Sistema de recomendación para aplicación de música

Modelos Avanzados de análisis de datos II - Microproyecto 2
Universidad de los Andes. - Maestría en inteligencia analítica para la toma de decisiones

Néstor Fabián Cholo Acevedo, Jaime Orjuela Viracachá

Resumen

Una aplicación de música quiere actualizar su aplicación online para que genere recomendaciones a sus usuarios de nuevos artistas para escuchar. El sistema de recomendación debe tomar en cuenta las preferencias de cada usuario, con el fin de ofrecer recomendaciones automáticas y personalizadas.

Entre las estrategias más usadas para crear sistemas de recomendación se encuentran: por popularidad, aconseja por la "popularidad" de los productos. Por ejemplo, "los más vendidos" globalmente, se ofrecen a todos los usuarios por igual sin aprovechar la personalización. Es fácil de implementar y en algunos casos es efectiva; Basado en el contenido, a partir de productos visitados por el usuario, se intenta "adivinar" qué busca el usuario y se ofrecen mercancías similares; Filtrado colaborativo, es el más novedoso, pues utiliza la información de "masas" para identificar perfiles similares y aprender de los datos para recomendar productos de manera individual.

Keywords: Analítica, sistemas de recomendación, aprendizaje computacional, toma de decisiones

1 Metodología propuesta

En la Figura 1 se muestra la metodología implementada, la cual inicia con el entendimiento del problema, seguido del análisis exploratorio de los datos, revisión de modelos aplicables, evaluación y comparación de modelos y finaliza con el análisis de resultados y conclusiones.

Figura 1. Metodología

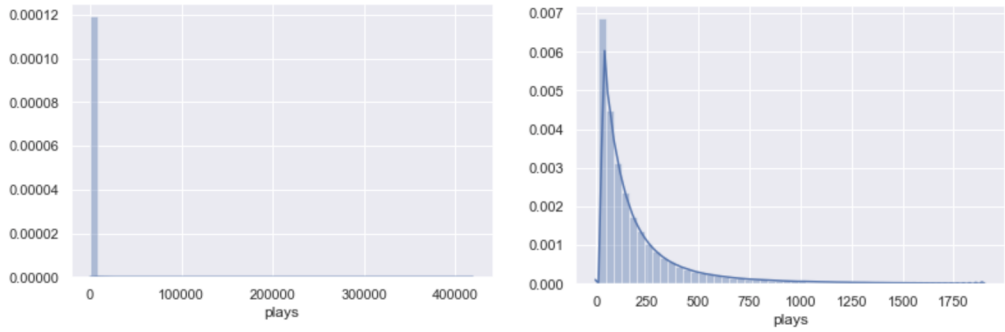
Teniendo en cuenta los datos disponibles en la base de datos del caso de análisis, se tiene acceso únicamente a información implícita, es decir que no se posee información de realimentación directa de los usuarios (si les gustó o no cada artista) sino la cantidad de reproducciones que cada usuario hizo para cada artista, se asume entonces que si un usuario escuchó con mayor frecuencia a un artista implica que ese artista es de su preferencia.

2 Análisis exploratorio de datos

Se importa el archivo de datos y se encuentra que de los 17.559.530 registros, algunos de ellos tienen información por fuera del formato establecido en el archivo README de la fuente, se realiza entonces una limpieza desde shell script (linea de comandos de la terminal) de los datos quedando con 17.328.427

Se encontró adicionalmente que muchos de los artistas era muy poco escuchados y generaban exceso de datos, por lo que se decidió agrupar por artist_id y seleccionar aquellos que tuvieran más de 300 reproducciones. Esa información quedó cargada en el archivo plano tabla4.csv que tiene 13.866.515 registros.

Considerando que se encuentran valores extremos, para evitar el sesgo debido a esos datos se quitan las colas extremas del 1 % arriba y del 10 % abajo.



(a) Histograma inicial (b) Histograma ajustado

Figura 2. Hay 350.978 usuarios y 7.428 artistas (id de artista), pero teniendo en cuenta el volumen de los datos y las limitaciones existentes de máquina, se extrae una porción (10 %) para los análisis, quedando con 1.236.271

3 Selección de modelos,resultados y análisis resultados

3.1 Recomendación por popularidad

Inicialmente se realizaron las recomendaciones buscando la pareja más popular para cada artista *Sistemas de recomendación | Aprende Machine Learning,* Se emparejó entonces cada artista con los demás escuchados por la misma persona y luego se contaron cuántas veces esa pareja estaba en el conjunto de datos. Se creó así una tabla de referencia que permitió determinar para cada artista su pareja más popular, partiendo del supuesto que si a una persona le gusta un artista lo más seguro es que también le guste su pareja más popular.

Se evidencia que aún con el 10 % de los datos (1.548.282), el numero de combinaciones resultantes es de 4.851.040; por lo que realizar este proceso para todo el conjunto de datos (13.866.515) requeriría unas capacidades computacionales muy altas que no poseemos. No obstante, buscando alternativas de desarrollo, se intentó realizar este emparejamiento en una Base de Datos MySQL, creando índices en el campo de usuario y artista, y realizando análisis segmentados mediante cursores, es decir que se accedía primero a la información del índice y luego se cargaba un buffer en memoria RAM sólo con la porción de la tabla que se requería, lo que funcionó a nivel de máquina pues no se bloqueó, sin embargo al consultar la información del QUERY EXPLAIN PLAN, había tantas iteraciones que implicaba un tiempo de procesamiento de muchas horas, por lo que también claudicamos en ese intento. Teniendo en cuenta lo anterior, continuamos el análisis en python con el conjunto de datos de prueba data_test, para lo cual se realizó un conteo por artista_a y por artista_b, para determinar de entre las parejas encontradas para el artista_a, cual era la más popular para los oyentes.

Teniendo en cuenta lo anterior, continuamos el análisis en python con el conjunto de datos de prueba data_test, para lo cual se realizó un conteo por artista_a y por artista_b, para determinar de entre las parejas encontradas para el artista_a, cual era la más popular para los oyentes.

Desempeño

De los 7.429 artistas analizados, se obtuvieron 78.986 recomendaciones. Para este método no usamos una métrica de bondad de ajuste, ya que corresponde al método de popularidad.

3.2 Filtrado colaborativo

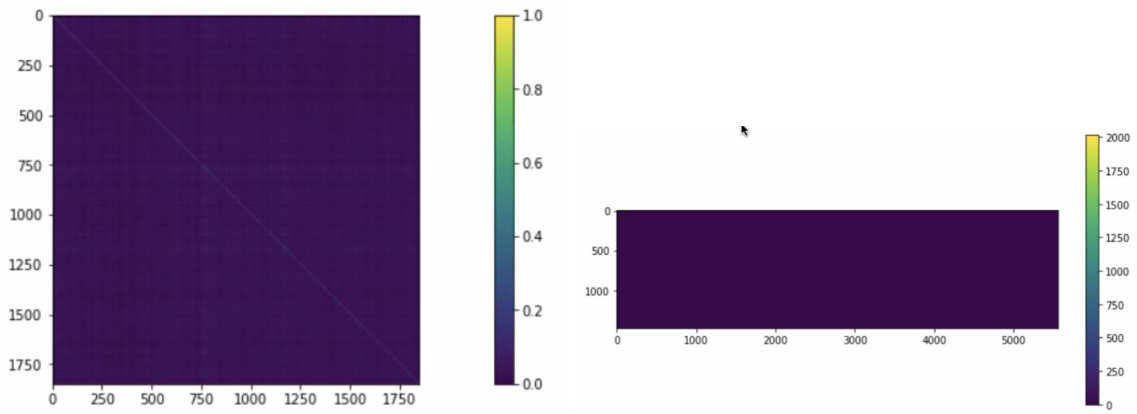
Para el desarrollo del filtrado Colaborativo, en lugar de tomar una muestra del conjunto total, se eligió como criterio de reducción el atributo de país, y se escogió a Colombia para aplicar este método después de probar con diferentes tamaños de muestra y de encontrar que no eran adecuados para este método dado que no se contaba con el nivel de detalle requerido.

De los 71.230 registros para Colombia, hay 1849 usuarios y 5556 artistas, en este punto identificamos que el gran número de artistas dificulta construir una recomendación adecuada dada la diversidad de información.

Teniendo en cuenta la alta dispersión de los datos, se usa la siguiente métrica para determinar si el filtrado

colaborativo es el más adecuado para recomendar, aún cuando pocos usuarios han valorado los mismos artistas:

```
#‘‘ceros’’ que rellenar (predecir)...
ratings = user_ratings_table.values
sparsity = float(len(ratings.nonzero()[0]))
sparsity /= (ratings.shape[0] * ratings.shape[1])
sparsity *= 100
print('Sparsity: {:4.2f}%'.format(sparsity))
Sparsity: 0.69%
```



(a) sim matrix (b) predicciones

Figura 3. Resultados filtrado colaborativo: cuanto más cercano a 1, mayor similitud entre esos usuarios, en este caso no se evidencia similitud.

Al graficar la solución para el conjunto de datos escogido, se identifica que no se obtienen predicciones dada la alta dispersión del conjunto de datos analizado. Lo cual nos impidió calcular la métrica de desempeño escogida en este caso RMSE.

4 Conclusiones

1. Como consultores externos se analiza la información recibida y se decide implementar la solución usando motores de recomendación.
2. Para llevar a cabo el requerimiento específico de: Desarrollar un algoritmo de recomendación de artistas para cada usuario y evaluar su desempeño, se implementó el método de Popularidad, mediante el cual se recomienda a los usuarios “los artistas más escuchados” globalmente, bajo esta premisa se calculan las 78.986 parejas más populares y se ofrecen a todos los usuarios por igual sin tener en cuenta la información demográfica.
3. Para llevar a cabo el requerimiento específico de: Mejorar el algoritmo considerando la información de tipo socio-demográfico por usuario, se implementó el método de filtrado colaborativo, el cual utiliza la información para identificar perfiles similares y aprender de los datos para recomendar artistas de manera individual. En la primera fase se usó toda la información disponible, pero la capacidad de cómputo requerida impidió procesar la información, y se plantearon las siguientes estrategias: segmentar por muestreo o por país, escogiendo a Colombia por afinidad y conocimiento de los datos y realizar una muestra aleatoria a nivel global la cual dada su heterogeneidad no permitió obtener resultados implementables.

5 Recomendaciones

1. La implementación de un sistema de recomendación se centra en tener los datos correctos, un volumen alto de información y altas capacidades de computo. Se filtró la base de datos de tal manera que de los 292.363 artistas iniciales, quedaron 7.424 los cuales acumulan el 80 % de las reproducciones, esta decisión se tomo después de

iterar los modelos de popularidad y filtrado colaborativo con la base de datos inicial y no obtener resultados interpretables.

2. Teniendo en cuenta el gran volumen de los datos, se puede buscar otras herramientas computacionales diversas diferentes as python, como lo son el shell script de Linux o MacOS, Bases de datos entre otros.

Referencias

Sistemas de recomendación | Aprende Machine Learning. Visitado 30 de noviembre de 2020. https://www.aprendemachinelearning.com/sistemas-de-recomendacion/.