



Proyecto Final: Algoritmo de recomendación de Videojuegos



Universidad Rafael Urdaneta
INSTITUCIÓN PRIVADA SIN FINES DE LUCRO

Asignado por: Paola Rodríguez

Realizado por:

- Leonardo Chourio C.I.: 25.950.178
- Clemente Castejón C.I.: 27.206.278
- Diego Torres C.I.: 27.284.855
- Ricardo Graziano C.I.: 27.637.837
- María Prada C.I.: 28.137.243
- William Prada C.I.: 28.137.251

Introducción

En este trabajo buscamos diseñar el algoritmo pseudo-inteligente que simule el aprendizaje para un programa que mediante unas preguntas evalúe el perfil de un usuario y otorgue una recomendación de un videojuego que coincida con los gustos descritos por el usuario y a su vez acepte recomendaciones del usuario que son guardadas en una base de datos y que pueden ser usadas para futuras recomendaciones.

Refresquemos conocimientos básicos. Un algoritmo es un conjunto de instrucciones o reglas definidas y no-ambiguas, ordenadas y finitas que permite, típicamente, solucionar un problema, realizar un cómputo, procesar datos y llevar a cabo otras tareas o actividades

Podemos clasificar los algoritmos en dos ramas, supervisados y no supervisados.

Algoritmo supervisado

Es el más común de los dos. Obtiene su nombre porque el desarrollador actúa como una guía a que aprenda o que tome su decisión. Es decir, aprenden de los datos introducidos por una persona, realiza funciones como la clasificación y la regresión.

Algoritmo no supervisado

Está mucho más alineado con lo que es la Inteligencia Artificial, en estos algoritmos la computadora puede aprender sin un ser humano guiando o dando seguimiento, usa técnicas como el Clustering y la asociación.

Nuestro diseño de algoritmo es supervisado. Sabemos que para la economía de un individuo o ente es racional si entre varias decisiones elige la que le provea la máxima utilidad, que no es más que la cantidad de beneficios que se obtienen por consumir un bien o servicio, dicho esto podemos decir que nuestro algoritmo es un ente racional ya que mientras más información sobre usuarios y juegos introducidos (Introducidos por los usuarios) recopila, más aumenta su utilidad hasta que llega a un umbral máximo donde el aumento de utilidad deja de ser

proporcional al consumo de bienes que en este caso es información sobre usuarios y videojuegos.

Desarrollo del Algoritmo

Para alcanzar nuestro objetivo, decidimos realizar un algoritmo que tenga en cuenta estrictamente el perfil del usuario, cosas como:

- La edad: Tomada en cuenta principalmente para dividir en dos categorías, por lo que decidimos crear un atributo booleano `+18` para obtener la edad estimada y así no recomendar juegos no aptos para menores. Esto divide nuestro algoritmo en dos grandes ramas. Excluyendo los juegos +18 para los menores pero permitiendo las recomendaciones de juegos infantiles para mayores.
- Gustos e Intereses del Usuario: Realizando diversas preguntas para conocer los gustos e intereses del usuario así como también descartando opciones al conocer que no le gusta al usuario, para asociar el perfil obtenido con el usuario creamos los siguientes atributos booleanos.
 - `Espíritu_Aventurero` : Nos sirve para saber si al usuario le gustan las aventuras.
 - `Grandes_Paisajes` : De esta manera podemos saber si el usuario quiere una aventura de mundo abierto o prefiere una aventura lineal.
 - `Cómics_Anime` : Para un buen fanático de los cómics o del Anime, podemos recomendarle un juego con ese estilo.
 - `Armas` : El usuario puede tener un disgusto hacia las armas, como también puede estar muy interesado en ellas, muchos juegos contienen armas de fuego.
 - `Competitivo` : Nos ayuda a conocer si el usuario es una persona competitiva y prefiere un juego donde haya competencia, ya sea con otros jugadores o contra una computadora.
 - `Acción` : Indica si el usuario se siente atraído hacia la acción o si decide evitarla.
 - `Deportes` : Para reconocer a un fanático de los deportes.

- **Vehículos** : Nos indica si el usuario es un fanático de los vehículos.
- **C_Callejeras** : A una persona fanática de los vehículos pueden no gustarles las carreras callejeras.
- **Seriedad** : Un jugador puede preferir un juego que se tome a si mismo enserio por encima de un juego humorístico.
- **Creativo** : Indica si el usuario se considera una persona creativa, para así recomendar juegos donde pueda dejar fluir su creatividad, consideramos juegos creativos los juegos que permitan la libertad de expresión del jugador.
- **Construcción** : Indica si el usuario (previamente diagnosticado como una persona creativa) tiene un interés en construir estructuras como forma de expresión creativa.
- **Adrenalina** : Nos indica si al usuario le gusta una buena dosis de adrenalina.
- **Valiente** : Nos permite conocer si el usuario puede aguantar un juego con un poco de terror.

Diagrama de Base de Datos

Con estos atributos, guardaremos los juegos que recomendaremos.

Utilizaremos una base de datos NOSQL (Not Only SQL) tambien conocida como una base de datos no relacional y guardaremos cada juego como un Objeto BSON con los mismos atributos. Nuestra base de datos será MongoDB.

Si el usuario no se siente satisfecho con la recomendación, se le otorga la opción de agregar un juego de su gusto para mejorar la experiencia, solo se le pedirá al usuario el nombre del juego que será el atributo **Titulo_del_Juego** ya que el usuario ya ha otorgado las demás características que necesitamos para guardar este juego.

Con todo esto dicho nuestros objetos serán guardados con la siguiente estructura:

```
"_id": {
  "$oid": "5f4000c905dbfa23982cd1b6" //Index generado automaticamente para
},                                     //cada objeto agregado.
"Titulo_del_Juego": "Juego",
```

```
"+18": true,  
"Espíritu_Aventurero": true,  
"Grandes_Paisajes": true,  
"Cómics_Anime": true,  
"Armas": true,  
"Competitivo": true,  
"Acción": true,  
"Deportes": true,  
"Vehículos": true,  
"C_Callejeras": true,  
"Seriedad": true,  
"Creativo": true,  
"Construcción": true,  
"Adrenalina": true,  
"Valiente": true.
```

Ejemplo de Uso del Algoritmo



