Final Presentacion

Olmo Baldoni, Nasser Chaouchi
922443@unizar.es, 922613@unizar.es

Mayo 2024

Contents

1 Introducion

El objetivo de este proyecto fue diseñar completamente los componentes de un Data Warehouse y analizarlo posteriormente utilizando técnicas de visualización. El trabajo se desarrolló en varias fases clave, cada una de las cuales contribuyó a la construcción de un sistema robusto y funcional para la gestión y análisis de datos cinematográficos, con un énfasis particular en las valoraciones de las películas.

La primera fase involucró el diseño del Data Warehouse utilizando la metodología de Kimball. Siguiendo los cuatro pasos fundamentales de esta metodología, construimos un data mart específico para el contexto cinematográfico. Este proceso incluyó la selección del proceso de negocio, la declaración del grano, la elección de las dimensiones y la identificación de los hechos.

Una vez definido el esquema, desarrollamos pipelines de datos para ejecutar el proceso ETL (Extract, Transform, Load). Este proceso implicó la extracción de datos de diversas fuentes, su transformación y la carga en el Data Warehouse. Durante esta fase, abordamos y resolvimos varios problemas de integración de datos, asegurando que la información fuera precisa y estuviera lista para su análisis.

En paralelo con la construcción del Data Warehouse, estudiamos e implementamos diferentes sistemas de recomendación, incluyendo el filtrado colaborativo y el filtrado basado en contenido. Estos sistemas se integraron luego en una sencilla interfaz gráfica, diseñada para sugerir nuevas películas a los usuarios en función de sus preferencias y comportamientos.

Finalmente, el Data Warehouse se utilizó para crear un dashboard interactivo para el análisis de películas. Este dashboard permite la visualización de diversas métricas y tendencias relacionadas con las valoraciones de las películas, ofreciendo una herramienta poderosa para el análisis en profundidad y la toma de decisiones informadas.

Este informe documenta detalladamente cada fase del proyecto, ilustrando las elecciones metodológicas, los desafíos enfrentados y los resultados obtenidos. La combinación de técnicas de Data Warehousing, procesos ETL y sistemas de recomendación ha permitido crear una solución completa para el análisis e interpretación de datos cinematográficos.

2 Proceso de Kimball

La metodología Kimball [1] facilita el diseño de almacenes de datos y sistemas de Business Intelligence. El método Kimball tiene un enfoque ascendente, es decir, se centra en las necesidades del negocio.
El uso de este método tiene varios aspectos clave:

- **Comprensión del negocio**: El almacén de datos permite satisfacer las necesidades críticas del negocio identificando y priorizando los procesos empresariales más importantes.
- **Flexibilidad y escalabilidad**: Utilizando el método de Kimball, el almacén de datos puede adaptarse con mayor facilidad, lo que le permite evolucionar más rápidamente en función de las necesidades del negocio y del sistema.
- **Usuarios**: El almacén de datos produce información relevante para los usuarios y satisface las necesidades de la empresa.

El método de Kimball consiste en seguir 4 pasos para modelar un almacén de datos :

1. Selecionar el proceso de negocio
2. Declarar el grano
3. Escoger las dimensiones
4. Identificar los hechos

2.1 Selecionar el proceso de negocio

La elección del proceso de negocio es el paso más importante, ya que nos permite conocer en profundidad la perspectiva empresarial del cliente y definir los procesos de negocio críticos que deben analizarse. Esta etapa es, por tanto, la base para la creación de nuestro almacén de datos, y todos nuestros KPI se derivarán de nuestra elección de proceso de negocio.

En nuestro caso de estudio, queremos analizar las valoraciones y las valoraciones medias de una película en función de los usuarios y los actores. Podríamos analizar otras métricas, pero en nuestro caso el proceso de negocio serán las valoraciones de una película. Este paso nos permite definir el grano que nos permitirá definir una línea desde un punto de vista atómico.

2.2 Declarar el grano

La granularidad se utiliza para definir el nivel de detalle adecuado para las mediciones en el almacén de datos. La granularidad corresponde al nivel de detalle de la tabla de hechos. Para definir el grano, determinamos los hechos físicos del proceso de negocio que van a ser almacenados por nuestro almacén de datos.
Así, en nuestro caso, cada fila de la tabla de hechos corresponderá a una calificación dada por un usuario a una película determinada en una fecha determinada.

2.3 Escoger las dimensiones

Las dimensiones proporcionan el contexto que rodea al evento del proceso de negocio. Nuestro objetivo es añadir un rico conjunto de dimensiones que representen el mayor número posible de características de los datos de la tabla de hechos. Son los elementos principales que definen las restricciones de una consulta, las agrupaciones y las etiquetas de un informe. Cada tabla de dimensiones tiene una única columna de clave primaria, que sirve como identificador único para cada fila de la dimensión. La clave primaria de la tabla de dimensiones se utiliza como clave ajena en la tabla de hechos asociada.
En el diseño del modelo dimensional para un sistema de calificación de películas, la atención se centra en capturar los detalles de las calificaciones de películas dadas por los usuarios, junto con el contexto en el que se realizan estas calificaciones. Esto incluye no sólo la valoración numérica en sí, sino también información sobre la película que se valora, el usuario que da la valoración y la fecha en que se dio la valoración.
El objetivo es crear un esquema en estrella que sirva de apoyo a las preguntas y al análisis de las valoraciones de las películas.
En la lógica relacionada con los géneros, las profesiones y los profesionales (a los que llamaremos X), tenemos las tablas `multiple_concatenate_X_key` y `multiple_concatenate_X_name` en la tabla `multiple_concatenate_X_dim_table`. El nombre es la concatenación de una combinación de X vista en las tablas fuente, de esta manera podemos asociar esta combinación con una persona o una película. Esta idea nos será útil más adelante, ya que utilizando otra tabla `multiple_X_dim_table` que contiene cada elemento `X_key` de un `multiple_concatenate_X_key` (separamos así los elementos), podremos seleccionar películas según sus géneros, o personas según sus profesiones, o simplemente personas, en PowerBi.

Dimensiones primarias

- **Dimensión Película (movie_dim_table):**
  - `movie_key` : Identificador único para cada película.
  - `title` : Nombre de la película.
  - `movieid` : Identificador de la película.
  - `tconst` : Identificador de la película en la base de datos externa.
  - `multiple_concatenate_genre_key` : Clave foránea de la tabla multiple_concatenate_genre_dim_table.
  - `multiple_concatenate_professional_key` : Clave foránea de la tabla multiple_concatenate_professional_dim_table.
  - `releaseDate` : Fecha de lanzamiento de la película.
  - `average_rating_imdb` : Calificación promedio en IMDB.
- **Dimensión Usuario (user_dim_table):**
  - `user_key` : Identificador único para cada usuario.
  - `useridml` : Identificador del usuario en MovieLens.
  - `user_counter` : Contador de usuarios.
- **Dimensión Fecha (date_dim_table):**
  - `date_key` : Identificador único para cada fecha.
  - `full_date` : Fecha completa.
  - `year` : Año.
  - `month` : Mes.
  - `day` : Día.
- **Dimensión Hora (time_dim_table):**
  - `time_key` : Identificador único para cada hora.
  - `time_value` : Valor de la hora.
  - `hours_24` : Hora en formato 24 horas.
  - `hours_12` : Hora en formato 12 horas.
  - `hour_minutes` : Minutos de la hora.
  - `day_minutes` : Minutos del día.
  - `seconds` : Segundos.

Dimensiones secundarias

- **Dimensión Profesión (profession_dim_table):**
  - `profession_key` : Identificador único para cada profesión.
  - `profession_name` : Nombre de la profesión.
- **Dimensión Profesión Múltiple (multiple_profession_dim_table):**
  - `multiple_concatenate_profession_key` : Clave foránea de la tabla multiple_concatenate_profession_dim_table.
  - `profession_key` : Clave foránea de la tabla profession_dim_table.
- **Dimensión Profesión Concatenada Múltiple (multiple_concatenate_profession_dim_table):**
  - `multiple_concatenate_profession_key` : Identificador único.
  - `multiple_concatenate_profession_name` : Nombre concatenado de la profesión.
- **Dimensión Profesional (professional_dim_table):**
  - `professional_key` : Identificador único.
  - `nconst` : Identificador del profesional en la base de datos externa.
  - `name` : Nombre del profesional.
  - `multiple_concatenate_profession_key` : Clave foránea de la tabla multiple_concatenate_profession_dim_table.
  - `birthyear` : Año de nacimiento.
  - `deathyear` : Año de fallecimiento.
- **Dimensión Profesional Múltiple (multiple_professional_dim_table):**
  - `multiple_concatenate_professional_key` : Clave foránea de la tabla multiple_concatenate_professional_dim_table.
  - `professional_key` : Clave foránea de la tabla professional_dim_table.
- **Dimensión Profesional Concatenada Múltiple (multiple_concatenate_professional_dim_table):**
  - `multiple_concatenate_professional_key` : Identificador único.
  - `multiple_concatenate_professional_name` : Nombre concatenado de los profesionales.
- **Dimensión Género (genre_dim_table):**
  - `genre_key` : Identificador único para cada género.
  - `genre_name` : Nombre del género.
- **Dimensión Género Múltiple (multiple_genre_dim_table):**
  - `multiple_concatenate_genre_key` : Clave foránea de la tabla multiple_concatenate_genre_dim_table.
  - `genre_key` : Clave foránea de la tabla genre_dim_table.

- **Dimensión Género Concatenada Múltiple (multiple_concatenate_genre_dim_table):**
  - `multiple_concatenate_genre_key` : Identificador único.
  - `multiple_concatenate_genre_name` : Nombre concatenado de los géneros.

2.4 Identificar los hechos

Los hechos son las métricas que resultan de un evento físico del proceso de negocio. Cada atributo de la tabla de hechos es un número natural distinto de cero. Todos estos hechos tienen un significado gracias a las tablas de dimensiones que dan contexto a los atributos de la tabla de hechos.
Una fila de una tabla de hechos tiene una relación de uno a uno con un evento de valoración descrito por el grano de la tabla de hechos. En este caso concreto, la métrica utilizada es *rating*, que representa las valoraciones dadas por los usuarios a las películas en una fecha determinada.

Tabla de Hechos de Valoraciones (rating_fact_table)

- `user_key` : Clave foránea que enlaza con la Dimensión Usuario (user_dim_table).
- `movie_key` : Clave foránea que enlaza con la Dimensión Película (movie_dim_table).
- `date_key` : Clave foránea que enlaza con la Dimensión Fecha (date_dim_table).
- `time_key` : Clave foránea que enlaza con la Dimensión Hora (time_dim_table).
- `ratings` : Valoración dada por el usuario a la película.

En la figura a continuación, se muestra el esquema E/R y el esquema en estrella que ilustran cómo las dimensiones se relacionan con la tabla de hechos para proporcionar el contexto necesario a las métricas registradas.

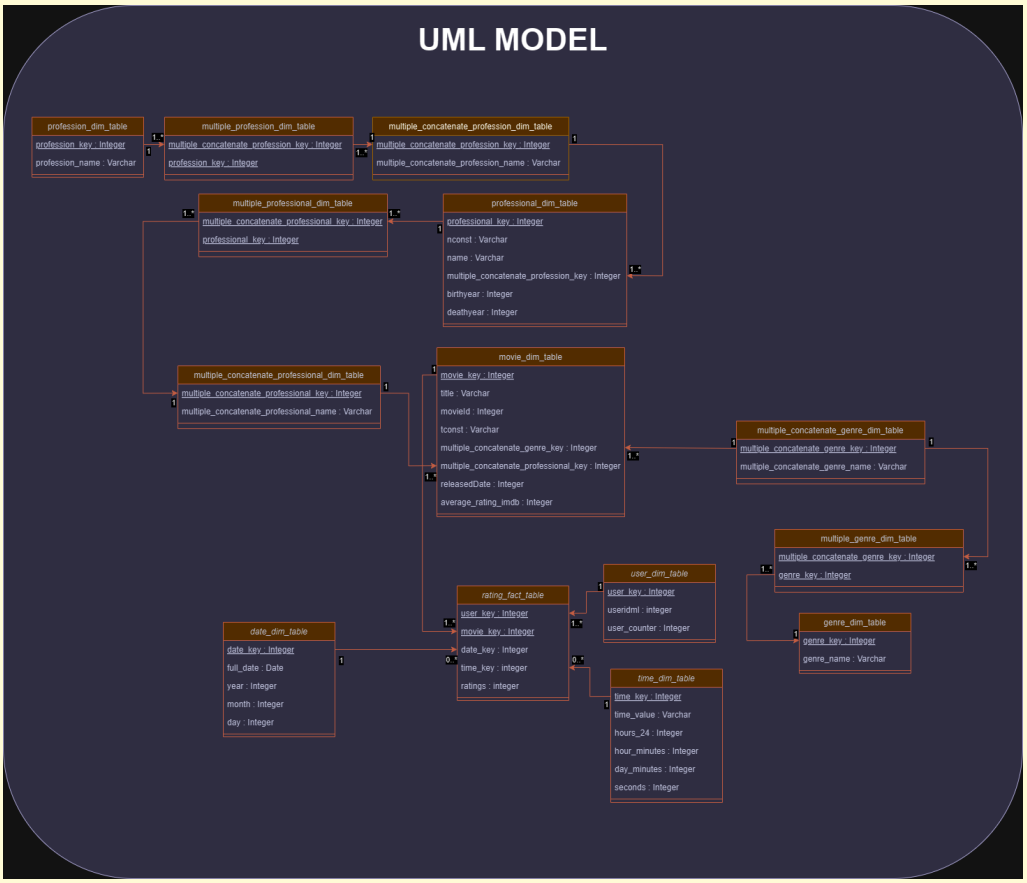

Figure 1: Esquema en estrella del modelo

Este diseño de la tabla de hechos permite un análisis detallado y flexible de las valoraciones de las películas, ofreciendo la posibilidad de cruzar la información con diversas dimensiones como usuarios, fechas y horas, para obtener insights más profundos y significativos sobre los patrones de valoración.

2.5 Conclusión de la sesión 1

En esta primera práctica, el objetivo era delinear una estructura de datos utilizando la técnica de los 4 pasos de Kimball. Comenzamos definiendo el proceso de negocio central, que en nuestro caso es la calificación de películas. Establecimos el grano al nivel de cada calificación individual, permitiendo un análisis detallado de las valoraciones.
El modelo de datos resultante se basa en una tabla de hechos, *rating_fact_table*, que captura cada valoración dada por un usuario a una película en una fecha y hora específicas. Las dimensiones asociadas proporcionan el contexto necesario para estas valoraciones, incluyendo información sobre usuarios, películas, fechas, horas, géneros y profesionales.
Con esta estructura, podemos realizar análisis profundos, como calcular puntuaciones medias por película, identificar tendencias temporales y evaluar la influencia de los profesionales y géneros en las valoraciones. Este modelo no solo facilita la creación de dashboards interactivos para el análisis de datos, sino que también sirve como base para sistemas de recomendación personalizados, optimizando así la experiencia del usuario. La combinación de estas capacidades ofrece una herramienta poderosa para entender y predecir patrones en la industria cinematográfica.

3 Proceso ETL

ETL (Extract, Transform, Load) es un proceso fundamental en la gestión de datos que se utiliza para recopilar datos de distintas fuentes, transformarlos si es necesario y cargarlos en una base de datos o almacén de datos para su posterior análisis.

- Extracción: durante esta fase, los datos se extraen de diversas fuentes, tanto estructuradas como no estructuradas, como bases de datos, archivos planos, API, sitios web, etc. El objetivo es recopilar todos los datos necesarios para el análisis.
- Transformación : Los datos extraídos se limpian, estructuran y transforman para satisfacer las necesidades de la empresa o el proyecto. Esto puede implicar la conversión de formatos de datos, la eliminación de duplicados, la normalización de valores, el cálculo de nuevas variables, etc. Esta etapa es crucial para garantizar la calidad y coherencia de los datos.
- Carga : Una vez extraídos y transformados los datos, se cargan en el destino final, como una base de datos relacional, un almacén de datos o un lago de datos. Esta etapa también puede implicar la creación de índices y otras estructuras para optimizar el rendimiento de la base de datos.

El proceso ETL es esencial en el ciclo de vida de la gestión de datos, ya que garantiza que los datos estén disponibles, sean fiables y estén listos para su uso en el análisis y la toma de decisiones. Espero que este documento le resulte útil como introducción a su trabajo práctico.

Para completar el proceso ETL, tenemos que pasar por varias etapas :

1. Elección del software dedicado al proceso ETL
2. Elección del servidor de base de datos
3. Elección del conjunto de datos
4. Extracción e inserción de datos en una base de datos
5. Programación de procedimientos SQL

Este es nuestro proceso ETL, que explicaremos en este informe:

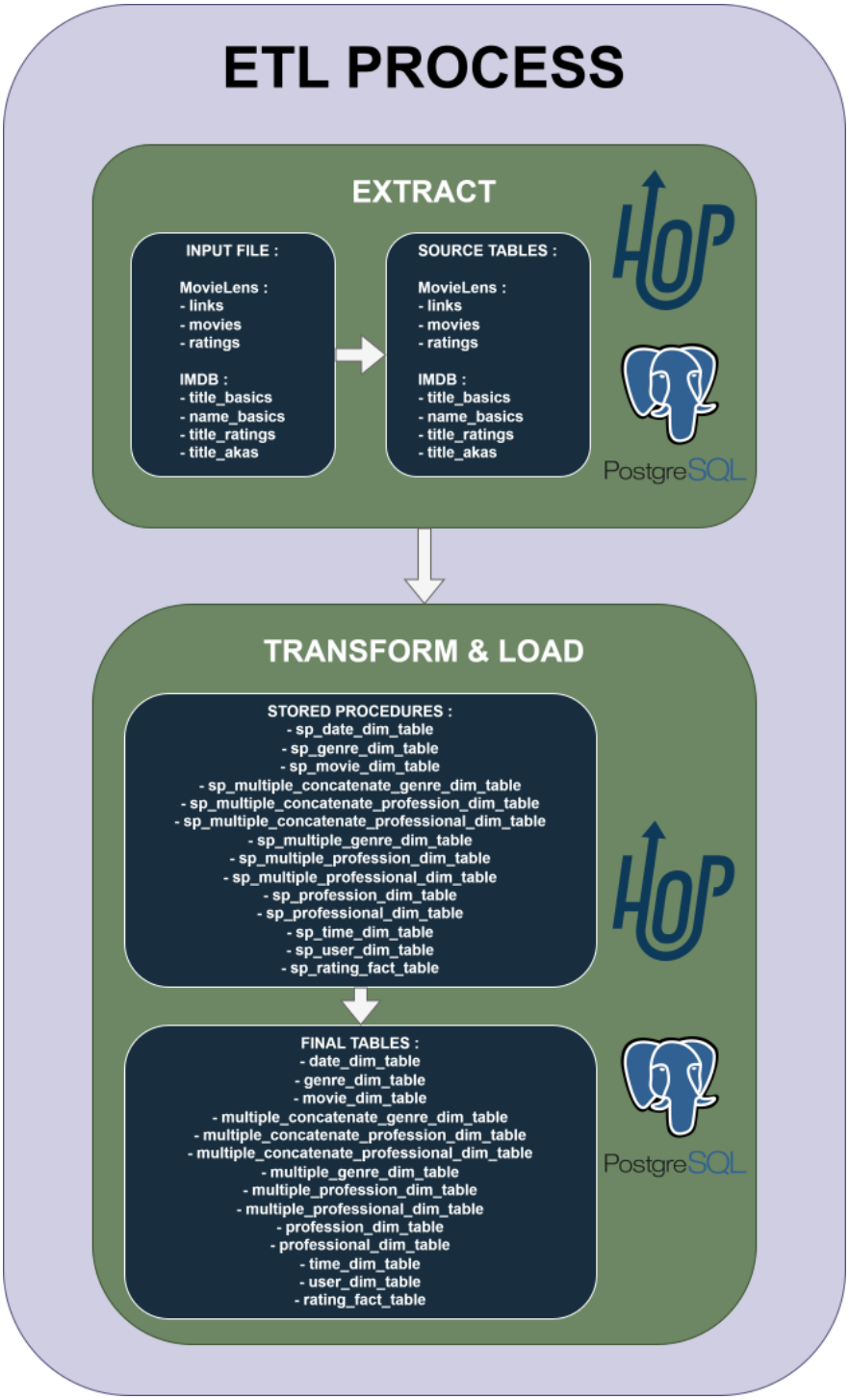

Figure 2: El proceso ETL

3.1 Elección del software dedicado al proceso ETL

El software que elegimos para el proceso ETL fue Apache HOP por varios factores clave. En primer lugar, su flexibilidad nos permite gestionar una amplia variedad de fuentes de datos, tipos de transformación y destinos finales, lo que se adapta perfectamente a nuestro complejo y cambiante entorno de datos. Además, la rica funcionalidad de Apache HOP proporciona multitud de herramientas y características para satisfacer nuestros requisitos específicos de limpieza, transformación y carga de datos.

En términos de rendimiento, Apache HOP es famoso por su capacidad para procesar grandes volúmenes de datos de forma eficiente, lo que resulta esencial para garantizar tiempos de respuesta rápidos en nuestros procesos ETL. Además, su arquitectura modular y extensible nos permite añadir fácilmente funciones personalizadas o conectores adicionales a medida que evolucionan nuestras necesidades.

Otra gran ventaja de Apache HOP es su modelo de código abierto, que nos da libertad para utilizar, modificar y distribuir el software sin las restricciones de costosas licencias.

Nuestra elección de Apache HOP para el proceso ETL se basa, por tanto, en su flexibilidad, gran funcionalidad, rendimiento y modelo de código abierto, que lo convierten en una solución ideal para nuestras necesidades de gestión de datos.

Figure 3: Logotipo de Apache hop

3.2 Elección del servidor de base de datos

Nuestra decisión de elegir PostgreSQL como base de datos para nuestro proyecto se debió a varios factores clave. En primer lugar, PostgreSQL tiene fama de fiable y estable. Su uso nos da confianza en su capacidad para gestionar nuestros datos de forma segura e impecable.

Además, PostgreSQL ofrece un rendimiento excepcional, incluso con grandes volúmenes de datos. Su motor de base de datos optimizado garantiza consultas rápidas y operaciones eficientes de lectura y escritura.

Otra razón clave por la que optamos por PostgreSQL es su riguroso cumplimiento de las normas SQL. Este cumplimiento facilita enormemente la integración con otros sistemas y herramientas, algo crucial en nuestro proceso ETL, en el que los datos pueden proceder de distintas fuentes y utilizarse en diferentes aplicaciones.

En general, nuestra elección de PostgreSQL se basó en su reputación de fiabilidad, rendimiento

probado, compatibilidad con los estándares SQL y gran funcionalidad, lo que lo convierte en un sistema perfecto para nuestro proceso ETL y nuestras necesidades de gestión de datos.

Figure 4: Logotipo de PostgreSQL

3.3 Elección del conjunto de datos

Para nuestra elección de los conjuntos de datos, optamos por los propuestos en el resumen del proyecto:

- IMDB
- MovieLens

Optamos por utilizar estos dos conjuntos de datos porque nos proporcionan las calificaciones de las películas, así como los nombres de los actores, productores y otros miembros del equipo técnico.

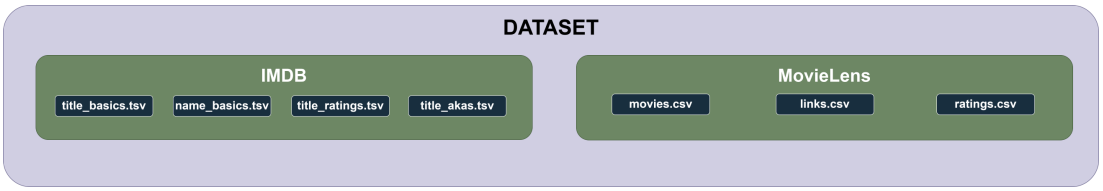

Figure 5: Elección de las tablas de origen

3.3.1 IMDB

Para el conjunto de datos IMDB, hemos optado por conservar 4 de los 7 archivos tsv para nuestro conjunto de datos.

title.akas.tsv.gz

- **titleId** (string) - a tconst, an alphanumeric unique identifier of the title
- **ordering** (integer) – a number to uniquely identify rows for a given titleId
- **title** (string) – the localized title

- **region** (string) - the region for this version of the title
- **language** (string) - the language of the title
- **types** (array) - Enumerated set of attributes for this alternative title. One or more of the following: "alternative", "dvd", "festival", "tv", "video", "working", "original", "imdbDisplay". New values may be added in the future without warning
- **attributes** (array) - Additional terms to describe this alternative title, not enumerated
- **isOriginalTitle** (boolean) – 0: not original title; 1: original title

title.basics.tsv.gz

- **tconst** (string) - alphanumeric unique identifier of the title
- **titleType** (string) – the type/format of the title (e.g. movie, short, tvseries, tvepisode, video, etc)
- **primaryTitle** (string) – the more popular title / the title used by the filmmakers on promotional materials at the point of release
- **originalTitle** (string) - original title, in the original language
- **isAdult** (boolean) - 0: non-adult title; 1: adult title
- **startYear** (YYYY) – represents the release year of a title. In the case of TV Series, it is the series start year
- **endYear** (YYYY) – TV Series end year. '\N' for all other title types
- **runtimeMinutes** – primary runtime of the title, in minutes
- **genres** (string array) – includes up to three genres associated with the title

title.ratings.tsv.gz

- **tconst** (string) - alphanumeric unique identifier of the title
- **averageRating** – weighted average of all the individual user ratings
- **numVotes** - number of votes the title has received

name.basics.tsv.gz

- **nconst** (string) - alphanumeric unique identifier of the name/person
- **primaryName** (string)– name by which the person is most often credited
- **birthYear** – in YYYY format
- **deathYear** – in YYYY format if applicable, else '\N'
- **primaryProfession** (array of strings)– the top-3 professions of the person
- **knownForTitles** (array of tconsts) – titles the person is known for

3.3.2 MovieLens

Table : movies

- **movieId** (integer)
- **title** (string)
- **genres** (string array)

Table : links

- **movieId** (integer)
- **imdbId** (integer)
- **tmdbId** (integer)

Table : rating

- **userId** (integer)
- **movieId** (integer)
- **rating** (float)
- **timestamp** (integer)

3.4 Extracción e inserción de datos en una base de datos

Para extraer los datos, utilizamos los widgets incorporados de Apache Hop, usando el "CSV Input File". Elegimos el delimitador adecuado: un tabulador para IMDB y una coma para MovieLens.

Para asegurar una correcta extracción e inserción de los datos sin errores, creamos workflows dedicados para cada tabla fuente. Estos workflows están diseñados para leer los archivos y realizar la inserción en la base de datos de manera eficiente, minimizando la posibilidad de errores.

Aquí hay un ejemplo para el archivo title_ratings:

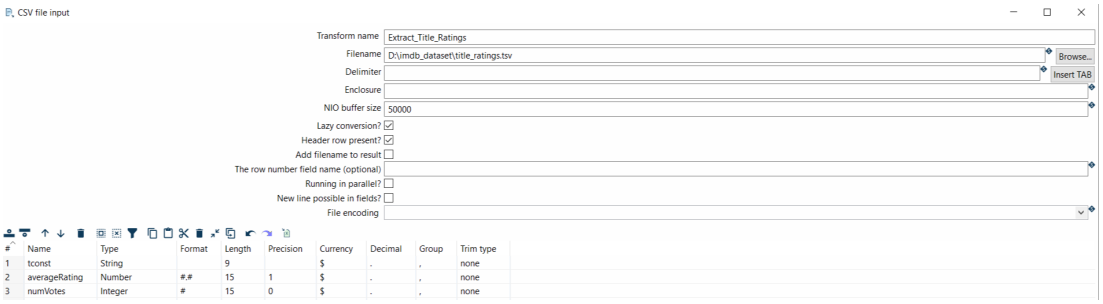

Figure 6: Parámetros de extracción del fichero title_ratings

Para insertar los datos de los archivos, conectamos nuestro Apache Hop a nuestro servidor PostgreSQL. Una vez realizada la conexión, utilizamos el widget "Insertar o actualizar" de Apache Hop y establecimos los parámetros de los campos y la tabla de destino deseada.

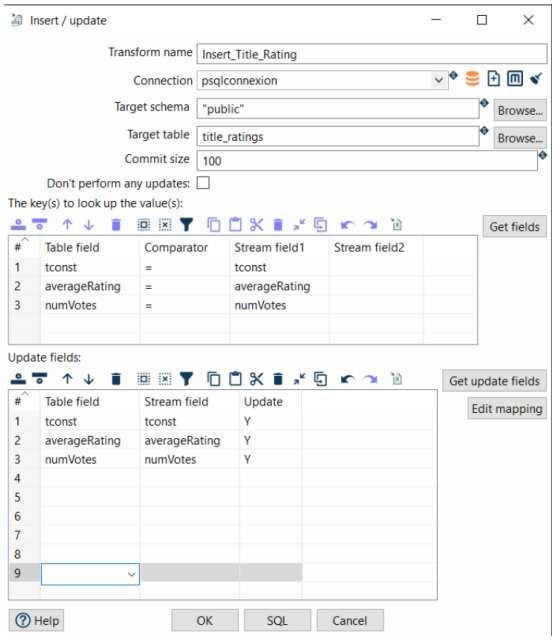

Figure 7: Parámetros de inserción del fichero title_ratings

Así que aquí está nuestra extracción e inserción de renderizado a través de la interfaz de Apache Hop:

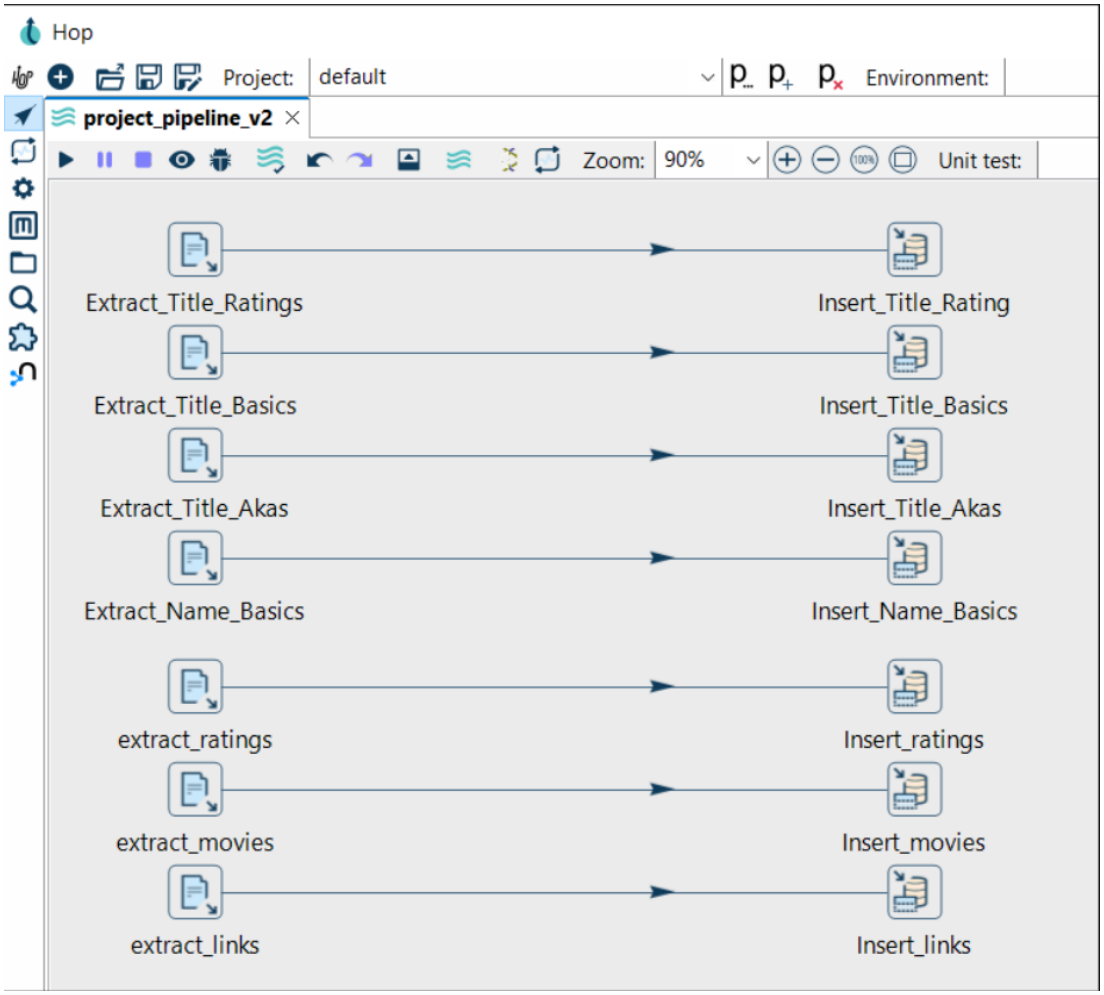

Figure 8: Extracción e inserción de renderizado a través de la interfaz de Apache Hop

3.5 Programación de procedimientos SQL

Para transformar y cargar nuestros datos para obtener nuestro nuevo modelo de datos utilizando nuestras tablas de origen, hemos decidido utilizar procedimientos SQL que llamarán a un script para seleccionar, modificar e insertar los datos de la tabla en otro formato. Este formato será el adecuado para analizar los datos posteriormente en PowerBI.

Los procedimientos almacenados son una herramienta indispensable en un proceso ETL por su capacidad para encapsular lógica de transformación de datos compleja y repetitiva. Al escribir la lógica de transformación en procedimientos almacenados, podemos centralizar y estandarizar el proceso, lo que facilita su mantenimiento y gestión a largo plazo. Además, los procedimientos almacenados pueden mejorar significativamente el rendimiento del proceso ETL al reducir la necesidad de transferir grandes volúmenes de datos entre el almacén de datos y otras capas del sistema. Esto se debe a que los procedimientos pueden ejecutarse directamente en la base de datos, lo que minimiza el tráfico de red y optimiza el tiempo de procesamiento.

Además, el uso de procedimientos almacenados proporciona un nivel adicional de seguridad y control sobre el proceso ETL. Al definir permisos específicos para ejecutar procedimientos almacenados, podemos limitar el acceso a los datos y garantizar que sólo los usuarios autorizados puedan modificar o manipular la información. Esto es especialmente importante en entornos corporativos en los que la seguridad de los datos es una prioridad. Además, los procedimientos almacenados proporcionan un nivel adicional de auditabilidad y trazabilidad, ya que cada paso del proceso ETL queda registrado en el registro de la base de datos, lo que facilita la resolución de problemas y la identificación de posibles problemas de rendimiento o integridad de los datos.

Haga clic aquí para ver la lógica simplificada de nuestros procedimientos sin tener en cuenta las concatenaciones y separaciones, sino sólo las tablas utilizadas, así como JOIN y UNION para transformar y cargar con éxito los datos en nuestras tablas finales.

Por último, sólo tenemos que ejecutar un script SQL en Apache Hop, llamando a los distintos procedimientos en este orden:

1. sp_time_dim_table
2. sp_date_dim_table
3. sp_user_dim_table
4. sp_profession_dim_table
5. sp_multiple_concatenate_profession_dim_table
6. sp_multiple_profession_dim_table
7. sp_genre_dim_table
8. sp_multiple_concatenate_genre_dim_table
9. sp_multiple_genre_dim_table
10. sp_multiple_concatenate_professional_dim_table
11. sp_multiple_professional_dim_table
12. sp_professional_dim_table
13. sp_movie_dim_table
14. sp_rating_fact_table

3.6 Conclusión de la sesion 2

El proceso ETL (Extract, Transform, Load) desempeña un papel esencial en la gestión de datos, ya que recopila datos de diversas fuentes, los transforma cuando es necesario y los carga en una base de datos o almacén de datos para su posterior análisis.

Este proceso es crucial en el ciclo de vida de la gestión de datos, ya que garantiza que los datos estén disponibles, sean fiables y estén listos para ser utilizados en el análisis y la toma de decisiones. Por tanto, la elección de las herramientas y tecnologías adecuadas es esencial para asegurar la eficacia y fiabilidad del proceso ETL.

En nuestro informe, hemos examinado en detalle las distintas fases del proceso ETL, incluyendo la elección del software dedicado, el servidor de bases de datos y los conjuntos de datos. También hemos explorado los métodos de extracción e inserción de datos, así como la programación de procedimientos SQL para transformar y cargar los datos en un formato adecuado para nuestros futuros análisis.

En conclusión, el proceso ETL es un elemento fundamental de la gestión de datos, y su eficacia depende de una combinación de herramientas adecuadas, tecnologías fiables y buenas prácticas de desarrollo para analizar los datos con mayor tranquilidad.

4 Sistemas de Recomendación

Elegir una película puede ser una experiencia desalentadora y laboriosa. Con la enorme variedad de películas disponibles, puede resultar abrumador decidirse por un solo título. Este problema se ve agravado por la enorme cantidad de contenidos disponibles, que hace cada vez más difícil para los usuarios descubrir películas nuevas y relevantes.

En los últimos años se ha reconocido cada vez más la importancia de las recomendaciones personalizadas de contenidos como factor clave para mejorar la experiencia del usuario. Al ofrecer a los usuarios recomendaciones personalizadas, los proveedores de contenidos pueden aumentar la participación de los usuarios, reducir el tiempo dedicado a la búsqueda de contenidos y, en última instancia, mejorar la experiencia general del usuario.

En esta sesión, evaluaremos la eficacia de varios enfoques de recomendación de películas utilizando dos conjuntos de datos distintos. Analizaremos el rendimiento de los enfoques de filtrado colaborativo y basado en el contenido utilizando los conjuntos de datos MovieLens y TMDB. Además, se ha desarrollado una pequeña interfaz que permite probar de forma práctica tanto los enfoques colaborativos como los basados en el contenido, ofreciendo a los usuarios la oportunidad de interactuar directamente con los sistemas de recomendación.

4.1 Tipos de sistemas de recomendación

Los sistemas de recomendación pueden clasificarse en tres tipos principales: Filtrado colaborativo, Filtrado basado en contenidos y Enfoques híbridos. Cada tipo tiene sus propios puntos fuertes y débiles, que se analizarán a seguir:

4.1.1 Filtrado colaborativo

El filtrado colaborativo es un tipo de sistema de recomendación que se basa en el comportamiento de usuarios similares para hacer predicciones sobre las preferencias de un usuario. Este enfoque se basa en la idea de que a los usuarios con gustos y preferencias similares también les gustan los mismos artículos. El filtrado colaborativo puede dividirse a su vez en dos subcategorías:

- Filtrado colaborativo basado en el usuario: este enfoque recomienda artículos basándose en el comportamiento de usuarios similares.
- Filtrado colaborativo basado en el artículo Este enfoque recomienda artículos similares a los que han gustado a un usuario o con los que ha interactuado en el pasado.

Este método es especialmente eficaz para descubrir nuevos intereses que los usuarios pueden no haber considerado, ya que no se basa en el conocimiento del contenido en sí, sino en las interacciones de los usuarios. Sin embargo, este enfoque presenta retos importantes: requiere una gran cantidad de datos para funcionar eficazmente, lo que provoca el problema del arranque en frío, es decir, la dificultad de hacer recomendaciones para nuevos usuarios o nuevos artículos. Además, es sensible a la escasez de datos y tiende a favorecer los artículos populares en detrimento de los menos conocidos.

4.1.2 Filtrado por contenido

El filtrado basado en el contenido es un tipo de sistema de recomendación que recomienda artículos en función de sus atributos o características. Este enfoque se basa en la idea de que

a los usuarios les gustan los artículos que tienen atributos similares a los que les han gustado en el pasado.

Este método ofrece recomendaciones muy personalizadas y no requiere información sobre otros usuarios, por lo que evita el problema del arranque en frío. Sin embargo, puede limitar a los usuarios a una burbuja, recomendando sólo artículos similares a los ya explorados. Además, la eficacia de este sistema depende de la capacidad de analizar con precisión el contenido, lo que puede resultar complejo para determinados tipos de datos o contenidos.

4.1.3 Filtrado híbrido

Los enfoques híbridos combinan los puntos fuertes del filtrado colaborativo y del filtrado basado en el contenido, integrando ambos métodos. Este enfoque puede aprovechar los puntos fuertes de ambos métodos y mitigar sus puntos débiles. Sin embargo, su complejidad de aplicación es mayor y requiere una calibración y gestión cuidadosas para explotar todo el potencial de ambas metodologías. La flexibilidad que ofrece este enfoque híbrido puede dar lugar a soluciones más robustas y versátiles, aunque a costa de una mayor complejidad sistémica.

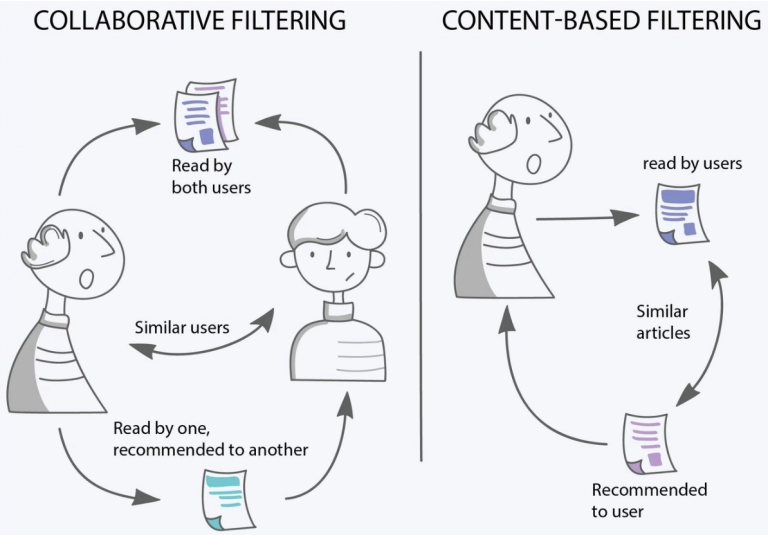

Figure 9: Gráfico que muestra la diferencia entre el filtrado colaborativo y el basado en el contenido.

4.2 Conjuntos de datos utilizados

El conjunto de datos MovieLens es una referencia muy utilizada para evaluar algoritmos de filtrado colaborativo. Se recopiló a partir del sitio web MovieLens, que permite a los usuarios valorar películas en una escala de 1 a 5. El conjunto de datos contiene más de 100.000 valoraciones de más de 600 usuarios sobre más de 9.000 películas. El conjunto de datos contiene más de 100.000 valoraciones de más de 600 usuarios sobre más de 9.000 películas. El conjunto de datos es especialmente adecuado para el filtrado colaborativo porque contiene un gran número de valoraciones de un conjunto heterogéneo de usuarios, lo que permite una evaluación sólida de la capacidad del algoritmo para ofrecer recomendaciones personalizadas.

El conjunto de datos TMDB, por su parte, es un completo conjunto de datos de películas y programas de televisión que contiene información como el título, el género, el director, el

argumento y el reparto. Este conjunto de datos es especialmente adecuado para el filtrado basado en el contenido, ya que contiene una gran cantidad de metadatos sobre las películas y los programas de televisión, que pueden utilizarse para generar recomendaciones basadas en los atributos de los elementos.

La elección de utilizar tanto MovieLens como TMDB para este estudio se debe a la complementariedad de ambos conjuntos de datos a la hora de tratar distintos aspectos de los sistemas de recomendación. MovieLens, con su abundante colección de valoraciones y comentarios de los usuarios, es ideal para probar y perfeccionar algoritmos de filtrado colaborativo que se basan en las interacciones y similitudes de los usuarios para predecir preferencias. Por otro lado, TMDB ofrece un vasto conjunto de datos descriptivos que pueden utilizarse para implantar y optimizar sistemas de recomendación basados en contenidos que requieren un análisis detallado de las características intrínsecas de las películas.

4.3 Experimentos y resultados

Como parte del enfoque colaborativo, el conjunto de datos MovieLens Small se procesó mínimamente, puesto que ya era apto para su uso. Los archivos CSV `movies.csv` y `ratings.csv` se integraron para formar un único conjunto de datos que contenía los atributos `userId`, `title` y `rating`.

Para el análisis se utilizó la biblioteca `surprise`, basada en scikit-learn y especializada en la creación y el análisis de sistemas de recomendación que manejan datos de clasificación explícitos. Este conjunto de herramientas ofrece diversos algoritmos de predicción listos para usar, incluidos métodos basados en k-Nearest Neighbours (kNNBasic, kNNWithZScore, KNNWithMeans), métodos de factorización matricial (SVD, NMF, etc.), así como herramientas para evaluar, analizar y comparar el rendimiento de los algoritmos. Los procedimientos de validación cruzada se facilitaron mediante iteradores CV inspirados en scikit-learn, lo que permitió realizar evaluaciones rigurosas y estructuradas.

La optimización de los hiperparámetros se llevó a cabo mediante el método GridSearchCV, un enfoque de búsqueda en cuadrícula que explora un subconjunto especificado manualmente de los hiperparámetros de un algoritmo de aprendizaje para identificar la configuración de entrenamiento óptima.

Las métricas de evaluación adoptadas para este estudio fueron RMSE (Root Mean Square Error) y MAE (Mean Absolute Error). De los resultados, presentados en la tabla de referencia, se desprende que el mejor rendimiento se obtuvo utilizando el método KNNBasic para los métodos basados en KNN, y también se registraron mejoras significativas con el uso de NMF (factorización de matrices no negativas) y CoClustering.

En el enfoque de filtrado de contenidos, se construyó un atributo agregado denominado ‘corpus‘ dentro del conjunto de datos TMDB, mediante la unión de la información de los atributos `overview`, `genre`, `cast` y `crew`. Este procedimiento permitió enriquecer el contexto con los datos disponibles. La estructura final del conjunto de datos incluía los atributos `movieId`, `title` y `corpus`.

Mediante la metodología TF-IDF, se cuantificó la importancia de las palabras del `corpus` en función de su frecuencia en el documento y su rareza en el corpus de documentos. En un sistema de recomendación, esta técnica transforma los atributos distintivos de las películas en un vector numérico que facilita la comparación entre entidades. Posteriormente, es posible calcular la similitud coseno entre estos vectores para identificar películas con características similares. Un ejemplo de este mecanismo puede verse en la salida producida por el sistema, donde se sugieren películas en función de sus similitudes con la película de referencia `Die Another Day`, destacando `Ojo de oro` como el título más relevante en función de la puntuación de similitud más alta.+

4.4 Creación de una interfaz gráfica de usuario para la recomendación de películas

Al diseñar la interfaz de usuario de nuestro sistema de recomendación de películas, el objetivo principal era ofrecer una experiencia intuitiva y sencilla a los usuarios. La interfaz se estructuró para guiar a los usuarios a través de dos flujos principales: uno para el filtrado colaborativo y otro para el filtrado basado en contenidos, ambos centralizados en una única plataforma web. Mediante formularios y selecciones guiadas, el usuario puede interactuar fácilmente con el sistema, dar su opinión sobre las películas ya vistas y recibir recomendaciones pertinentes. Un aspecto clave fue el uso de Streamlit, una herramienta que permite convertir scripts de Python en aplicaciones web interactivas con relativa facilidad. En combinación con la biblioteca de gestión de datos pandas y un módulo `utils` personalizado, el sistema es capaz de gestionar y presentar los datos con eficacia.

Figure 10: Cuadro de mando para el filtrado colaborativo

Streamlit fue elegido por su capacidad para integrar funciones de entrada y visualización de datos, como cuadros de selección y deslizadores, que permiten a los usuarios indicar sus preferencias con sólo unos clics. El diseño de la interfaz utiliza elementos gráficos como títulos y botones claramente identificables para separar visual y funcionalmente los dos modos de recomendación. Además, las interacciones de los usuarios van acompañadas de mensajes de respuesta y éxito, que confirman visualmente las acciones realizadas y mejoran la experiencia global.

Figure 11: Cuadro de mandos para el filtrado basado en contenidos

La interfaz permite a los usuarios seleccionar el tipo de algoritmo deseado para las recomendaciones colaborativas, añadir valoraciones para las películas vistas y ver las recomendaciones proporcionadas por el sistema. Para el filtrado de contenidos, los usuarios pueden seleccionar una película para recibir recomendaciones basadas en las características del contenido.

4.5 Conclusiones de la sesion 3

Este trabajo examinó la eficacia de varios enfoques para la recomendación de películas utilizando dos conjuntos de datos diferentes, MovieLens y TMDB. Los resultados mostraron que ambos enfoques, el filtrado colaborativo y el filtrado basado en el contenido, pueden ser eficaces a la hora de recomendar películas. El filtrado colaborativo resultó más eficaz a la hora de ofrecer recomendaciones personalizadas, mientras que el filtrado basado en el contenido fue más eficaz a la hora de ofrecer recomendaciones basadas en el contenido de las películas.

La creación de una interfaz de usuario para la recomendación de películas demostró ser una parte importante del proceso de recomendación, ya que permite a los usuarios interactuar directamente con el sistema y recibir recomendaciones personalizadas.

5 Creación de Vistas para PowerBI

Para facilitar el análisis de datos en PowerBI, hemos decidido crear vistas en lugar de utilizar directamente las tablas. Las vistas son una herramienta poderosa en SQL que permite crear una representación lógica de los datos sin duplicar físicamente la información almacenada en las tablas. A continuación, se presentan las vistas creadas y el código SQL asociado para cada una.

5.1 Código de la creación de las vistas

5.1.1 Vista para Dimensión de Tiempo

```
CREATE VIEW public.time_dim_view AS
SELECT
    time_key,
    time_value,
    hours_24,
    hours_12,
    hour_minutes,
    day_minutes,
    seconds
FROM
    public.time_dim_table;
```

5.1.2 Vista para Hechos de Calificación

```
CREATE VIEW public.rating_fact_view AS
SELECT
    user_key,
    movie_key,
    date_key,
    time_key,
    rating
FROM
    public.rating_fact_table;
```

5.1.3 Vista para Dimensión de Profesionales

```
CREATE VIEW public.professional_dim_view AS
SELECT
    professional_key,
    nconst,
    name,
    multiple_concatenate_profession_key,
    birthyear,
    deathyear
FROM
    public.professional_dim_table;
```

5.1.4 Vista para Profesionales Múltiples

```
CREATE VIEW public.multiple_professional_dim_view AS
SELECT
    multiple_concatenate_professional_key,
    professional_key
FROM
    public.multiple_professional_dim_table;
```

5.1.5 Vista para Profesiones Múltiples

```
CREATE VIEW public.multiple_profession_dim_view AS
SELECT
    multiple_concatenate_profession_key,
    profession_key
FROM
    public.multiple_profession_dim_table;
```

5.1.6 Vista para Géneros Múltiples

```
CREATE VIEW public.multiple_genre_dim_view AS
SELECT
    multiple_concatenate_genre_key,
    genre_key
FROM
    public.multiple_genre_dim_table;
```

5.1.7 Vista para Profesionales Concatenados Múltiples

```
CREATE VIEW public.multiple_concatenate_professional_dim_view AS
SELECT
    multiple_concatenate_professional_key,
    multiple_concatenate_professional_name
FROM
    public.multiple_concatenate_professional_dim_table;
```

5.1.8 Vista para Profesiones Concatenadas Múltiples

```
CREATE VIEW public.multiple_concatenate_profession_dim_view AS
SELECT
    multiple_concatenate_profession_key,
    multiple_concatenate_profession_name
FROM
    public.multiple_concatenate_profession_dim_table;
```

5.1.9 Vista para Dimensión de Género

```
CREATE VIEW public.genre_dim_view AS
SELECT
    genre_key,
    genre_name
```

```
FROM
    public.genre_dim_table;
```

5.1.10 Vista para Dimensión de Fecha

```
CREATE VIEW public.date_dim_view AS
SELECT
    date_key,
    full_date,
    year,
    month,
    day
FROM
    public.date_dim_table;
```

5.1.11 Vista para Dimensión de Profesión

```
CREATE VIEW public.profession_dim_view AS
SELECT
    profession_key,
    profession_name
FROM
    public.profession_dim_table;
```

5.1.12 Vista para Dimensión de Películas

```
CREATE VIEW public.movie_dim_view AS
SELECT
    m.movie_key,
    m.title,
    m.movieid,
    m.tconst,
    m.multiple_concatenate_genre_key,
    m.multiple_concatenate_professional_key,
    m.releasedate,
    m.averagerating,
    m.num_votes,
    t.runtimeminutes
FROM
    public.movie_dim_table m
LEFT JOIN
    public.title_basics t
ON
    m.tconst = t.tconst;
```

5.1.13 Vista para Agregados de Calificaciones de Películas

```
CREATE VIEW public.movie_rating_aggregates AS
SELECT
    movie_key,
```

```
    SUM(rating) AS total_ratings,
    COUNT(rating) AS number_of_ratings
FROM
    public.rating_fact_table
GROUP BY
    movie_key;
```

5.1.14 Vista para Dimensión de Películas con Calificaciones

```
CREATE VIEW public.movie_dim_with_ratings AS
SELECT
    m.movie_key,
    m.title,
    m.movieid,
    m.tconst,
    m.multiple_concatenate_genre_key,
    m.multiple_concatenate_professional_key,
    m.releasedate,
    m.averagerating,
    m.num_votes,
    r.total_ratings,
    r.number_of_ratings,
    (m.averagerating * m.num_votes + r.total_ratings * 2) /
    (m.num_votes + r.number_of_ratings) AS weighted_average_rating
FROM
    public.movie_dim_view m
LEFT JOIN
    public.movie_rating_aggregates r
ON
    m.movie_key = r.movie_key;
```

5.2 Razones para Utilizar Vistas

- **Simplicidad** : Las vistas pueden simplificar consultas complejas al encapsular la lógica de las consultas en una vista predefinida. Esto facilita la elaboración de informes y análisis en PowerBI.
- **Seguridad** : Las vistas permiten restringir el acceso a ciertas columnas o filas de datos, proporcionando un nivel adicional de seguridad.
- **Mantenimiento** : Las vistas pueden facilitar el mantenimiento y la gestión de cambios en la estructura de las bases de datos. Si se realizan cambios en las tablas subyacentes, sólo es necesario actualizar la vista en lugar de todas las consultas que la utilizan.
- **Rendimiento** : Las vistas pueden mejorar el rendimiento al permitir la reutilización de consultas optimizadas y simplificar la lógica de las consultas para el motor de base de datos.

5.3 Proceso de Creación de Vistas

Para crear las vistas necesarias para nuestro análisis en PowerBI, seguimos estos pasos:

1. **Definir los Requisitos de Información** : Identificamos las necesidades específicas de análisis e informes para determinar qué datos deben estar disponibles en las vistas.

2. **Escribir Consultas SQL** : Redactamos las consultas SQL que encapsulan la lógica de negocio necesaria para cada vista. Estas consultas se centran en las métricas y dimensiones clave definidas en nuestro esquema en estrella.

3. **Crear las Vistas en PostgreSQL** : Utilizamos las consultas SQL para crear las vistas en nuestra base de datos PostgreSQL. A continuación se presentan ejemplos de la creación de algunas de las vistas utilizadas.

4. **Validar las Vistas** : Probamos las vistas para asegurarnos de que proporcionan los datos correctos y cumplen con los requisitos de análisis definidos.

5. **Integrar con PowerBI** : Importamos las vistas creadas en PowerBI y las utilizamos para construir los informes y paneles interactivos necesarios para el análisis de datos.

5.4 Integración de Datos en PowerBI

Para realizar el análisis de datos en PowerBI, decidimos conectar nuestra base de datos PostgreSQL. Sin embargo, debido al tamaño de los datos y las limitaciones de nuestro sistema, optamos por exportar los datos a archivos CSV y subirlos a PowerBI. A continuación, se detallan los pasos que seguimos para realizar esta integración y la creación de medidas en PowerBI para análisis específicos.

5.4.1 Exportación de Datos a CSV

Dado que nuestro sistema no podía manejar la conexión directa a PostgreSQL debido al tamaño de los datos, exportamos las vistas creadas a archivos CSV. Este método nos permitió manejar los datos de manera más eficiente en PowerBI.

Pasos para la Exportación a CSV

1. **Crear Vistas en PostgreSQL**: Primero, creamos todas las vistas necesarias en nuestra base de datos PostgreSQL, como se describió en la sección anterior.

2. **Exportar Vistas a CSV**: Utilizamos comandos SQL y herramientas de PostgreSQL para exportar cada vista a un archivo CSV.

3. **Subir CSV a PowerBI**: Subimos los archivos CSV exportados a PowerBI para su análisis.

5.4.2 Creación de Medidas en PowerBI

Una vez que los datos fueron subidos a PowerBI, procedimos a crear varias medidas para facilitar el análisis de los datos. Estas medidas nos permitieron realizar cálculos específicos y visualizar los datos de manera más efectiva.

Medidas Creadas en PowerBI

- **Total Movies With Ratings**:

```
Total Movies With Ratings =
COUNTROWS(FILTER('movie_dim_view', 'movie_dim_view'[total_ratings] <> 0))
```

Descripción: Esta medida cuenta el número de películas en $movie_dim_viewdondetotal\_ratingsesdiferentede0$.

- **Average Weighted Rating With Ratings**:

```
Average Weighted Rating With Ratings =
CALCULATE(
    AVERAGE('movie_dim_with_ratings'[weighted_average_rating]),
    'movie_dim_with_ratings'[total_ratings] <> 0
)
```

Descripción: Esta medida calcula la media de *weighted_average_rating* para las películas donde *total_ratings* es diferente de 0 en *movie_dim_with_ratings*.

- **Total Ratings by Year**:

```
Total Ratings by Year = COUNT('rating_fact_table'[rating])
```

Descripción: Esta medida cuenta el número de calificaciones en *rating_fact_table* para cada año.

Columnas Calculadas

- **Year** (en *rating_fact_table*):

```
Year = RELATED('date_dim_table'[year])
```

Descripción: Esta columna calcula el año a partir de *date_dim_table* utilizando la relación con *date_key*.

- **Runtime Minutes as Integer** (en *movie_dim_view*):

```
Runtime Minutes as Integer =
IF(
    OR('movie_dim_view'[runtimeminutes] = "\N", 'movie_dim_view'[runtimeminutes] = ""),
    BLANK(),
    VALUE('movie_dim_view'[runtimeminutes])
)
```

Descripción: Esta columna convierte *runtimeminutes* a un valor entero, manejando valores nulos o vacíos adecuadamente.

5.4.3 Vistas en PowerBI

Creamos varias vistas en PowerBI para proporcionar diferentes perspectivas del análisis de datos, incluyendo:

- **Vista General**: Un tablero que muestra el promedio de todas las calificaciones de películas, el conteo de películas por año y otras métricas generales.
- **Vista IMDB**: Un tablero específico que se centra en las calificaciones de IMDB, mostrando métricas como el promedio de calificaciones en IMDB y la distribución de calificaciones.

- **Vista MovieLens**: Un tablero que se enfoca en las calificaciones de MovieLens, con métricas como el promedio de calificaciones en MovieLens y el número de calificaciones por película.

Estas vistas y medidas nos permitieron realizar un análisis exhaustivo de los datos y obtener insights valiosos sobre las calificaciones de las películas, tanto en general como en plataformas específicas como IMDB y MovieLens.

5.5 Descripción de las Vistas en PowerBI

En esta sección, se describen las diferentes vistas generadas en PowerBI a partir de los datos exportados. Se presentan las visualizaciones creadas y su propósito en el análisis de los datos.

5.5.1 Vista General

La vista general en PowerBI proporciona una visión amplia de los datos de películas y calificaciones. La imagen muestra varias visualizaciones clave, que se describen a continuación:

- **Total Movies**: Este indicador muestra el número total de películas en la base de datos, que es 629K.
- **Average Weighted Rating**: Este indicador muestra la calificación ponderada promedio de las películas, que es 6.52.
- **Average Run Minutes**: Este indicador muestra la duración promedio de las películas, que es 71 minutos.
- **Number of Users**: Este indicador muestra el número total de usuarios que han dado calificaciones, que es 331K.
- **Average of Number of Votes**: Este indicador muestra el promedio de votos recibidos por película, que es 102.22.
- **Average of weighted_average_rating by title**: Este gráfico de barras muestra la calificación ponderada promedio para diferentes títulos de películas. Permite identificar rápidamente las películas con mejores y peores calificaciones.
- **Count of movie_key by weighted_average_rating**: Este histograma muestra la distribución de las calificaciones ponderadas. Permite ver la cantidad de películas que tienen diferentes rangos de calificaciones.
- **Total Ratings by Year**: Este gráfico de barras muestra la cantidad total de calificaciones recibidas por año. Permite identificar tendencias y picos en la actividad de calificación a lo largo del tiempo.
- **Total Movie by Year by releasedate**: Este gráfico de líneas muestra la cantidad total de películas lanzadas cada año. Permite ver la tendencia en el número de lanzamientos de películas a lo largo del tiempo.

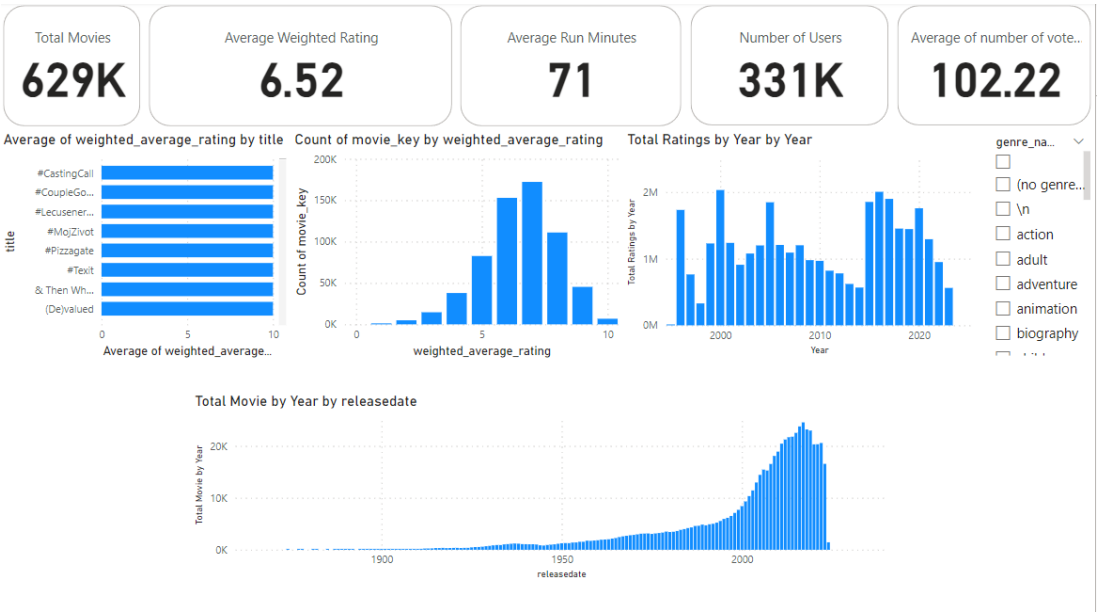

Figure 12: Vista General en PowerBI

La vista general proporciona una visión comprensiva del rendimiento y la popularidad de las películas a lo largo del tiempo. Permite a los usuarios identificar rápidamente patrones, tendencias y anomalías en los datos de películas y calificaciones.

5.5.2 Vista IMDB

La vista IMDB en PowerBI proporciona una visión específica de los datos relacionados con las calificaciones y métricas de películas según la base de datos de IMDB. La imagen muestra varias visualizaciones clave, que se describen a continuación:

- **Total Movies**: Este indicador muestra el número total de películas en la base de datos, que es 629K.
- **Average from IMDB**: Este indicador muestra la calificación promedio de las películas según los datos de IMDB, que es 6.59.
- **Average Run Minutes**: Este indicador muestra la duración promedio de las películas, que es 71 minutos.
- **Average of weighted_average_rating by title**: Este gráfico de barras muestra la calificación ponderada promedio para diferentes títulos de películas. Permite identificar rápidamente las películas con mejores y peores calificaciones.
- **Count of movie_key by averagerating**: Este histograma muestra la distribución de las calificaciones promedio. Permite ver la cantidad de películas que tienen diferentes rangos de calificaciones.
- **Total Ratings by Year**: Este gráfico de barras muestra la cantidad total de calificaciones recibidas por año. Permite identificar tendencias y picos en la actividad de calificación a lo largo del tiempo.

- **Total Movie by Year by releasedate**: Este gráfico de líneas muestra la cantidad total de películas lanzadas cada año. Permite ver la tendencia en el número de lanzamientos de películas a lo largo del tiempo.

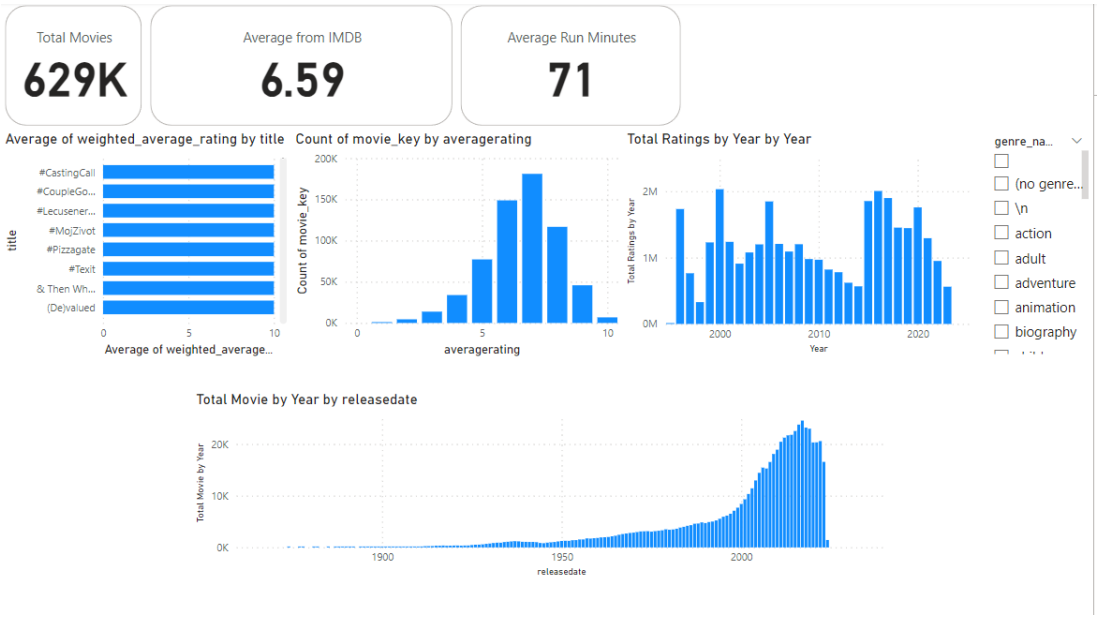

Figure 13: Vista IMDB en PowerBI

La vista IMDB proporciona una visión específica del rendimiento y la popularidad de las películas según las calificaciones y métricas de IMDB. Esta vista es útil para analizar las tendencias y patrones específicos de esta base de datos en particular.

5.5.3 Vista MovieLens

La vista MovieLens en PowerBI proporciona una visión específica de los datos relacionados con las calificaciones y métricas de películas según la base de datos de MovieLens. La imagen muestra varias visualizaciones clave, que se describen a continuación:

- **Total Movies From MovieLens**: Este indicador muestra el número total de películas en la base de datos MovieLens, que es 78K.
- **Average Weighted From MovieLens**: Este indicador muestra la calificación promedio de las películas según los datos de MovieLens, que es 6.10.
- **Average Run Minutes**: Este indicador muestra la duración promedio de las películas, que es 71 minutos.
- **Number of Users**: Este indicador muestra el número total de usuarios que han calificado películas, que es 331K.
- **Average of number of votes**: Este indicador muestra el número promedio de votos por película, que es 102.22.

- **Average Weighted From MovieLens by title**: Este gráfico de barras muestra la calificación ponderada promedio para diferentes títulos de películas. Permite identificar rápidamente las películas con mejores y peores calificaciones.
- **Count of movie_key by Rating ML**: Este histograma muestra la distribución de las calificaciones promedio. Permite ver la cantidad de películas que tienen diferentes rangos de calificaciones.
- **Total Ratings by Year**: Este gráfico de barras muestra la cantidad total de calificaciones recibidas por año. Permite identificar tendencias y picos en la actividad de calificación a lo largo del tiempo.

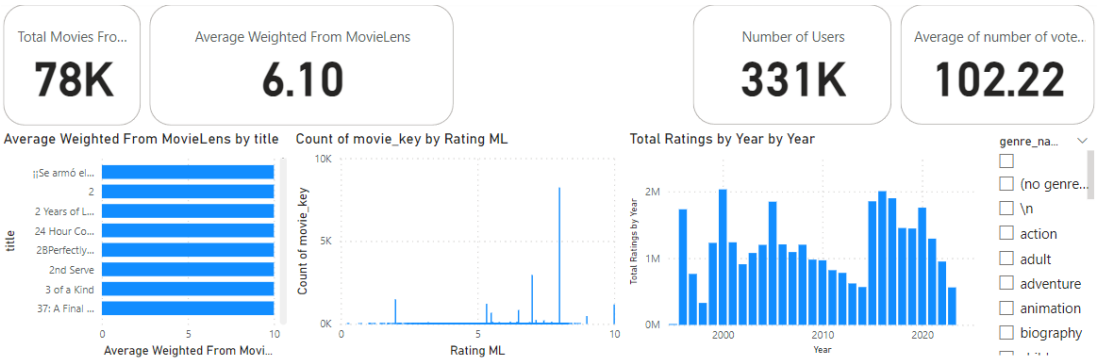

Figure 14: Vista MovieLens en PowerBI

La vista MovieLens proporciona una visión específica del rendimiento y la popularidad de las películas según las calificaciones y métricas de MovieLens. Esta vista es útil para analizar las tendencias y patrones específicos de esta base de datos en particular.

5.5.4 Interacción del Usuario: Filtrado por Género de Películas

PowerBI permite a los usuarios interactuar con los datos de manera dinámica. Una de las funcionalidades clave es la capacidad de filtrar los datos por género de películas. Esto proporciona a los usuarios una herramienta poderosa para personalizar y enfocar su análisis en categorías específicas de interés.

En las vistas presentadas, los usuarios pueden utilizar los filtros de género para seleccionar y visualizar datos correspondientes a géneros específicos. Esto se muestra en las siguientes imágenes, donde los géneros se pueden seleccionar desde una lista desplegable, permitiendo actualizar las visualizaciones en tiempo real basándose en la selección del usuario.

- **Filtrado por Género**: Los usuarios pueden seleccionar uno o varios géneros desde la lista de filtros en el panel derecho. Al seleccionar un género, todas las visualizaciones en el tablero se actualizan automáticamente para reflejar únicamente los datos correspondientes a las películas de ese género.

- **Comparación de Géneros**: Esta funcionalidad también permite a los usuarios comparar diferentes géneros entre sí. Por ejemplo, pueden seleccionar "action" y "comedy" para ver cómo se comparan las calificaciones promedio, el número de películas y otras métricas entre estos dos géneros.

A continuación, se muestran ejemplos de cómo los filtros de género afectan las visualizaciones en PowerBI:

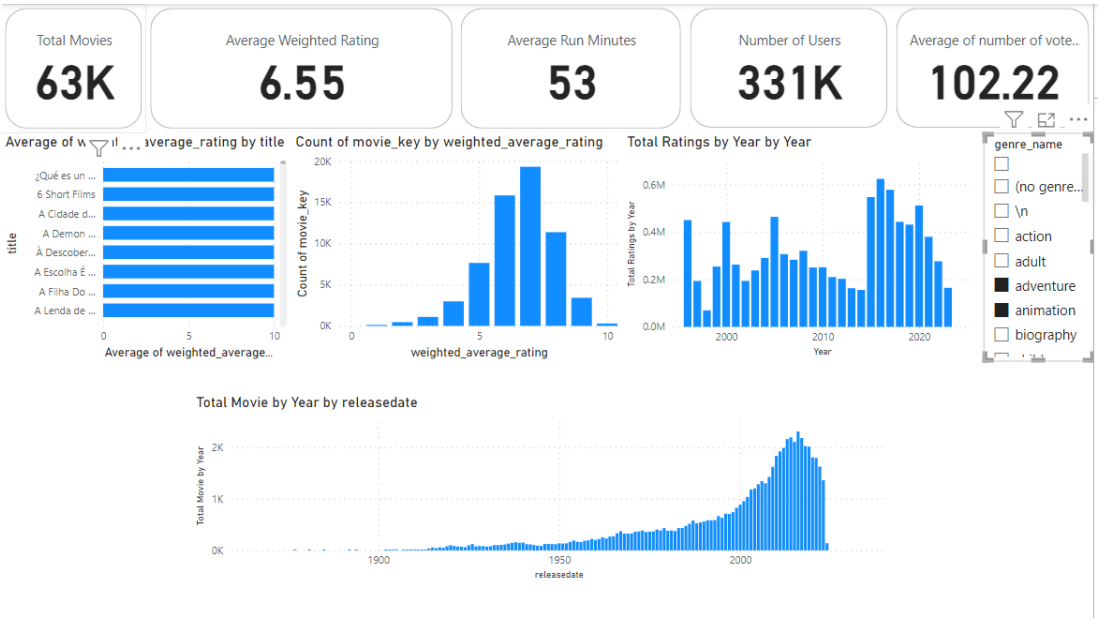

Figure 15: Vista filtrada por Género: Adventure y Animation

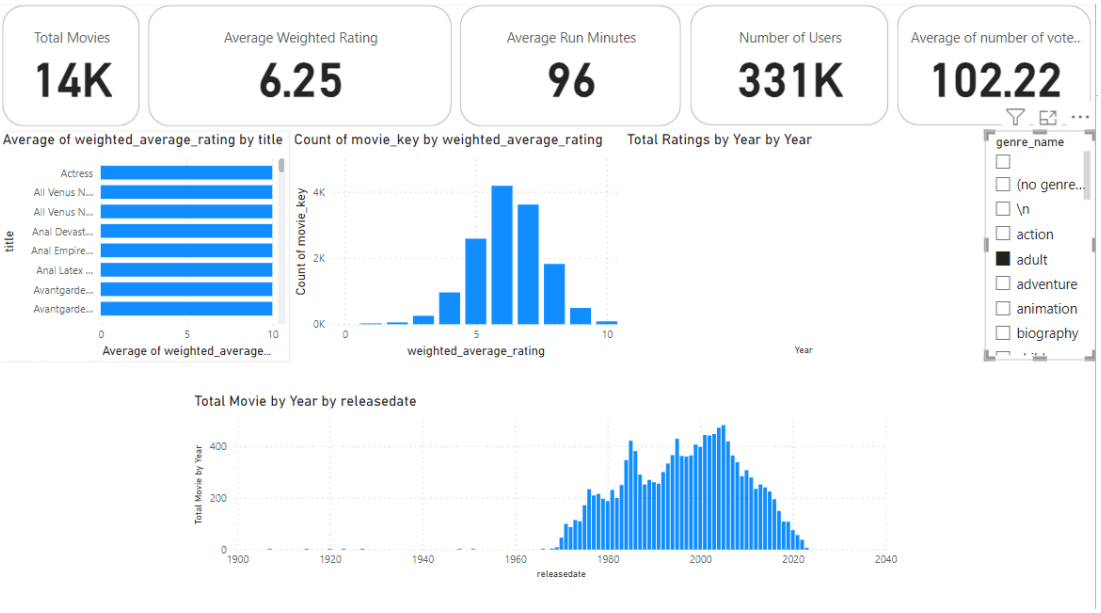

Figure 16: Vista filtrada por Género: Adulto

Esta capacidad de filtrado dinámico mejora significativamente la experiencia del usuario, permitiéndole explorar los datos de manera más detallada y obtener insights específicos según sus necesidades de análisis.

5.6 Conclusión de las sesiones 4 y 5

En estas sesiones, logramos integrar y analizar los datos de películas utilizando PowerBI, apoyándonos en la creación de vistas en PostgreSQL y la exportación de los datos a archivos CSV debido a las limitaciones de nuestro sistema para manejar grandes volúmenes de datos de forma directa. Esta metodología nos permitió manejar y visualizar los datos de manera más eficiente, aunque nos enfrentamos a ciertos desafíos técnicos.

A través de las vistas generales y específicas (IMDB y MovieLens), pudimos obtener insights valiosos sobre las calificaciones y métricas de las películas. Sin embargo, debido al tamaño de los archivos CSV y las limitaciones de rendimiento de nuestros equipos, no fue posible explorar todos los casos de uso potenciales. La carga y manipulación de grandes conjuntos de datos en PowerBI resultaron en un rendimiento lento, lo que nos impidió realizar un análisis más exhaustivo y detallado.

A pesar de estas limitaciones, el uso de vistas y la capacidad de filtrado dinámico en PowerBI nos proporcionaron una herramienta poderosa para analizar y comparar datos de diferentes géneros de películas, permitiéndonos obtener una comprensión más profunda de las tendencias y patrones en los datos de calificaciones de películas. Con equipos más potentes y recursos adicionales, se podría expandir este análisis para abarcar una gama más amplia de casos de uso y obtener insights aún más detallados.

6 Conclusión General

Este proyecto comprendió el diseño completo y la implementación de un Almacén de Datos centrado en el análisis de las calificaciones de películas. Siguiendo la metodología de Kimball, definimos un esquema en estrella adecuado para almacenar y analizar datos de calificación de películas a partir de múltiples dimensiones.

Mediante el proceso ETL, utilizando Apache HOP, pudimos extraer, transformar y cargar datos de fuentes como IMDB y MovieLens en una base de datos PostgreSQL. Superamos varios retos de limpieza e integración de datos para garantizar la coherencia de la información.

Paralelamente, investigamos y pusimos en marcha sistemas de recomendación basados en el filtrado colaborativo y de contenidos, que ofrecen a los usuarios recomendaciones de películas personalizadas a través de una interfaz gráfica.

Por último, creamos un conjunto completo de vistas en la base de datos que sirvió de base para un cuadro de mandos interactivo en PowerBI. Este cuadro de mandos permite a los usuarios visualizar y analizar intuitivamente diversas métricas y tendencias relacionadas con las clasificaciones de películas.

En conjunto, este proyecto demuestra la aplicación práctica de los principios de almacenamiento de datos, ETL y sistemas de recomendación en el campo del análisis de películas. La solución desarrollada ofrece sólidas capacidades para comprender los modelos de clasificación de películas, identificar posibles aciertos y ofrecer recomendaciones precisas a los espectadores. Además, sienta las bases para futuras ampliaciones e integraciones de nuevas funcionalidades y fuentes de datos.

References

[1] Ralph Kimball and Margy Ross. *The data warehouse toolkit: The definitive guide to dimensional modeling*. John Wiley & Sons, Nashville, TN, 3 edition, 2013.