Caso de Estudio 4: Banco o Fintech

● Caso de Uso: Detección de fraude en tiempo real.

● Descripción Ampliada: Las entidades financieras enfrentan el reto constante de proteger las transacciones de sus clientes. Un sistema de detección de fraude tradicional es lento y puede fallar con patrones de ataque modernos. El Big Data lo resuelve así:

○ Análisis en Tiempo Real: Cada transacción de tarjeta de crédito, transferencia bancaria o movimiento en una cuenta genera datos que se analizan en milisegundos. El sistema no solo ve la transacción actual, sino que la compara con el historial de compras del cliente (ubicación, hora, monto, tipo de comercio) y con patrones de fraude conocidos a nivel global.

○ Modelos Predictivos: Un modelo de machine learning se entrena con millones de transacciones fraudulentas y legítimas para identificar anomalías. Si una compra no encaja con el comportamiento habitual del cliente o con los patrones
de la mayoría, el sistema puede bloquearla instantáneamente o enviar una alerta.

Las 5 V's

Volumen: Cada transacción de tarjeta de crédito, transferencia bancaria o movimiento en una cuenta genera una cantidad masiva de datos. Bancos como Walmart, por ejemplo, procesan más de un millón de transacciones por hora, acumulando petabytes de datos. La proliferación de dispositivos móviles y la interconexión de objetos también contribuyen a un volumen de datos sin precedentes en el sector financiero. Para detectar fraudes, es necesario analizar este vasto historial y el flujo constante de nuevas transacciones

Variedad: La información utilizada proviene de múltiples fuentes heterogéneas. Esto incluye datos estructurados (historial de compras del cliente: ubicación, hora, monto, tipo de comercio) desde bases de datos relacionales tradicionales, y también datos no estructurados o semiestructurados (como comentarios en redes sociales sobre un posible ataque, patrones de comportamiento de gasto que no son fácilmente tabulables)
"la variedad es de diferentes tipos de archivos"

Velocidad: Necesitas procesamiento en tiempo real - Cinco minutos es tarde. Las transacciones deben analizarse en milisegundos. El sistema debe comparar la transacción actual con el historial del cliente y patrones de fraude en tiempo real. Esto es crucial para bloquear operaciones sospechosas o enviar alertas instantáneas

Veracidad: En la detección de fraude, la fiabilidad y confiabilidad de la información son primordiales. Si los datos de las transacciones o los patrones de comportamiento no son veraces, los modelos predictivos podrían generar falsas alarmas o, peor aún, no detectar fraudes reales

Valor: Detección de fraude es generar valor económico para la entidad financiera. Al identificar y bloquear transacciones fraudulentas de forma rápida, se reducen pérdidas financieras y se protege la confianza del cliente, lo que se traduce en un mejor desempeño y rentabilidad para la

empresa. El análisis de patrones ocultos y la toma de decisiones basada en evidencia son cruciales para esta generación de valor

Almacenamiento

• ¿Dónde se almacenarán estos datos? ¿Creen que sería un sistema de archivos distribuido como HDFS, un Data Lake o una base de datos más tradicional?

Dado el **gran volumen de datos** generados por cada transacción y el historial de compras del cliente [Descripción Ampliada], así como la variedad de fuentes (ubicación, hora, monto, tipo de comercio [Descripción Ampliada], y posiblemente otros patrones de fraude que pueden ser semiestructurados o no estructurados), la solución más adecuada sería un **Data Lake** o un **Data Lakehouse**.

◦ Un **Data Lake** es un repositorio que almacena datos en su estado original, crudos y sin una estructura predefinida. Esto es ideal para la exploración de datos, ciencia de datos y proyectos de *Machine Learning*, donde los científicos de datos necesitan trabajar con los datos en su forma más pura y sin filtros. Esta flexibilidad permite almacenar datos estructurados, semiestructurados (como archivos JSON o XML) y no estructurados (como imágenes o texto libre).
◦ Un **Data Lakehouse** es un concepto más reciente que busca combinar la flexibilidad y el bajo costo de un Data Lake con la estructura y las herramientas de análisis de un Data Warehouse. Permite almacenar datos crudos y flexibles, pero con herramientas que añaden una capa de estructura y gestión para facilitar el uso de herramientas de *Business Intelligence* (BI) y análisis.
◦ Un **sistema de archivos distribuido como HDFS** (*Hadoop Distributed File System*) sería una parte fundamental de un Data Lake. HDFS está diseñado para almacenar archivos de gran tamaño con una filosofía de "escribir solo una vez y permitir múltiples lecturas", y es robusto para replicar datos con redundancia a través del clúster, asegurando que el proceso de cálculo no se interrumpa incluso si alguna parte falla.
◦ Aunque las **bases de datos relacionales tradicionales** (*Relational Database Management Systems - RDBMS*) siguen siendo una fuente fundamental y extendida de información estructurada y transaccional, no son tan eficientes para manejar cantidades masivas y escalables de datos no estructurados o para consultas analíticas complejas en grandes volúmenes. Para el análisis de Big Data, se han desarrollado alternativas como las bases de datos **NoSQL** y **"en memoria"** (*in-memory*).

• ¿Qué desafíos de escalabilidad y costo enfrentarían al almacenar estos datos?

Escalabilidad: El volumen de datos es inmenso y crece exponencialmente. Bancos como Walmart procesan más de un millón de transacciones por hora, acumulando petabytes de datos. Este *volumen* requiere una arquitectura que pueda escalar masivamente, lo que significa pasar de la era del petabyte a la del exabyte y zettabyte. Las soluciones deben ser **redimensionables**, permitiendo agregar nuevos nodos sin cambiar el formato o la forma de cargar los datos. Las bases de datos NoSQL y *in-memory* están diseñadas para alta escalabilidad y rendimiento.

Costo: Históricamente, el análisis de grandes volúmenes de datos con herramientas tradicionales era "prohibitivo" en términos de presupuesto. Sin embargo, la aparición de **software de código abierto** como **Apache Hadoop** ha hecho posible el procesamiento de grandes volúmenes de datos a un costo muy económico, utilizando clústeres de computadoras "ordinarias" o "básicas" (*commodity hardware*). La **computación en la nube** (*cloud computing*)

también reduce las inversiones iniciales y ofrece un costo predecible, permitiendo incluso a pequeñas *startups* alquilar tiempo en servidores. A pesar de esto, el costo total puede ser impredecible y la migración a la nube o la compatibilidad entre varias nubes puede implicar costos adicionales.

Procesamiento y Análisis:

• ¿Qué tipo de procesamiento se necesita (por lotes o en *streaming*)?

• Se necesita principalmente **procesamiento en tiempo real (*streaming*)**. Para la detección de fraude, "cinco minutos es demasiado tarde". Los datos deben ser procesados en el mismo momento en que se generan, fluyendo de manera constante y siendo procesados inmediatamente. Esto es crucial para aplicaciones que requieren una respuesta rápida, como la detección de fraude. Aunque el procesamiento por lotes (*batch*) es útil para datos que no requieren un análisis inmediato o para procesar grandes bloques de información periódicamente (como transacciones al final del día), no es suficiente para la detección proactiva de fraude.

• ¿Qué herramientas de análisis serían las más adecuadas (ej. SQL, Python, *machine learning*)?

◦ **Machine Learning (ML)**: Es fundamental. El caso de uso menciona explícitamente que un "modelo de *machine learning* se entrena con millones de transacciones fraudulentas y legítimas para identificar anomalías" [Descripción Ampliada]. ML es una rama de la inteligencia artificial que permite a las computadoras "aprender" de los datos e identificar patrones y hacer predicciones sin ser programadas explícitamente. Los científicos de datos utilizan el aprendizaje automático para construir algoritmos predictivos y probar y mejorar continuamente su precisión. Herramientas y bibliotecas como **Apache Mahout** (una biblioteca escalable de aprendizaje automático para Hadoop), **Scikit-learn** (en Python) son adecuadas.

◦ **Python**: Es un lenguaje de programación muy popular para tareas complejas de procesamiento y transformación de datos en entornos de Big Data. Su facilidad de uso y la disponibilidad de bibliotecas como Pandas y NumPy lo hacen ideal para la limpieza y transformación de datos, así como para algoritmos de *Machine Learning*. Los científicos de datos utilizan Python.

◦ **SQL (y HiveQL)**: Aunque los datos sean variados, SQL sigue siendo relevante. **Apache Hive** es una infraestructura de *data warehouse* sobre Hadoop que proporciona un lenguaje similar a SQL llamado **Hive Query Language (HQL)**, permitiendo a los usuarios familiarizados con SQL consultar grandes volúmenes de datos en un ambiente distribuido. SQL se puede usar para preparar datos para modelos de *Machine Learning*.

◦ **Herramientas de Procesamiento en Tiempo Real**: Para la *velocidad* requerida, herramientas como **Apache Spark** son esenciales. Spark es un motor de procesamiento de datos de código abierto que acelera el procesamiento y la analítica de Big Data, especialmente en memoria. También se pueden usar sistemas de computación distribuida en tiempo real como **Apache Storm** y plataformas de *streaming* como **Kafka** para flujos de datos en tiempo real entre sistemas.

◦ **OLAP (*Online Analytical Processing*)**: Este enfoque permite responder a consultas analíticas complejas y multidimensionales de manera rápida y eficiente, a menudo utilizado en *Business Intelligence* (BI).

◦ **Minería de Datos (*Data Mining*)**: Es el proceso de encontrar patrones y correlaciones en grandes conjuntos de datos. Es una parte fundamental de la ciencia de datos y es crucial para identificar patrones de fraude.

◦ **Análisis Predictivo (*Predictive Analytics*)**: Consiste en analizar los datos para construir modelos y pronosticar resultados futuros, lo cual aporta un valor añadido claro. Es una capacidad clave en la detección de fraude.

◦ **Ciencia de Datos (*Data Science*)**: Es el campo interdisciplinario que proporciona los métodos científicos, procesos, algoritmos y sistemas para extraer conocimiento y perspectivas de los datos, ya sean estructurados o no estructurados. Implica el uso de habilidades en matemáticas, estadística, informática y conocimientos del negocio.

Gobernanza y Seguridad:

• ¿Qué datos sensibles o personales podrían estar manejando? (ej. datos personales de clientes, historial de navegación)?
• Se manejarían datos altamente sensibles y personales, incluyendo:
◦ **Datos transaccionales**: Historial de compras del cliente (ubicación, hora, monto, tipo de comercio [Descripción Ampliada]), transferencias bancarias, movimientos de cuenta [Descripción Ampliada].
◦ **Datos personales**: Si se recogen, pueden incluir sexo, edad, gustos, hábitos, preferencias, aficiones, profesión.
◦ **Datos de comportamiento**: Patrones de comportamiento de gasto que no son fácilmente tabulables [Descripción Ampliada].
◦ **Datos generados por las personas**: Información de centros de llamadas, correos electrónicos, documentos electrónicos, registros médicos electrónicos (aunque el caso es financiero, las fuentes mencionan estos como ejemplos de datos sensibles).
• La gestión de estos datos se encuadra en la **privacidad de datos**, que es crucial.

• ¿Qué desafíos de seguridad y privacidad tendrían que considerar para proteger la información?
• La detección de fraude con Big Data plantea importantes desafíos en seguridad y privacidad:
◦ **Seguridad de los datos**: Proteger los datos personales contra el acceso no autorizado, el robo, la pérdida o el daño accidental es fundamental. Esto implica utilizar sistemas de seguridad robustos, como la encriptación de datos. La seguridad es uno de los "grandes riesgos" que afrontan las organizaciones con Big Data y *cloud computing*.
◦ **Privacidad de los datos y cumplimiento normativo (*Compliance*)**: Es primordial asegurar la **confidencialidad, integridad y disponibilidad (CIA)** de los datos. Se debe cumplir con las regulaciones de protección de datos nacionales e internacionales, incluyendo el derecho a que los datos sean borrados o anonimizados después de un tiempo justificado. La proliferación de datos desde diversas fuentes hace que asegurar la fiabilidad (*veracidad*) y el cumplimiento normativo sea un gran reto.
◦ **Calidad de los datos**: La *veracidad* (fiabilidad) de la información es primordial en la detección de fraude; si los datos no son veraces, los modelos predictivos pueden generar falsas

alarmas. Asegurar la calidad implica procesos de análisis sintáctico, estandarización, validación y verificación. La gran variedad de fuentes y la velocidad de ingesta pueden introducir errores e inconsistencias.

◦ **Gobierno de Big Data**: Es esencial establecer políticas para optimizar, proteger y potenciar la información, identificando datos sensibles y estableciendo políticas de uso aceptable. Esto incluye la gestión del ciclo de vida de los datos (desde su creación hasta su eliminación) para asegurar el cumplimiento legal y controlar el crecimiento y costo de los datos.

◦ **Integración y Arquitectura de Seguridad**: La arquitectura de Big Data debe integrar las nuevas tecnologías con las infraestructuras existentes, lo que plantea desafíos de seguridad, ya que los datos sensibles deben protegerse sin importar dónde residan. Los grupos de trabajo como el *Big Data Working Group* de la *Cloud Security Alliance (CSA)* se dedican a desarrollar soluciones para estos problemas de seguridad y privacidad.

◦ **Escasez de profesionales especializados**: La "escasez de profesionales especializados", como científicos e ingenieros de datos, dificulta la correcta gestión y protección de la información, ya que son ellos quienes deben diseñar e implementar los algoritmos y sistemas para el manejo y aseguramiento de los datos

mas caracteres