Proyecto individual Henry

**Planteamiento y propuesta de trabajo.**

Se nos plantea el problema de realizar un sistema de recomendación de videojuegos para Steam.

Se nos entrega una base de datos sumamente sucia, por lo cual hay que primeramente hacer una EDA y posteriormente una ETL. Hay que tener en cuenta que el formato de nuestros datos es en tipo JSON.

Nuestro sistema de recomendación va en base a los comentarios de los usuarios sobre los juegos, nuestro sistema tiene que clasificar los comentarios en positivos (2), neutros (1) y malos (0).

Se nos plantea el que no son necesarias transformaciones de datos, pero para unos datos más limpios se va a realizar una ETL sobre los mismos para un resultado más prolijo.

Se tiene que crear la columna “sentiment\_analysis”, en esta es donde pondremos los resultados de nuestro sistema según si el comentario del juego es buena, mala o neutra. Esto utilizando un modelo NLP (Natural Language Processing) y esta nueva columna debe de reemplazar la de “user\_reviews.review”. Si la reseña esta ausente hay que colocar el valor de 1.

Se nos plantea el hacer el deployment de nuestro proyecto con una API mediante FastApi. Las funciones que deberá tener nuestra Api son:

* def PlayTimeGenre(genero : str ): Debe devolver año con más horas jugadas para dicho género.
* def UsersRecommend(año : int ): Devuelve el top 3 de juegos MÁS recomendados por usuarios para el año dado. (reviews.recommend = True y comentarios positivos/neutrales)
* def UsersNotRecommend(año : int ): Devuelve el top 3 de juegos MENOS recomendados por usuarios para el año dado. (reviews.recommend = False y comentarios negativos)
* def sentiment\_analysis(año : int ): Según el año de lanzamiento, se devuelve una lista con la cantidad de registros de reseñas de usuarios que se encuentren categorizados con un análisis de sentimiento.

Este modulo va más enfocado al DataEngineering que a los analytics, por lo tanto, lo más importante es:

*“El MVP tiene que ser una API que pueda ser consumida segun los criterios de API REST o RESTful desde cualquier dispositivo conectado a internet. Algunas herramientas como, por ejemplo, Streamlit, si bien pueden brindar una interfaz de consulta, no cumplen con las condiciones para ser consideradas una API, sin workarounds.”*

Así que vayamos comenzando con la EDAActualización

Actualización: 27/01/2024

Fecha de actualización: 28/01/2024

Se termino de realizar el archivo de EDA, donde pudimos leer los archivos .json y transformarlos a CSV para su posterior análisis visual.

De este análisis se obtuvieron los siguientes resultados:

1. Output\_steam\_games:
   1. Se van a conservar las columnas: Publisher, genres, title, release date, tags, specs, Price, id y developer.
   2. Se van a eliminar las columnas app name, reviews\_url y early acces.
2. Australian\_reviews:
   1. Se van a conservar las columnas userid y reviews.
   2. Se va a eliminar la columna user\_url.
3. Australian\_items:
   1. Se van a conservar las columnas user id, items count, Steam id, ítems.
   2. Se van a eliminar la columna user\_url.

Estos resultados se obtuvieron de primeramente reconocer las necesidades de nuestro proyecto, las funciones que vamos a desarrollar y los datos que necesitamos para desarrollar dichas funciones, por lo tanto, las columnas seleccionadas contienen la información pertinente para empezar a desarrollar nuestras funciones. Sin embargo, aún falta el paso de la ETL, ya que los dos datasets de ‘australian\_x’ en su última columna tenían información anidada, por lo tanto, se tiene que hacer un tratamiento a esa columna para extraer esos datos y poderlos utilizar posteriormente.

Entonces en nuestra ETL se eliminarán columnas con información poco pertinente para nuestro proyecto y se le dará un tratamiento a la última columna de australians.

Una vez terminada la ETL se procederá a continuar con el desarrollo de las funciones y el desarrollo de la API.

Fecha de actualización: 29/01/2024

El día de hoy me centré en realizar la ETL de nuestros datos. Para esto se tuvieron que expandir dos de los datasets, ya que contenían información anidada. Así que se generaron dos CSV’s más con la información expandida para en algún momento de la implementación de las funciones utilizar SQL y poner hacer el join de las tablas.

En los días posteriores se continuará con la aplicación y desarrollo de las funciones y posteriormente en el desarrollo de la API.