# Filtrado basado en contenido

## Introducción

La tecnología digital cada vez se integra más a nuestra vida cotidiana, en los últimos años ha habido un

Las transacciones vía internet se han incrementado de manera considerable en los últimos años, tan solo de 2015-2020 el porcentaje de usuarios en México que realizan transacciones se incrementó en un 20%, al día de hoy la cifra debe ser mucho mayor debido a la pandemia surgida durante el 2019.

Ilustración . Proporción de usuarios que realizan transacciones vs usuarios de internet.

Los

## Filtrado basado en contenido

El filtrado basado en contenido (FBC) es una de las técnicas de recomendación más exitosas, esta técnica utiliza los atributos de los objetos que el usuario ha consumido, visto o mostrado interés para sugerir nuevos objetos con atributos similares basado en el nivel de correlación entre el objeto observado y el objeto sugerido.

Existen diferentes métricas utilizadas para medir el nivel de correlación entre existente entre dos objetos, éstas son llamadas medidas de similitud y algunas de las más usuales son:

### Vectores Similares (Coseno)

Esta técnica calcula el coseno entre dos vectores de atributos de los objetos como medida de correlación, el resultado es un valor que va desde 0 hasta 1, donde 1 implica que los vectores son paralelos y por lo tanto presentan una correlación fuerte entre ellos, mientras que un valor 0 significa que los vectores son perpendiculares entre sí y no presentan una buena correlación. Su fórmula es:

### Coeficiente de Correlación de Pearson

Esta métrica fue propuesta por Karl Pearson como medida de correlación lineal entre dos vectores (muestras), el resultado es un valor que va desde -1 a 1, donde -1 indica una fuerte correlación lineal negativa, 0 una nula correlación lineal y 1 una fuerte correlación lineal positiva. Su fórmula es:

### Distancia Euclidiana

La métrica calcula la distancia entre dos puntos n dimensionales (donde n = número de atributos) en un espacio Euclidiano, si la distancia entre los puntos es pequeña, indica que ambos objetos presentan características muy similares, mientras que, si aumenta la diferencia entre éstos, la distancia aumenta. La fórmula es:

### Distancia Manhattan

Es también conocida como cuadras de ciudad (city blocks), la métrica evalúa la suma de las diferencias absolutas, es decir, siempre sumará distancias positivas, lo cual se representa como una trayectoria que se desplaza solamente de manera horizontal y vertical (de ahí su nombre). Su fórmula es:

## Conjunto de datos

Los datos ha utilizar en este proyecto fueron extraídos del sitioweb [The Movie Data DB](https://www.themoviedb.org/) a través de su API. Para la obtención de una llave API puede consultarse el archivo [readme.md](https://github.com/juanagsolano/content_based_filtering/blob/4070c0d5f5f8ef6fcc645d78fb22f8efcd071a10/README.md) anexo en la [carpeta del proyecto](https://github.com/juanagsolano/content_based_filtering).

### Recopilación de datos por película

Para ilustrar la obtención de datos utilizaremos la siguiente película como ejemplo:

Id = 20 Nombre: My Life Without Me

Los detalles de la película se pueden obtener a través de la siguiente url:

https://api.themoviedb.org/3/movie/[movie\_id]?api\_key=[api\_key]

donde movie\_id = 20 es el identificador de la película y api\_key = “---” es la llave generada para el uso de la herramienta.

Para la obtención de la información hacemos uso de la biblioteca requests de Python, el resultado es un archivo en formato json, el cual podemos cargar mediante la biblioteca json. El resultado del proceso es el siguiente:

{'adult': False,

'backdrop\_path': '/kZyurQjTMLHalUxs7sHgH5XeiwO.jpg',

'belongs\_to\_collection': None,

'budget': 2500000,

'genres': [{'id': 18, 'name': 'Drama'}, {'id': 10749, 'name': 'Romance'}],

'homepage': 'http://www.clubcultura.com/clubcine/clubcineastas/isabelcoixet/mividasinmi/index.htm',

'id': 20,

'imdb\_id': 'tt0314412',

'original\_language': 'en',

'original\_title': 'My Life Without Me',

'overview': 'A fatally ill mother with only two months to live creates a list of things she wants to do before she dies without telling her family of her illness.',

'popularity': 13.863,

'poster\_path': '/9Fa7tCEKIha1llGH7E41mxSpaF6.jpg',

'production\_companies': [{'id': 49,

'logo\_path': '/xpf5iHdvvBtsH8jBMlgIJHAET0c.png',

'name': 'El Deseo',

'origin\_country': 'ES'},

{'id': 77,

'logo\_path': None,

'name': 'Milestone Productions',

'origin\_country': ''}],

'production\_countries': [{'iso\_3166\_1': 'CA', 'name': 'Canada'},

{'iso\_3166\_1': 'ES', 'name': 'Spain'}],

'release\_date': '2003-03-07',

'revenue': 12300000,

'runtime': 106,

'spoken\_languages': [{'english\_name': 'English',

'iso\_639\_1': 'en',

'name': 'English'}],

'status': 'Released',

'tagline': '',

'title': 'My Life Without Me',

'video': False,

'vote\_average': 5.8,

'vote\_count': 364}

### Atributos a utilizar dentro del modelo

Para los propósitos de este proyecto, solo utilizaremos una porción de los atributos mostrados, concretamente éstos son:

* budget: presupuesto de la película.
* genres: lista de géneros a los que pertenece.
* id: identificador de la película.
* original\_title: título original de la película.
* popularity: popularidad de la película (métrica generada por el sitio).
* release\_date: fecha de lanzamiento.
* revenue: ingresos recaudados.
* runtime: duración de la película.
* vote\_average: calificación media otorgada por usuarios.
* vote\_count: cantidad de usuarios que calificaron la película.

### Binarización del atributo género

Dado que este atributo es una lista de géneros, es necesario binarizarla para poder evaluarla en el modelo de FBC. Para ello necesitamos conocer los diferentes géneros que utiliza esta base de datos, esto lo logramos a través de su API mediante la siguiente url:

https://api.themoviedb.org/3/genre/movie/list?api\_key=[api\_key]&language=en-US

Este enlace nos retorna los géneros oficiales utilizados, los cuales son:

{28: 'Action', 12: 'Adventure', 16: 'Animation', 35: 'Comedy', 80: 'Crime', 99: 'Documentary', 18: 'Drama', 10751: 'Family', 14: 'Fantasy', 36: 'History', 27: 'Horror', 10402: 'Music', 9648: 'Mystery', 10749: 'Romance', 878: 'Science Fiction', 10770: 'TV Movie', 53: 'Thriller', 10752: 'War', 37: 'Western'}

Esta binarización realizará la siguiente transformación:

|  |  |
| --- | --- |
| Atributo antes de procesarse | Atributo después de procesarse |
| [{'id': 18, 'name': 'Drama'}, {'id': 10749, 'name': 'Romance'}] | {'Action': 0, 'Adventure': 0, 'Animation': 0, 'Comedy': 0, 'Crime': 0, 'Documentary': 0, 'Drama': 1, 'Family': 0, 'Fantasy': 0, 'History': 0, 'Horror': 0, 'Music': 0, 'Mystery': 0, 'Romance': 1, 'Science Fiction': 0, 'TV Movie': 0, 'Thriller': 0,  'War': 0, 'Western': 0} |

### Atributos por película

Los atributos para esta película quedarían de la siguiente manera:

------------------------------------------------------------------

### Directorio de películas de la base de datos

Para realizar este proceso necesitamos saber qué películas se encuentran en la base de datos, la API nos brinda acceso a un directorio de películas, durante este proyecto se procesarán todas las películas del directorio actualizado al día 24 de marzo de 2022, el directorio se encuentra en el siguiente enlace:

<http://files.tmdb.org/p/exports/movie_ids_03_24_2022.json.gz>

La cantidad de películas a procesar en la base de datos es de 688316.

### Obtención de datos por lotes

Dado que la cantidad de películas es larga, esta se procedió a realizarla en lotes, en total son 4000 lotes que contienen 172 películas cada uno, esto se hizo así para evitar errores por problemas de conexión o tamaño de memoria, ya que, si existe un error, solo es necesario reiniciar el contador al número de lote en el que se quedó.

El resultado es un archivo csv llamado [dataset\_movies.csv](https://github.com/juanagsolano/content_based_filtering/blob/4070c0d5f5f8ef6fcc645d78fb22f8efcd071a10/dataset_movies.csv) al cual se le va añadiendo los datos procesados por lote, de modo que entre más lotes se procesen, más películas contendrá el conjunto.

### Manejo de valores nulos

En muchas películas se desconoce ciertos datos como el presupuesto, fecha de lanzamiento, ingreso recaudado, duración, etc. Hasta este punto se manejarán estos valores nulos cambiándolos por valor 0, además de realizar un chequeo para saber si existen duplicados.

### Exportación de la base de datos

Al realiza la iteración por lotes, se obtiene un set de datos en formato csv, sin embargo, contiene valores nulos, por lo que en este punto solo se realiza una actualización de dicho set de datos para lidiar momentáneamente con estos valores nulos. El set de datos final es el siguiente:



Todo el código y procedimientos abarcados hasta punto pueden encontrarse en el archivo [etl\_process.ipynb](https://github.com/juanagsolano/content_based_filtering/blob/98f85851d27423cf0da11924f709061dbdac3060/etl_process.ipynb) del proyecto.

## Preprocesado de datos para el modelo FBS

## Preprocesado de datos para el modelo

## Métricas utilizadas

## Bibliografía

De 2013 a 2014: INEGI. Módulo sobre Disponibilidad y Uso de Tecnologías de la Información en los Hogares. Consultado https://www.inegi.org.mx/temas/ticshogares/