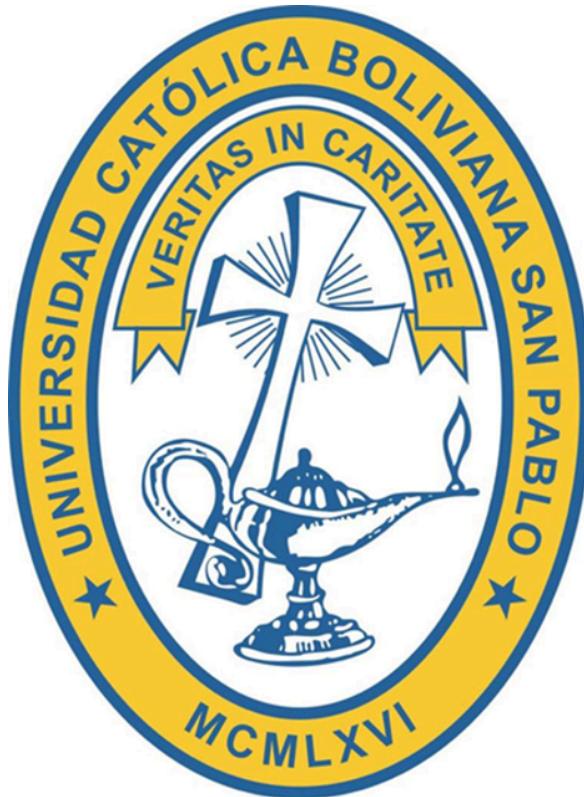


**UNIVERSIDAD CATÓLICA BOLIVIANA
“SAN PABLO” - SEDE LA PAZ**



**Análisis de Actividad de Usuarios en Publicaciones
de Facebook**

INTEGRANTES:

- SOTO FLORES PABLO EDUARDO
- MANZANEDA CASTRO JAVIER EZEQUIEL

DOCENTE: PACHECO ARTEAGA MIGUEL ANGEL

MATERIA: SIS 221 — BASE DE DATOS II

SEMESTRE: 2-2025

Perfil General del Proyecto

1. Introducción

En la actualidad, las redes sociales se han consolidado como uno de los principales medios de interacción social y digital a nivel mundial. Plataformas como Facebook concentran millones de usuarios que interactúan constantemente mediante publicaciones, reacciones y comentarios, generando grandes volúmenes de datos de forma continua. Estos datos, si son almacenados, organizados y analizados de manera adecuada, pueden transformarse en información valiosa para comprender el comportamiento de los usuarios y apoyar la toma de decisiones estratégicas.

El presente proyecto aborda el diseño e implementación de un sistema integral de gestión y análisis de datos para una red social basada en el funcionamiento de Facebook. El trabajo se desarrolla de manera progresiva, abarcando desde la construcción de una base de datos relacional, hasta la implementación de un Data Mart orientado al análisis multidimensional y la representación de la información en entornos NoSQL, complementado con técnicas de análisis y minería de datos.

En una primera etapa, se diseña una base de datos relacional utilizando PostgreSQL, la cual permite gestionar las principales entidades de una red social, tales como usuarios, publicaciones, comentarios y reacciones. En esta fase se aplican conceptos avanzados de bases de datos, como procedimientos almacenados, funciones, cursores y triggers, con el objetivo de automatizar procesos, garantizar la integridad de los datos y optimizar el rendimiento del sistema. Asimismo, se adopta el modelo ANSI/SPARC para asegurar una correcta separación de niveles y el uso de roles y permisos que fortalecen la seguridad de la información.

Posteriormente, el proyecto se orienta al análisis de datos mediante la construcción de un Data Mart basado en un esquema en estrella. Este modelo multidimensional permite analizar la actividad de los usuarios en publicaciones desde diferentes perspectivas, considerando dimensiones como el género del usuario, el tipo de publicación, el tipo de reacción y el tiempo. Para ello, se desarrollan procesos ETL (Extract, Transform, Load) que permiten transformar los datos operacionales del sistema relacional en información analítica, facilitando la identificación de patrones, tendencias y comportamientos relevantes.

Finalmente, se incorpora el uso de bases de datos NoSQL mediante MongoDB, con el propósito de representar una versión orientada a documentos del modelo relacional previamente desarrollado. A través de una colección que encapsula publicaciones junto con sus reacciones y comentarios embebidos, se demuestra la flexibilidad y eficiencia de los modelos NoSQL para manejar relaciones de uno a muchos sin la necesidad de operaciones complejas de unión. Esta etapa sienta las bases para el análisis exploratorio y predictivo de los datos, complementando el enfoque tradicional de las bases de datos relacionales.

De esta manera, el proyecto integra distintas tecnologías y enfoques de gestión de datos, evidenciando cómo la combinación de bases de datos relacionales, modelos analíticos y soluciones NoSQL permite transformar grandes volúmenes de datos en información útil,

relevante y orientada a la toma de decisiones tanto en contextos académicos como empresariales.

2. Objetivos del Proyecto

Diseñar e implementar un sistema integral de gestión y análisis de datos para una red social basada en el funcionamiento de Facebook, utilizando bases de datos relacionales y NoSQL, procesos ETL y modelos analíticos, con el fin de transformar los datos generados por la interacción de los usuarios en información útil para el análisis y la toma de decisiones.

2.1 Objetivos Específicos

- Diseñar una base de datos relacional en PostgreSQL que represente de forma clara y ordenada las principales entidades de una red social, como usuarios, publicaciones, comentarios y reacciones.
- Implementar funciones, procedimientos almacenados, cursores y triggers en PostgreSQL para automatizar procesos, mantener la integridad de los datos y facilitar la gestión de la información.
- Aplicar el modelo ANSI/SPARC mediante la definición de roles y permisos, con el fin de controlar el acceso a la información y mejorar la seguridad de la base de datos.
- Identificar las dimensiones más importantes para analizar la actividad de los usuarios en las publicaciones, considerando aspectos como el género, el tipo de publicación, el tipo de reacción y el tiempo.
- Construir un Data Mart utilizando un esquema en estrella que permita analizar la interacción de los usuarios desde diferentes puntos de vista y de manera más eficiente.
- Desarrollar un proceso ETL que permita extraer los datos desde la base relacional, transformarlos según las necesidades del análisis y cargarlos en el Data Mart.
- Realizar consultas analíticas que ayuden a identificar patrones y tendencias en el comportamiento de los usuarios dentro de la red social.
- Implementar una base de datos NoSQL en MongoDB que represente la información de las publicaciones mediante documentos, integrando datos de usuarios, reacciones y comentarios en una sola estructura.
- Ejecutar consultas en MongoDB para analizar la información almacenada y comparar el enfoque NoSQL con el modelo relacional.
- Utilizar herramientas de análisis y minería de datos como apoyo para obtener una mejor interpretación de los datos y sentar las bases para modelos predictivos.

3. Marco Conceptual

3.1 Sistemas de Información y Gestión de Datos

En la actualidad, los sistemas de información cumplen un rol fundamental dentro de las organizaciones, ya que permiten capturar, almacenar y procesar grandes volúmenes de datos generados por las actividades diarias. Estos sistemas facilitan el control de la información y sirven como base para el análisis y la toma de decisiones. Para que un sistema de información sea eficiente, es necesario contar con una estructura de datos adecuada y con herramientas que garanticen la integridad, seguridad y disponibilidad de la información.

El manejo correcto de los datos se ha convertido en un elemento clave en entornos digitales como las redes sociales, donde las interacciones de los usuarios generan información de manera constante. Por este motivo, el uso de tecnologías de bases de datos resulta esencial para organizar y aprovechar estos datos de forma eficiente.

3.2 Bases de Datos Relacionales

Las bases de datos relacionales permiten organizar la información en tablas relacionadas entre sí mediante claves primarias y foráneas. Este modelo ofrece una estructura clara y ordenada, lo que facilita el almacenamiento y la consulta de datos. Gracias a las restricciones de integridad, se puede garantizar la consistencia de la información y evitar errores como la duplicación o la pérdida de datos.

Este tipo de bases de datos es ampliamente utilizado en sistemas transaccionales, ya que permite registrar operaciones de manera segura y confiable. Además, el uso del lenguaje SQL facilita la manipulación de los datos y la creación de consultas complejas que permiten obtener información específica según las necesidades del sistema.

3.3 PostgreSQL como Sistema Gestor de Base de Datos

PostgreSQL es un sistema gestor de bases de datos relacional de código abierto que destaca por su estabilidad, rendimiento y soporte para características avanzadas. Es una herramienta ampliamente utilizada tanto en el ámbito académico como profesional, debido a su capacidad para manejar grandes volúmenes de información y garantizar la integridad de los datos.

Entre sus principales características se encuentra el soporte para transacciones, vistas, funciones, procedimientos almacenados y triggers, lo que permite implementar lógica de negocio directamente en la base de datos. Estas funcionalidades contribuyen a la automatización de procesos y a una mejor organización del sistema de información.

3.4 Programación PL/pgSQL

PL/pgSQL es el lenguaje de programación procedural de PostgreSQL, el cual permite crear funciones y procedimientos almacenados dentro de la base de datos. A través de este

lenguaje, es posible ejecutar operaciones más complejas que van más allá de las sentencias SQL básicas.

El uso de PL/pgSQL permite validar datos, realizar cálculos automáticos, generar auditorías y controlar el flujo de las operaciones. De esta manera, la base de datos no solo actúa como un repositorio de información, sino también como un componente activo en la lógica del sistema.

3.5 Triggers y Automatización de Procesos

Los triggers son mecanismos que se ejecutan automáticamente cuando ocurre un evento específico en una tabla, como una inserción, actualización o eliminación de datos. Estos permiten automatizar tareas que deben realizarse de forma inmediata, sin necesidad de intervención del usuario.

La automatización mediante triggers ayuda a mantener la consistencia de los datos y a reducir errores humanos, ya que ciertas reglas se aplican de manera automática. Su uso es común en sistemas donde se requiere un control estricto de las operaciones realizadas sobre la información.

3.6 Modelo ANSI/SPARC y Seguridad de la Información

El modelo ANSI/SPARC define una arquitectura de tres niveles para los sistemas de bases de datos: externo, conceptual e interno. Esta separación permite independizar la forma en que los datos se almacenan físicamente de la forma en que son vistos por los usuarios.

Gracias a este modelo, es posible implementar diferentes niveles de acceso a la información mediante roles y permisos, lo que contribuye a la seguridad del sistema. La correcta gestión de accesos es fundamental para proteger la información y garantizar que cada usuario solo pueda interactuar con los datos que le corresponden.

3.7 Análisis de Datos y Toma de Decisiones

El análisis de datos consiste en examinar la información almacenada con el objetivo de identificar patrones, tendencias y comportamientos relevantes. A diferencia de los sistemas transaccionales, que se enfocan en el registro diario de operaciones, los sistemas analíticos permiten estudiar la información desde una perspectiva histórica.

Este tipo de análisis es especialmente importante en entornos donde se generan grandes volúmenes de datos, ya que permite transformar datos en conocimiento útil para la toma de decisiones estratégicas.

3.8 Data Warehouse y Data Mart

Un Data Warehouse es un repositorio de datos orientado al análisis, diseñado para almacenar información histórica proveniente de distintas fuentes. Su principal objetivo es facilitar la consulta y el análisis de datos para apoyar la toma de decisiones.

Un Data Mart es una versión más específica de un Data Warehouse, ya que se enfoca en un área particular del análisis. Este tipo de estructura permite acceder de manera rápida y eficiente a información relevante, simplificando el proceso de análisis.

3.9 Modelo Dimensional y Esquema Estrella

El modelo dimensional organiza los datos de forma que resulten fáciles de analizar. Uno de los esquemas más utilizados es el esquema en estrella, el cual se compone de una tabla de hechos central y varias tablas de dimensiones.

La tabla de hechos almacena los datos principales del análisis, mientras que las dimensiones permiten observar esa información desde diferentes perspectivas, como el tiempo o el tipo de interacción. Este modelo mejora el rendimiento de las consultas y facilita la creación de reportes y visualizaciones.

3.10 Proceso ETL

El proceso ETL (Extract, Transform, Load) es fundamental para la construcción de sistemas analíticos. Este proceso permite extraer datos desde sistemas operacionales, transformarlos según las necesidades del análisis y cargarlos en estructuras analíticas como los Data Mart.

Durante la etapa de transformación, los datos son limpiados y organizados para garantizar su calidad y consistencia. Un proceso ETL bien diseñado asegura que la información utilizada para el análisis sea confiable y útil.

3.11 Bases de Datos NoSQL

Las bases de datos NoSQL surgen como una alternativa a los modelos relacionales tradicionales, especialmente cuando se requiere mayor flexibilidad en el manejo de los datos. Estas bases de datos permiten trabajar con información semiestructurada o no estructurada y se adaptan mejor a ciertos tipos de aplicaciones modernas.

A diferencia del modelo relacional, las bases de datos NoSQL no dependen estrictamente de tablas y relaciones fijas, lo que permite una mayor libertad en el diseño de la información.

3.12 MongoDB como Base de Datos Orientada a Documentos

MongoDB es una base de datos NoSQL orientada a documentos que almacena la información en formato JSON. Este enfoque permite representar datos relacionados dentro de un mismo documento, facilitando el acceso a la información sin necesidad de realizar múltiples uniones.

El uso de documentos embebidos resulta especialmente útil en escenarios donde se manejan relaciones de uno a muchos, mejorando el rendimiento y la eficiencia del sistema.

3.13 Minería de Datos y Análisis Predictivo

La minería de datos es un proceso que permite descubrir patrones y relaciones a partir de grandes volúmenes de información. A través del uso de algoritmos y técnicas estadísticas, es posible obtener conocimiento que no es evidente a simple vista.

El análisis predictivo utiliza estos patrones para realizar estimaciones sobre comportamientos futuros, apoyando la toma de decisiones basadas en datos históricos.

4. Unidad 1 – PostgreSQL

4.1 Diseño lógico y físico de la base de datos

La base de datos fue diseñada para simular el funcionamiento de una red social, donde los usuarios pueden registrarse, interactuar entre sí, publicar contenido, reaccionar, comentar, enviar mensajes y seguir páginas.

El modelo lógico está organizado en entidades principales como usuarios, publicaciones, páginas y reacciones, junto con tablas de apoyo que permiten clasificar y relacionar la información, como género, país, categorías, tipo de publicación y tipo de reacción. Este diseño permite mantener la información normalizada y evitar la duplicación de datos.

La tabla usuarios es el eje central del sistema, ya que se relaciona con varias funcionalidades como amistades, publicaciones, mensajes, sesiones y reacciones. Para representar las relaciones entre usuarios se utiliza la tabla amistades, que almacena la conexión entre dos usuarios junto con la fecha en que se realizó la amistad.

El sistema permite manejar distintos tipos de contenido multimedia mediante las tablas fotos y videos, las cuales están asociadas directamente a los usuarios. Asimismo, las interacciones se gestionan mediante tablas como comentarios_publicaciones, reacciones y mensajes, lo que permite registrar la actividad de los usuarios dentro de la plataforma.

En cuanto al diseño físico, se implementaron claves primarias en todas las tablas para garantizar la unicidad de los registros y claves foráneas para asegurar la integridad referencial entre las entidades. La base de datos fue implementada en PostgreSQL, utilizando tipos de datos adecuados como INT, VARCHAR y DATE, lo que facilita el almacenamiento y la consulta eficiente de la información.

4.2 Programación PL/pgSQL (funciones y triggers)

Para automatizar y controlar ciertos procesos dentro de la base de datos, se implementaron funciones y triggers utilizando PL/pgSQL. Estas estructuras permiten que la base de datos no solo almacene información, sino que también realice validaciones y acciones automáticas.

Las funciones se utilizaron principalmente para realizar consultas específicas y cálculos, como obtener información resumida de la actividad de los usuarios o validar ciertos datos antes de ser procesados por el sistema.

Por otro lado, los triggers permiten ejecutar acciones automáticamente ante eventos como inserciones o actualizaciones de datos. Por ejemplo, se pueden usar para registrar fechas automáticamente, validar relaciones entre tablas o controlar el correcto funcionamiento de procesos internos del sistema.

El uso de PL/pgSQL mejora la consistencia de los datos y reduce errores, ya que muchas reglas se controlan directamente desde la base de datos.

5. Unidad 2 – Data Mart, ETL y Power BI

5.1 Conexión al Data Mart

Para hacer el Data Mart, primero conectamos la base de datos normal con el esquema analítico usando **PostgreSQL FDW**. Esto nos permitió traer las tablas que necesitábamos sin tener que copiar todo a mano.

Con FDW creamos un servidor remoto y un esquema local (**fdw_red_social**) donde quedaron las tablas listas para analizarlas. Lo bueno es que los datos siempre están actualizados y separados de la base normal.

5.2 Modelo Dimensional

El Data Mart se armó con un **esquema estrella**, que básicamente es una tabla central (tabla de hechos) y varias tablas alrededor que la describen (dimensiones). Esto hace que analizar los datos sea mucho más fácil.

Dimensiones:

- **d_genero:** Para ver la actividad de los usuarios según su género.
- **d_tiempo:** Para analizar los datos según día, mes y año.
- **d_tipo_publicacion:** Para clasificar las publicaciones por tipo (texto, foto, video).
- **d_tipo_reacciones:** Para saber qué tipo de reacciones usan más los usuarios.

Tabla de hechos (**f_publicaciones**):

Aquí se guarda la cantidad de reacciones según cada combinación de género, tipo de publicación, tipo de reacción y fecha. Con esto podemos ver patrones y tendencias.

5.3 Proceso ETL

El ETL es básicamente llevar los datos de la base normal al Data Mart, pero limpio y ordenado.

Extracción

Sacamos los datos de las tablas principales usando FDW, como:

- usuarios, genero, publicaciones, tipo_publicacion, reacciones, tipo_reacciones, tiempo

Con **IMPORT FOREIGN SCHEMA** los trajimos al esquema local.

Transformación

Aquí limpiamos y organizamos todo:

- Elegimos solo los datos que sirven para el análisis.
- Quitamos lo que estaba repetido.
- Revisamos que las fechas fueran correctas.
- Creamos tablas limpias para las dimensiones.
- Agregamos la info para la tabla de hechos (cuántas reacciones por género, tipo de publicación, tipo de reacción y día).

Carga

Por último, cargamos todo al Data Mart:

- Las **dimensiones** se llenaron con INSERT desde las tablas importadas.
- La **tabla de hechos** se llenó con los datos agregados del paso anterior.

Así quedó listo para analizar y usar en dashboards.

5.4 Visualizaciones y KPIs (Power BI)

Después usamos Power BI para hacer dashboards interactivos. Ahí se puede ver:

- Reacciones por tipo
- Actividad de los usuarios por género

- Publicaciones a lo largo del tiempo
- Comparaciones entre tipos de publicaciones y reacciones

Con estos KPIs podemos entender mejor el comportamiento de los usuarios y sacar conclusiones claras.

6.Unidad 3 – MongoDB y Weka

En esta unidad se trabajó con MongoDB y Weka como complemento al sistema desarrollado previamente en PostgreSQL y el Data Mart. El objetivo principal fue migrar parte de la información de la red social a un entorno NoSQL y aplicar técnicas básicas de minería de datos para analizar y predecir el comportamiento de los usuarios.

Colecciones

Los datos de la red social fueron migrados desde PostgreSQL hacia MongoDB, organizándolos en colecciones que representan las entidades principales del sistema. Se crearon colecciones para usuarios, publicaciones, comentarios y reacciones, utilizando el modelo orientado a documentos.

Este enfoque permitió manejar la información de forma más flexible y facilitar el acceso a datos relacionados sin necesidad de realizar joins complejos, como ocurre en bases de datos relacionales.

Consultas

Sobre las colecciones creadas se implementaron consultas avanzadas utilizando el framework de agregación de MongoDB. Estas consultas permitieron obtener información relevante como:

- Total de publicaciones por género.
- Total de publicaciones por tipo de publicación.
- Relación entre género, tipo de publicación y tipo de reacción.

Los resultados de estas consultas se utilizaron para generar datasets estructurados que luego fueron exportados para su análisis en Weka.

Modelos Predictivos

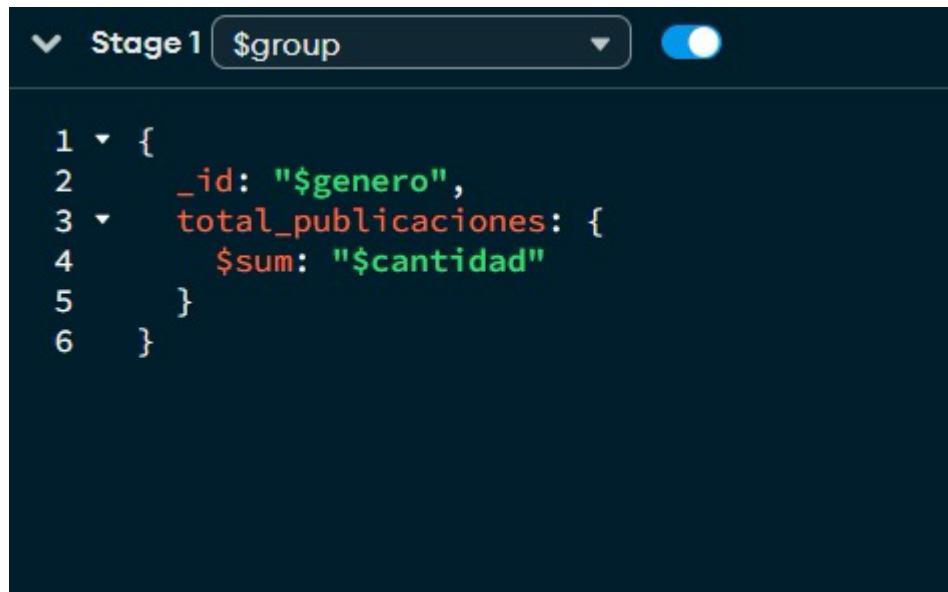
Una vez obtenidos los datasets, estos fueron exportados en formatos compatibles con Weka (CSV/ARFF). En esta herramienta se aplicaron modelos de clasificación, como Naive Bayes, para realizar predicciones relacionadas con el tipo de publicación y las interacciones generadas.

Además, se utilizaron técnicas de clustering para agrupar usuarios y publicaciones con comportamientos similares. Los resultados obtenidos permitieron identificar patrones de interacción y apoyar la toma de decisiones dentro del contexto de la red social.

```
_id: ObjectId('693009a26cb67240254499b4')
id_publicacion : 1
cantidad : 1
id_genero : 1
genero : "Masculino"
id_tipo : 1
tipo_publicacion : "Texto"
id_tipo_reaccion : 1
tipo_reaccion : "Me gusta"
id_t : 2
anio : 2024
mes : 1
dia : 11
```

8.Consultas Avanzadas Requeridas

A) Total de publicaciones por género



The screenshot shows the MongoDB aggregation pipeline interface. Stage 1 is selected, and the operation is set to '\$group'. The pipeline stage is defined as:

```
1 ▼ {
2   _id: "$genero",
3   total_publicaciones: {
4     $sum: "$cantidad"
5   }
6 }
```

```
▼ Stage 2 $sort ▾
1 ▼ {
2     total_publicaciones: -1
3 }
```

B) Total de publicaciones por tipo de publicación

```
▼ Stage 1 $group ▾
1 ▼ {
2     _id: "$tipo_publicacion",
3     total_publicaciones: {
4         $sum: "$cantidad"
5     }
6 }
```

Stage 2 \$sort

```
1 ▼ {  
2     total_publicaciones: -1  
3 }
```

ALL RESULTS

```
_id: "Texto"  
total_publicaciones : 53
```

```
_id: "Foto"  
total_publicaciones : 26
```

```
_id: "Video"  
total_publicaciones : 14
```

```
_id: "Enlace"  
total_publicaciones : 12
```

Dataset : Publicaciones por género ,tipo de publicación y tipo de reacción

▼ Stage 1 \$group

```
1 ▶ {  
2 ▶   _id: {  
3       genero: "$genero",  
4       tipo_publicacion: "$tipo_publicacion",  
5       tipo_reaccion: "$tipo_reaccion"  
6     },  
7 ▶   total_publicaciones: {  
8       $sum: "$cantidad"  
9     }  
10    }
```

▼ Stage 2 \$project

```
1 ▶ {  
2   _id: 0,  
3   genero: "$_id.genero",  
4   tipo_publicacion: "$_id.tipo_publicacion",  
5   tipo_reaccion: "$_id.tipo_reaccion",  
6   total_publicaciones: 1  
7 }
```

▼ Stage 3 \$sort

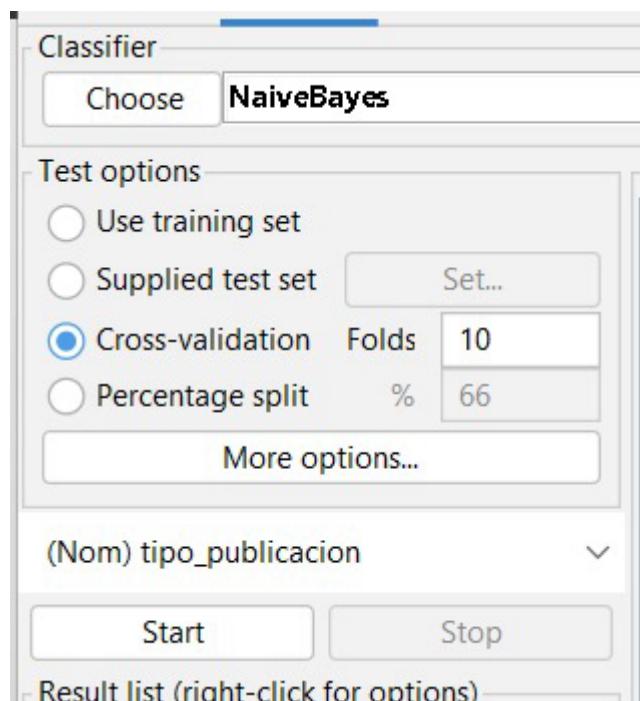
```
1 ▶ {  
2   total_publicaciones: -1  
3 }
```

Ejemplo :

```
{  
    "total_publicaciones": 2,  
    "genero": "Femenino",  
    "tipo_publicacion": "Foto",  
    "tipo_reaccion": "Me\\nencanta"  
}
```

9. Modelo de weka

- NaiveBayes : predicciones de tipo de una publicación



```

Classifier output
  _____
  tipo_reaccion
Test mode: 10-fold cross-validation

== Classifier model (full training set) ==

Naive Bayes Classifier

      Class
Attribute    Me gusta  Me\nencanta  Me entristece  Me asombra  Me divierte
              (0.27)     (0.32)      (0.14)      (0.14)      (0.14)
=====
total_publicaciones
  mean          2          1.5          1          1          1
  std. dev.    1.2649      0.5        0.1667      0.1667      0.1667
  weight sum    5          6          2          2          2
  precision     1          1          1          1          1

genero
  Masculino    3.0        4.0        2.0        2.0        1.0
  Femenino     3.0        3.0        2.0        2.0        3.0
  No\nbinario   2.0        2.0        1.0        1.0        1.0
  [total]       8.0        9.0        5.0        5.0        5.0

tipo_publicacion
  Texto         4.0        2.0        3.0        1.0        2.0
  Foto          3.0        3.0        1.0        1.0        2.0
  Video          1.0        4.0        1.0        1.0        1.0
  Enlace         1.0        1.0        1.0        3.0        1.0
  [total]        9.0       10.0       6.0       6.0       6.0

```

```

Time taken to build model: 0 seconds

==== Stratified cross-validation ====
==== Summary ===

Correctly Classified Instances          7           41.1765 %
Incorrectly Classified Instances       10          58.8235 %
Kappa statistic                         0.1943
Mean absolute error                     0.2681
Root mean squared error                 0.3804
Relative absolute error                 85.0966 %
Root relative squared error            94.6123 %
Total Number of Instances               17
Ignored Class Unknown Instances        6

==== Detailed Accuracy By Class ====

      TP Rate   FP Rate   Precision   Recall    F-Measure   MCC     ROC Area   PRC Area   Class
      0.400     0.167     0.500     0.400     0.444     0.251     0.372     0.258   Me gusta
      0.667     0.364     0.500     0.667     0.571     0.290     0.784     0.580   Me\nencanta
      0.000     0.133     0.000     0.000     0.000    -0.133     0.476     0.119   Me entristece
      0.500     0.000     1.000     0.500     0.667     0.685     1.000     1.000   Me asombra
      0.000     0.133     0.000     0.000     0.000    -0.133     0.560     0.132   Me divierte
Weighted Avg.      0.412     0.209     0.441     0.412     0.411     0.225     0.626     0.428

==== Confusion Matrix ====

a b c d e   <-- classified as
2 2 1 0 0 | a = Me gusta
2 4 0 0 0 | b = Me\nencanta
0 1 0 0 1 | c = Me entristece
0 0 0 1 1 | d = Me asombra
0 1 1 0 0 | e = Me divierte

```

10. Resultados Esperados

Se espera obtener una base de datos NoSQL en MongoDB correctamente estructurada y capaz de manejar de manera flexible la información de usuarios, publicaciones y reacciones. A través de las consultas avanzadas con pipelines de agregación, se obtendrán métricas claras como publicaciones por género, tipo de contenido y frecuencia de reacciones, generando datasets listos para su análisis en Weka.

También se espera entrenar modelos de clasificación que permitan predecir comportamientos, como el tipo de reacción o el nivel de interacción de una publicación, junto con modelos de clustering capaces de segmentar usuarios y agrupar publicaciones según su rendimiento.

Finalmente, el análisis permitirá identificar patrones significativos sobre la actividad de la red social, aportando información útil para comprender el comportamiento de los usuarios y mejorar las estrategias de contenido dentro del sistema