.sinks.HDFS.hdfs.fileType=DataStream

TwitterAgent.sinks.HDFS.hdfs.writeFormat = Text

TwitterAgent.sinks.HDFS.hdfs.batchSize=1000

```
TwitterAgent.sinks.HDFS.hdfs.rollSize=0

TwitterAgent.sinks.HDFS.hdfs.rollCount=10000

TwitterAgent.sinks.HDFS.hdfs.rollInterval=600

TwitterAgent.channels.MemChannel.type=memory

TwitterAgent.channels.MemChannel.capacity=10000

TwitterAgent.channels.MemChannel.transactionCapacity=100
```

Ejecución

Para ejecutar el agente de Flume se debe ejecutar el siguiente comando.

```
flume-ng agent -n TwitterAgent -c conf -f /usr/lib/apache-flume-1.4.0-bin/conf/flume.conf
```

Se observa que se lanza un agente de Flume, cuyo fichero de configuración se encuentra en la carpeta indicada con –conf, el nombre del agente que se ejecuta viene indicado con –c y el fichero en cuestión se referencia con el indicador –f. Finalmente se añade una indicación para mostrar por la consola información de proceso.

Almacenamiento de datos

Para el almacenamiento de los datos se emplea la herramienta HDFS, *Hadoop Distributed File System*, es decir, un sistema de ficheros distribuido.

El uso de este sistema es transparente, puesto que no se ve cómo almacena este sistema los ficheros ni los módulos que crea. No obstante, se sabe cómo es el sistema en sí, teniendo un nodo maestro y varios nodos esclavos que almacenan los ficheros en partes, de forma que cada parte irá a un nodo diferente, gestionado por el maestro.

Con el sistema Hadoop, se accede a través del navegador al sistema de archivos para visualizar qué contenidos se encuentran almacenados en la ruta que se ha especificado en la configuración del agente de Flume. El acceso es a través de la URL http://quickstart.cloudera:50070/explorer.html#/user/cloudera/flume que muestra las carpetas existentes.

Por lo tanto, se considera que los ficheros están almacenados de la forma más eficiente posible y que se accede a dichos datos de una forma muy sencilla para trabajar con ellos; sólo es necesario acceder a través de la ruta en la que se encuentran almacenados. Para

trabajar con estos datos, Hadoop se basa en los comandos de Linux, como –cp para copiarlos o --- rm para borrarlos.

Procesamiento de datos

En este caso, los datos no requieren de un análisis de gran profundidad, se deben procesar con un sistema de MapReduce. Por lo tanto, los datos se procesan para verse reducidos.

mapper_cuentapalabras.py

```
#!/usr/bin/env python
import sys
for linea in sys.stdin:
        linea = linea.strip()
        claves = linea.split()
for clave in claves:
        valor = 1
        print('{0}\t{1}'.format(clave, valor) )
```

reducer_cuentapalabras.py

```
#!/usr/bin/env python
import sys
ultima_clave = None
total_palabra = 0
for linea in sys.stdin:
linea = linea.strip()
   clave, valor = linea.split("\t", 1)
   valor = int(valor)
   if ultima_clave == clave:
      total_palabra += valor
   else:
      if ultima_clave:
print( "{0}\t{1}".format(ultima_clave, total_palabra) )
        total_palabra = valor
        ultima_clave = clave
if ultima_clave == clave:
print( "{0}\t{1}".format(ultima_clave, total_palabra))
```

Ejecución mapreduce

Una vez se ha instalado la herramienta, es necesario construir el proyecto, indicándole el fichero de entrada, donde están los datos guardados en el HDFS, y el fichero de destino

donde se almacenarán una vez se hayan procesado, que también será el sistema de ficheros. El comando para ejecutarlo es el siguiente:

```
hadoop jar /usr/lib/hadoop-mapreduce/hadoop-streaming.jar -input /user/cloudera/flume -output
/user/cloudera/flume/output -mapper /home/cloudera/mapper_cuentapalabras.py -reducer
/home/cloudera/reducer_cuentapalabras.py
```

Como se observa, como entrada se cogen todos los ficheros existentes en esa ruta, puesto que se analizarán todos los datos descargados de Twitter.

Consulta de datos

Una vez que los datos se han reducido y se consulta de una forma mucho más sencilla, será necesario realizar las consultas oportunas para visualizar los resultados. Para ello, se emplea la herramienta Mysql, que proporciona el lenguaje de consulta SQL aplicable a información desestructurada.

Configuración MySQL

A continuación, se va a presentar los pasos necesarios para crear tablas en Mysql. En primer lugar, será necesario acceder a la *shell* de Mysql para luego crear una base de datos y dentro de ella crear la tabla con las columnas, una de tipo Varchar denominada palabra y una de tipo Int con la cuenta obtenida del MapReduce.

```
$ mysql

mysq > CREATE DATABASE twitter;
```

Los datos almacenados en HDFS se cargan en la tabla para que puedan ser consultados. Se muestran a continuación los comandos necesarios para la creación de la tabla dentro de la base de datos que se ha denominado *twitter*.

```
mysql>create table tweets(palabra varchar(250), cuenta Int);
```

También el comando para la carga de datos a la tabla tweets.

```
mysql>LOAD DATA LOCAL INFILE '/home/cloudera/Downloads/part-00000' INTO
TABLE COLUMNS TERMINATED BY '\011'
```