Introducción al BigData

Diego Sepúlveda

Motivación

El término de Big Data en general, se usa para las estrategias y tecnologías no tradicionales necesarias para recopilar, organizar, procesar y recopilar información de grandes volúmenes de datos. Si bien el problema de trabajar con datos que exceden la capacidad de computación o el almacenamiento de una sola máquina no es nuevo, la penetración, la escala y el valor de este tipo de informática se han expandido enormemente en los últimos años.

¿Qué es el Big Data?

Formalmente no hay una definición de Big Data que pueda describir con exactitud todo lo que engloba, pero si podemos mencionar ciertas característias que son comunes en un sistema de Big Data que podrían ayudar a definirlo:

"El Big Data significa que un conjunto de datos es muy grande para procesarse o almacenarse razonablemente con herramientas tradicionales o en una sola máquina."

¿Cómo podemos identificar un sistema de Big Data?

En 2001, Doug Laney de Gartner presentó lo que se conoce como las *"tres V de Big Data"* para describir algunas características que hacen que los sistemas de big data sean diferentes de otros procesos de datos:

Volumen

Estos conjuntos de datos pueden estar en órdenes de magnitud mayores que los conjuntos de datos tradicionales, lo que exige más reflexión en cada etapa del ciclo de vida de procesamiento y almacenamiento. A menudo, debido a que los requisitos de trabajo superan las capacidades de una sola máquina, esto se convierte en un desafío para agrupar, asignar y coordinar recursos de grupos de máquinas. La gestión de clusters y los algoritmos capaces de dividir las tareas en piezas más pequeñas cobran cada vez más importancia.

Velocidad

A menudo, se espera que los datos sean procesados en tiempo real para obtener información del sistema. En este enfoque la retroalimentación es casi instantánea. Los datos se analizan constantemente para mantenerse al día con la afluencia de nueva información y dar resultados valiosos desde el principio. Estas estrategias requieren sistemas robustos con componentes altamente disponibles para protegerse contra fallas a lo largo de la cadena de datos.

Variedad

Los datos pueden ser subministrados desde diversas fuentes, tanto internos o externos. El Big Data busca manejar datos **potencialmente útiles** independiente de dónde provengan al consolidar. Mientras los sistemas más tradicionales suelen esperar que los datos ingresen formateados y organizados, el Big Data **usualmente** almacena datos más cercanos a su estado original. Idealmente, cualquier transformación o cambio en los datos brutos ocurrirá en la memoria en el momento del procesamiento.

Otras características

Además de las tres *V originales*, se han descrito algunas cualidades

adiciones:

- **Veracidad**: la variedad de fuentes y la complejidad del procesamiento pueden generar desafíos para evaluar la validéz de los datos.
- **Variabilidad**: es posible que se necesiten recursos adicionales para identificar, procesar o filtrar datos de baja calidad.
- **Valor**: los sistemas establecidos son pueden ser complejos como para que el uso de los datos y la extracción del valor real se vuelvan difíciles.

¿Cómo se vería un ciclo de vida en Big Data?

Las categorías generales para el procesamiento de Big Data son:

1. Ingesta de datos en el sistema
2. Persistencia de datos
3. Análisis
4. Visualización de resultados

Clustered Computing (1/2)

Para poder cumplir con los requerimientos y exigencias, es posible que una sola máquina no sea capáz de cumplir con la demanda, es por ello que las herramientas usadas para el tratamiento de datos estan pensadas para el uso distribuido de los recursos.

Clustered Computing (2/2)

Algunos conceptos que debemos tener presentes son:

- **Distrubución de recursos**: la combinación de almacenamiento, CPU y memoria es extremadamente importante. Procesar grandes conjuntos de datos requiere grandes cantidades de recursos.

- **Alta disponibilidad**: los clusters proporcionan niveles de tolerancia a errores y garantías de disponibilidad para evitar que las fallas de hardware o software.
- **Escalabilidad**: los clusters facilitan la escalabilidad horizontal al agregar máquinas adicionales al grupo.

Ingesta de datos

La ingesta es tomar datos sin procesar y agregarlos al sistema. La complejidad depende del formato y la calidad de las fuentes de datos. Tecnologías como **Apache Sqoop** pueden tomar datos de bases de datos relacionales y agregarlos a un sistema de big data. Del mismo modo, **Apache Flume** y **Apache Chukwa** son para agregar e importar registros de aplicaciones y servidores. Los sistemas de colas como **Apache Kafka** también se pueden usar como una interfaz entre varios generadores de datos y un sistema de big data. Durante el proceso de ingestión, se suele tener cierto nivel de análisis, clasificación y etiquetado.

Persistencia de datos

La ingestión entrega los datos a los componentes de almacenamiento. Si bien parece que sería una operación simple, el volumen de datos entrantes, los requisitos de disponibilidad y la capa de computación distribuida hacen que sea necesario contar con sistemas de almacenamiento más complejos. Esto generalmente significa un sistema de archivos distribuidos. Soluciones como el sistema de archivos HDFS de **Apache Hadoop** permiten que se graben grandes cantidades de datos en múltiples nodos del clúster. Esto garantiza que se pueda acceder a los datos mediante recursos de cómputo, que se puedan cargar en la RAM del clúster para operaciones en memoria, y que pueda manejar con elegancia los fallos de los componentes. Se pueden usar otros sistemas de archivos distribuidos en lugar de HDFS, incluidos Ceph y GlusterFS. Las bases de datos distribuidas son adecuadas para este rol

porque se diseñan con las mismas consideraciones de tolerancia a errores.

Análisis de Datos (1/2)

Teniendo los datos disponibles, el sistema puede comenzar a procesarlos. La capa de cálculo es quizás la parte más diversa del sistema, ya que los requisitos y el mejor enfoque pueden variar significativamente según el tipo de conocimiento que se desee. A menudo, los datos se procesan de manera repetida, ya sea iterativamente con una sola herramienta o mediante el uso de varias herramientas para mostrar diferentes tipos de información. El **procesamiento por lotes** (batch) implica dividir el trabajo en partes más pequeñas, programar cada pieza en una máquina individual, reorganizar los datos en función de los resultados intermedios y luego, calcular y ensamblar el resultado final. Estos pasos a menudo se denominan individualmente como división, mapeo, mezcla, reducción y ensamblaje, o colectivamente como un algoritmo de reducción de mapas distribuidos.

Análisis de Datos (2/2)

Si bien el procesamiento por lotes es una buena opción para ciertos tipos de datos y cálculos, otras cargas de trabajo requieren más procesamiento en tiempo real. El procesamiento en tiempo real exige que la información se procese y prepare inmediatamente y requiere que el sistema reaccione a medida que se disponga de nueva información. Una forma de lograr esto es el procesamiento continuo, que opera en un flujo continuo de datos compuesto por elementos individuales. Otra característica común de los procesadores en tiempo real es la computación en memoria, que funciona con representaciones de los datos en la memoria del clúster para evitar tener que volver a escribir en el disco. Apache Storm, Apache Flink y Apache Spark proporcionan

diferentes formas de lograr un procesamiento en tiempo real o casi en tiempo real. Hay intercambios con cada una de estas tecnologías, que pueden afectar qué enfoque es el mejor para cada problema individual. En general, el procesamiento en tiempo real es el más adecuado para analizar trozos más pequeños de datos que están cambiando o se están agregando al sistema rápidamente.

Visualizando los Resultados

Debido al tipo de información que se procesa, reconocer las tendencias de los datos a lo largo del tiempo es más importante que los valores mismos. La visualización de datos es una de las formas más útiles de detectar tendencias y dar sentido a una gran cantidad de puntos de datos. El procesamiento en tiempo real se utiliza para visualizar las métricas de aplicaciones y servidores. Los datos cambian con frecuencia y los grandes deltas en las métricas generalmente indican impactos significativos en la salud de los sistemas u organizaciones. En estos casos, proyectos como Prometheus pueden ser útiles para procesar las secuencias de datos como una base de datos de series de tiempo y visualizar esa información. Una forma popular de visualizar datos es con Elastic Stack, anteriormente conocida como la pila ELK. Compuesto por Logstash para la recopilación de datos, Elasticsearch para indexar los datos y Kibana para la visualización, la pila Elastic se puede usar con los sistemas de big data para interactuar visualmente con los resultados de los cálculos o métricas sin formato. Se puede lograr una pila similar usando Apache Solr para indexar y una horquilla Kibana llamada Banana para visualización. La pila creada por estos se llama Seda. Otra tecnología de visualización típicamente utilizada para el trabajo interactivo de ciencia de datos es un "cuaderno de datos". Estos proyectos permiten la exploración y visualización interactiva de los datos en un formato propicio para compartir, presentar o colaborar. Ejemplos populares de este tipo de interfaz de visualización son Jupyter Notebook y Apache Zeppelin.

Anexo

Glosario

- **Batch Processing**: el procesamiento por lotes es una estrategia informática que implica procesar datos en conjuntos grandes. Esto es ideal para trabajos que no requieren mucho tiempo y que operan en grandes conjuntos de datos. El proceso se inicia y en un momento posterior, el sistema devuelve los resultados.

Glosario

- **Data lake**: Data lake es un término para un gran repositorio de datos recopilados en un estado relativamente crudo. Esto se usa con frecuencia para referirse a los datos recopilados en un sistema de big data que podría no estar estructurado y cambiar con frecuencia. Esto difiere en espíritu de los almacenes de datos (definidos a continuación).

Glosario

- **Data Mining**: la minería de datos es un término amplio para la práctica de tratar de encontrar patrones en grandes conjuntos de datos. Es el proceso de tratar de refinar una gran cantidad de datos en un conjunto de información más comprensible y coherente.

Glosario

- **Data Warehouse**: los almacenes de datos son depósitos de datos grandes y ordenados que se pueden usar para análisis e informes. A diferencia de un lago de datos, un almacén de datos se compone

de datos que se han limpiado, integrado con otras fuentes y, en general, está bien ordenado. A menudo se habla de almacenes de datos en relación con big data, pero normalmente son componentes de sistemas más convencionales.

Glosario

- **ETL**: ETL significa extraer, transformar y cargar. Se refiere al proceso de tomar datos en bruto y prepararlos para el uso del sistema. Tradicionalmente, este es un proceso asociado a los almacenes de datos, pero las características de este proceso también se encuentran en las tuberías de ingestión de los sistemas de big data.

Glosario

- **Hadoop**: Hadoop es un proyecto de Apache que fue el éxito inicial de código abierto en big data. Consiste en un sistema de archivos distribuido llamado HDFS, con una administración de clúster y un programador de recursos en la parte superior llamado YARN (Sin embargo, otro negociador de recursos). Las capacidades de procesamiento por lotes son proporcionadas por el motor de cálculo MapReduce. Se pueden ejecutar otros sistemas computacionales y de análisis junto con MapReduce en las implementaciones modernas de Hadoop.

Glosario

- **In-memory Computing**: la computación en memoria es una estrategia que implica mover los conjuntos de datos en funcionamiento por completo dentro de la memoria colectiva de un clúster. Los cálculos intermedios no se escriben en el disco y se guardan en la memoria. Esto le da a los sistemas de cómputo en

memoria como Apache Spark una gran ventaja en la velocidad sobre los sistemas de E / S vinculados como MapReduce de Hadoop.

Glosario

- **Machine Learning**: el aprendizaje automático es el estudio y la práctica de diseñar sistemas que pueden aprender, ajustar y mejorar en función de los datos que se les proporcionan. Esto generalmente implica la implementación de algoritmos predictivos y estadísticos que pueden concentrarse continuamente en el comportamiento y las percepciones "correctas" a medida que fluyen más datos a través del sistema.

Glosario

- **Map reduce** : Map reduce (el algoritmo de big data, no el motor de cálculo MapReduce de Hadoop) es un algoritmo para programar el trabajo en un clúster de computación. El proceso implica dividir la configuración del problema (asignarla a diferentes nodos) y calcular sobre ellos para producir resultados intermedios, mezclando los resultados para alinear conjuntos similares, y luego reduciendo los resultados al generar un único valor para cada conjunto.

Glosario

- **NoSQL**: NoSQL es un término amplio que se refiere a bases de datos diseñadas fuera del modelo relacional tradicional. Las bases de datos NoSQL tienen diferentes ventajas y desventajas en comparación con las bases de datos relacionales, pero a menudo son muy adecuadas para los sistemas de big data debido a su flexibilidad y su arquitectura de distribución frecuente.

Glosario

- **Stream Processing**: El procesamiento de flujo es la práctica de computar sobre elementos de datos individuales a medida que se mueven a través de un sistema. Esto permite el análisis en tiempo real de los datos que se están alimentando al sistema y es útil para operaciones sensibles al tiempo que usan métricas de alta velocidad