

Algoritmos y Estructura de Datos

Proyecto 2 Fase # 1

Investigación de Algoritmos existentes para resolver la situación representada.

Introducción

Para el Proyecto 2 se realizará un programa de recomendación de videojuegos, donde el usuario ingresara sus gustos y a base de eso se realizarán las recomendaciones. Se investigarán diferentes algoritmos de recomendación para poder elegir el mejor para el Proyecto 2.

Algoritmos de Recomendación

Los sistemas de recomendación son algoritmos y técnicas diseñados para predecir y sugerir elementos que podrían interesar a un usuario específico. Y se basan en el análisis de datos para comprender patrones y preferencias, lo que permite ofrecer sugerencias personalizadas, (Bi, 2024).

Los sistemas de recomendación se basan en datos. Recopilan información sobre las preferencias, el comportamiento y los patrones de consumo de los usuarios. Luego, realizan el análisis de datos y utilizan algoritmos para identificar similitudes, tendencias y relaciones entre usuarios y los elementos que consumen, (Bi, 2024).

Existen diversos tipos de Algoritmos de recomendación

- **Sistemas de popularidad:** Toman como referencia la popularidad del objeto de estudio por una variable principal la cual es una característica especial y se muestra de forma general a todos los usuarios que investiguen el área a la que pertenece el objeto. (GraphEverywhere, 2019).
- **Sistemas de Contenido:** Analizan las características y atributos de los elementos recomendados y los comparan con las preferencias previas del usuario, (Bi, 2024).
- **Sistemas colaborativos:** Genera recomendaciones analizando datos, identificando perfiles y haciendo contraste entre la información del perfil de usuario y la de un colectivo de usuarios, (GraphEverywhere, 2019).

- Filtrado Colaborativo: Se basa en el comportamiento y las elecciones de múltiples usuarios. Si dos usuarios tienen preferencias similares en el pasado, es probable que sus recomendaciones coincidan, (Bi, 2024).
- Modelos Híbridos: Estos sistemas combinan los algoritmos de recomendación basados en contenido y filtrado colaborativo para proporcionar sugerencias más completas, (Bi, 2024).

Ahora veremos los principales tipos de algoritmos aplicados a los sistemas de recomendación:

- Algoritmo K de vecinos cercanos o Nearest Neighbourhood: Este algoritmo clasifica los datos para predecir el comportamiento o el cambio de estos. El algoritmo determina un número K de puntos cercanos o vecinos para clasificar, donde todos los puntos ya son conocidos debido a que es un algoritmo supervisado. Los puntos conocidos, o las preferencias del usuario son analizadas y contrastadas con otros elementos que se encuentran “cerca”. Esto busca que las observaciones de los datos sean cercanas y permitan predecir con facilidad las sugerencias que necesita el usuario, (GraphEverywhere, 2019a).
- Correlación de Pearson: Estos son los algoritmos que toman el concepto de correlación de Pearson los cuales toman en cuenta un índice de desigualdad. Establecen valores de +1 y -1 para determinar la existencia de correlaciones lineales, no lineales o negativas. Dependiendo del valor arrojado por el cálculo se puede determinar si un elemento o conjunto de datos es similar a otro. Se utiliza para evaluar los elementos y determinar si existe una semejanza que se ajuste a los datos del usuario, (GraphEverywhere, 2019a).
- Algoritmo de distancia euclidiana: Aplica una fórmula matemática que determina la distancia existente entre dos puntos de un espacio n-dimensional. Esto permite establecer la similitud entre dos elementos o pares de datos. Ayuda a conocer esquemas de datos de características similares. Puede tomar como referencia un esquema de puntuaciones o valoraciones para desarrollar las recomendaciones, (GraphEverywhere, 2019a).
- Algoritmo de similitud de coseno: Calcula la similitud de coseno que es una medida que se obtiene entre dos vectores dentro del espacio interno de un producto en el que se mide un ángulo. El coseno de 0 grados es 1, y es menor que 1 para cualquier ángulo que se encuentre en el intervalo $(0, \pi]$ radianes. Este cálculo permite recuperar información muy importante de un conjunto amplio de datos ya que permite determinar si existe cohesión o similitud entre los conjuntos a analizar. Este algoritmo también sirve para estudiar o detectar repeticiones de datos, como veces que aparece un calor específico en un documento, permitiendo a los motores de recomendaciones calcular y predecir sugerencias similares, (GraphEverywhere, 2019a).

¿Qué algoritmos podemos utilizar?

El sistema implementa un enfoque híbrido que combina los algoritmos basados en contenido y los sistemas colaborativos. El componente principal es el algoritmo k-Vecinos Más Cercanos, que analiza las características similares de los videojuegos. Este algoritmo utiliza dos métricas fundamentales: el Coeficiente de Jaccard para comparar géneros mediante análisis de conjuntos y la Distancia Euclidiana para evaluar variables numéricas como horas jugadas y puntuaciones en un espacio multidimensional. Para cada juego que el usuario ha disfrutado, el sistema identifica los k juegos más similares que aún no ha probado.

Luego está el filtrado colaborativo mejorado, que analiza los patrones de comportamiento de usuarios con gustos afines. Examina las interacciones completas considerando múltiples señales como reseñas y tiempo de juego. Al combinar ambos sistemas mediante una ponderación dinámica, donde el peso relativo de cada componente se ajusta automáticamente según la disponibilidad de datos, dando mayor peso a k-NN para usuarios nuevos y al componente colaborativo cuando existe suficiente historial de interacciones.

Antes de llegar a soluciones más complejas como el algoritmo de k-Vecinos o los sistemas híbridos, es importante considerar un enfoque inicial más sencillo y directo: los algoritmos de recomendación basados en listas. Estos algoritmos se fundamentan en estructuras simples como ArrayList en Java o list en Python, donde los gustos, atributos o historiales de usuarios se almacenan y manipulan directamente para producir recomendaciones.

En esencia, este enfoque compara directamente listas de videojuegos favoritos entre usuarios (filtrado colaborativo simple) o analiza los atributos de los videojuegos que le gustan a un usuario para encontrar otros similares (filtrado basado en contenido). Por ejemplo, si un usuario ha marcado como favoritos juegos de rol y fantasía, el sistema recorre la lista de todos los videojuegos disponibles y recomienda aquellos que compartan géneros similares y que aún no ha jugado. Este proceso se basa en intersecciones de listas, búsqueda de coincidencias y filtros por atributos compartidos.

La principal ventaja de este enfoque es su simplicidad e interpretabilidad. No requiere cálculos complejos ni entrenamiento previo. Es ideal para sistemas de escala pequeña a media, como prototipos académicos o pruebas funcionales, y permite una primera aproximación al comportamiento de los usuarios antes de implementar métodos más avanzados. Además, en combinación con bases de datos orientadas a grafos como Neo4j, las listas pueden representarse mediante relaciones entre nodos (USUARIO → JUEGO) y procesarse mediante consultas Cypher desde un backend en Python.

Aunque este método tiene limitaciones en cuanto a escalabilidad y profundidad de análisis, es sumamente útil como fase 1 del desarrollo, donde se busca validar la lógica de recomendación y comprender los patrones de consumo antes de añadir algoritmos más robustos como k-NN, distancia euclidiana o similitud de coseno. De hecho, muchos sistemas reales utilizan esta lógica como base inicial para comparar la efectividad de otros algoritmos más sofisticados.

Conclusiones

Utilizar el algoritmo híbrido supera las limitaciones de los sistemas de recomendación tradicionales al ofrecer lo mejor de ambos mundos. Por un lado, el k-NN garantiza recomendaciones personalizadas basadas en las características objetivas de los juegos, funcionando incluso para nuevos usuarios. Por otro lado, el componente colaborativo introduce diversidad y descubre conexiones sociales no evidentes. La integración de ambos métodos mediante ponderación dinámica crea un sistema adaptable que mejora continuamente sus sugerencias.

Bibliografías

Bi, T. (2024, 22 mayo). Sistemas de Recomendación - Tecnología bi. *Tecnología bi*. <https://tecnologiabi.com/sistemas-de-recomendacion/#:~:text=Los%20sistemas%20de%20recomendaci%C3%B3n%20son,redes%20sociales%20y%20mucho%20m%C3%A1s>.

GraphEverywhere, E. (2019a, diciembre 2). *Sistemas de recomendación | Algoritmos de funcionamiento*. GraphEverywhere. <https://www.grapheverywhere.com/sistemas-de-recomendacion-algoritmos-de-funcionamiento/>

GraphEverywhere, E. (2019b, diciembre 2). *Sistemas de recomendación | Qué son, tipos y ejemplos*. GraphEverywhere. <https://www.grapheverywhere.com/sistemas-de-recomendacion-que-son-tipos-y-ejemplos/#:~:text=Los%20sistemas%20de%20recomendaci%C3%B3n%20basados,mayor%20presencia%20en%20la%20actualidad>.