

Trabajo 2

Datos Base

SCAR

**Sistemas Complejos Adaptativos y
Recomendación**



Official Master's Degree in Artificial Intelligence,
Pattern Recognition and Digital Imaging

MIARFID

Objetivos

Los elementos a recomendar (ítems) son películas de la BD de MovieLens

El objetivo es conseguir una lista de películas (lista de ítems recomendados) adaptada al usuario

Para ello, se calcula un ratio de interés del usuario en cada una de las películas

Se recomiendan las películas de mayor ratio

Objetivos

El objetivo de esta parte del trabajo es

Preparar la forma en la que se almacenan los datos de los ficheros del dataset

Definir las estructuras de datos necesarias para el funcionamiento del recomendador


Dataset

Dataset de Movielens

<https://grouplens.org/datasets/movielens/>

- Muy utilizado en SR
- 100.000 puntuaciones de usuarios a películas
- Contiene datos de 943 usuarios, con puntuaciones a 1682 películas (en conjunto)
- Cada usuario tiene puntuadas, al menos, 20 películas (con un ratio entre 1 y 5)

Dataset



Existe otro dataset con
26.000.000 de datos

Hay un dataset de películas que
tiene enlaces a las carátulas

- <https://grouplens.org/datasets/hetrec-2011/>

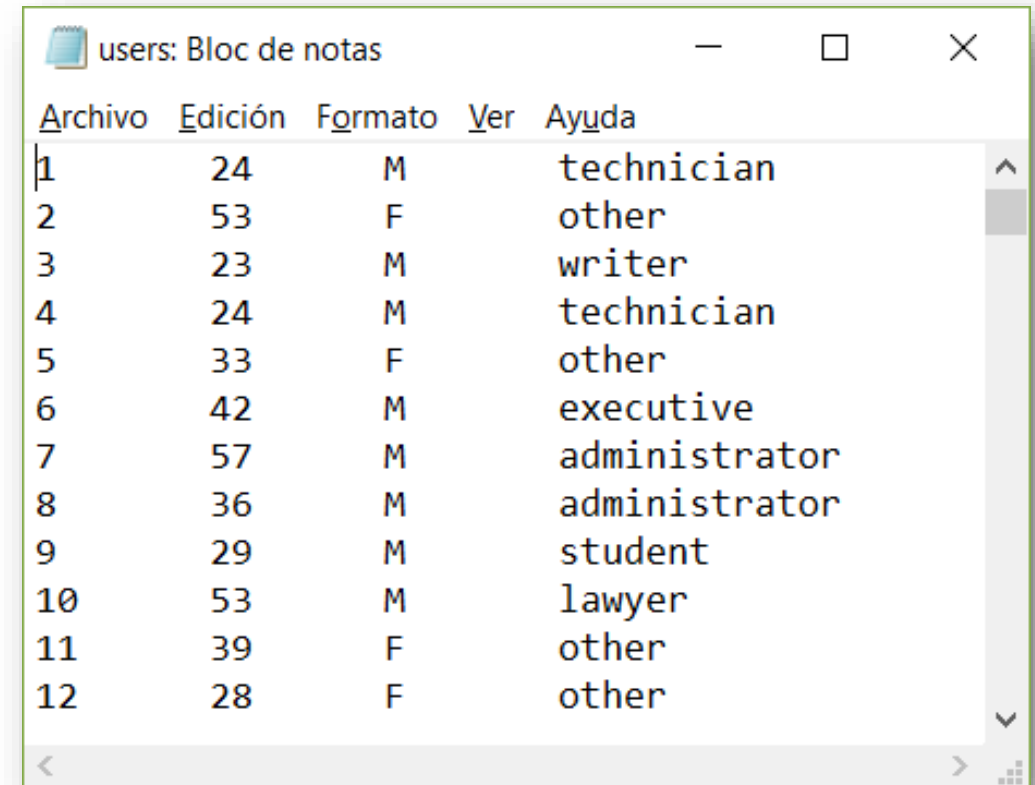
Para la práctica, y para facilitar
su tratamiento, se ha realizado
un proceso previo de los
ficheros del dataset

- Se deja disponible el dataset original por si
alguna persona desea utilizar su propio
proceso de datos

Ficheros del dataset

Users.txt

- Datos demográficos de los usuarios
- Para cada usuario contiene
 - Identificador
 - Edad
 - Sexo (**F**emale o **M**ale)
 - Ocupación



Archivo	Edición	Formato	Ver	Ayuda
1	24	M	technician	
2	53	F	other	
3	23	M	writer	
4	24	M	technician	
5	33	F	other	
6	42	M	executive	
7	57	M	administrator	
8	36	M	administrator	
9	29	M	student	
10	53	M	lawyer	
11	39	F	other	
12	28	F	other	

Ficheros del dataset

Genre.txt

- Géneros o categorías en los que se clasifican las películas
- Contiene **19** géneros, identificados del 0 al 18
- El fichero contiene por cada género
 - Identificador
 - Descripción

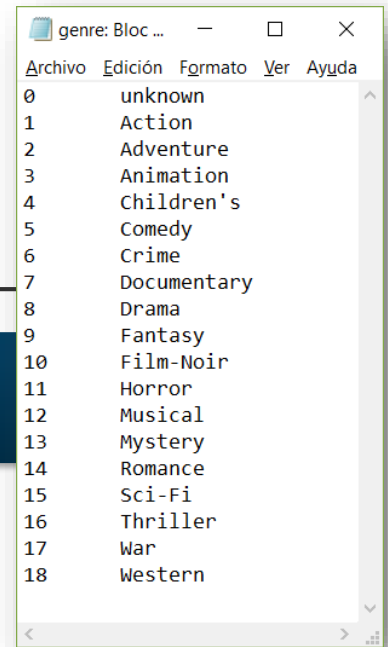


Archivo	Edición	Formato	Ver	Ayuda
0	unknown			
1	Action			
2	Adventure			
3	Animation			
4	Children's			
5	Comedy			
6	Crime			
7	Documentary			
8	Drama			
9	Fantasy			
10	Film-Noir			
11	Horror			
12	Musical			
13	Mystery			
14	Romance			
15	Sci-Fi			
16	Thriller			
17	War			
18	Western			

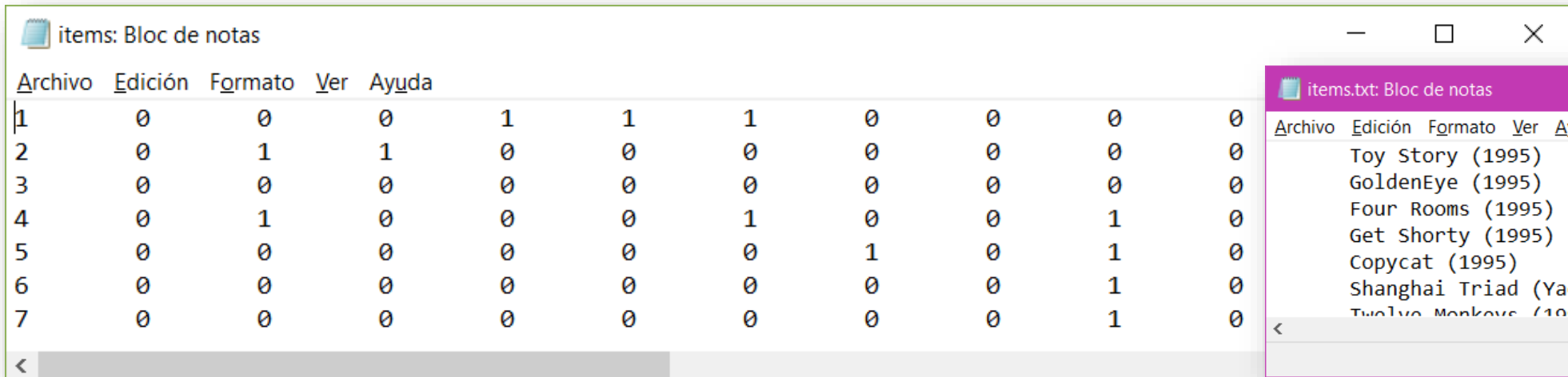
Ficheros del dataset

Items.txt

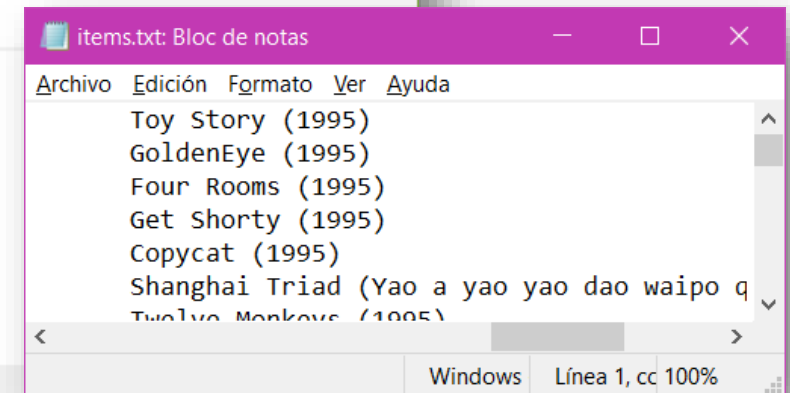
- Clasificación de las películas en géneros
- Contiene
 - Id de la película
 - Ratio en cada uno de los 19 géneros (el ratios es sólo 0 o 1, indicando que la película se clasifica en ese género o no)
 - Título de la película



	genre
0	unknown
1	Action
2	Adventure
3	Animation
4	Children's
5	Comedy
6	Crime
7	Documentary
8	Drama
9	Fantasy
10	Film-Noir
11	Horror
12	Musical
13	Mystery
14	Romance
15	Sci-Fi
16	Thriller
17	War
18	Western



	Id	Action	Adventure	Animation	Children's	Comedy	Crime	Documentary	Drama	Fantasy	Film-Noir	Horror	Musical	Mystery	Romance	Sci-Fi	Thriller	War	Western
1	1	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
2	2	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	4	0	1	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0
5	5	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
6	6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0



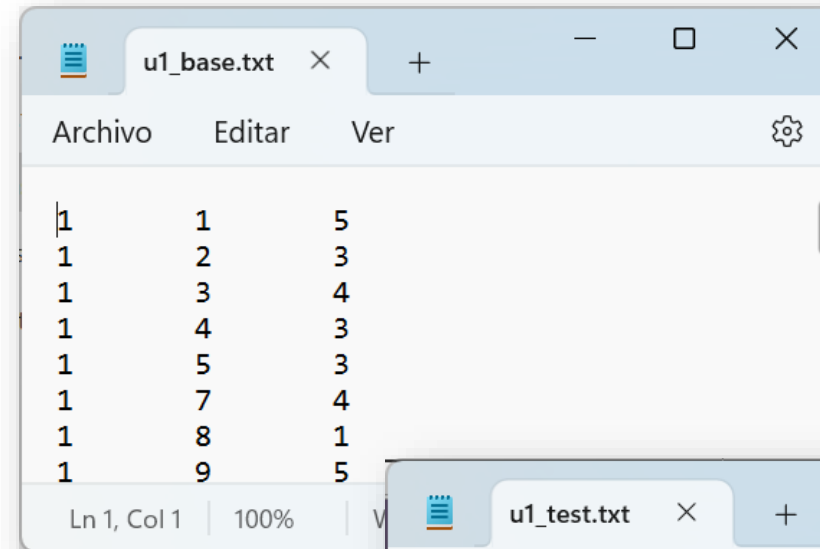
	items.txt
1	Toy Story (1995)
2	GoldenEye (1995)
3	Four Rooms (1995)
4	Get Shorty (1995)
5	Copycat (1995)
6	Shanghai Triad (Yao a yao yao dao waipo q)
7	Twelve Monkeys (1995)

Ficheros del dataset

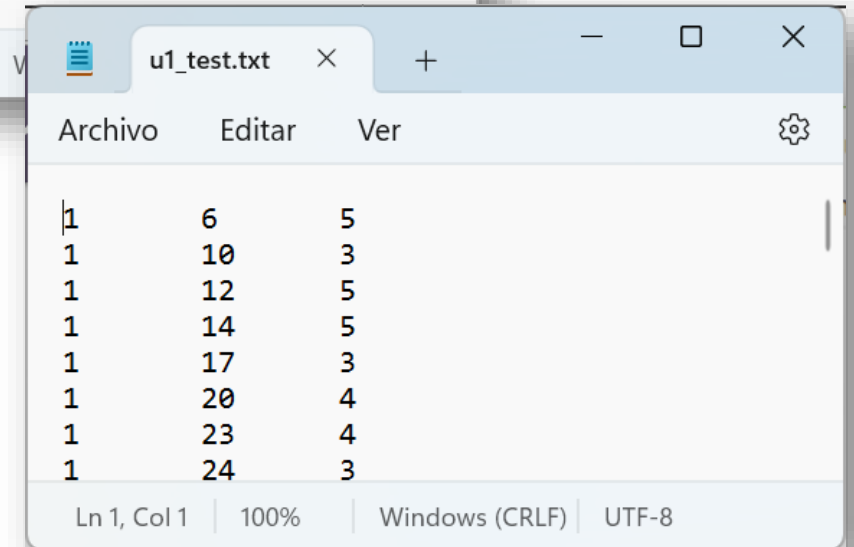
U1_base.txt

U1_test.txt

- Datos de las puntuaciones de los usuarios a las películas
- Contiene
 - Id del usuario
 - Id de la película
 - Ratio dado por el usuario a la película (1..5)



1	1	5
1	2	3
1	3	4
1	4	3
1	5	3
1	7	4
1	8	1
1	9	5



1	6	5
1	10	3
1	12	5
1	14	5
1	17	3
1	20	4
1	23	4
1	24	3

Ficheros del dataset

U1_base.txt U1_test.txt

Se parte el dataset
en dos partes

u1_base.txt

- Contiene los datos con los que se entrena el sistema

u1_test.txt

- Contiene los datos con los que se evalúa el sistema
- Cuando se calcula una recomendación para el usuario se evalúa si está en este fichero y si se ha acertado con el ratio

Estructura de datos: película

Película (ítems.txt)

- Identificador: valor entero
- Título de la película (string)
- Ratio de la película en cada género (0 o 1). Vector de 19 enteros. Indica en qué géneros está clasificada una película

items: Bloc de notas

Archivo	Edición	Formato	Ver	Ayuda																
1	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	Toy Story (1995)
2	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	GoldenEye (1995)
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	Four Rooms (1995)
4	0	1	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	Get Shorty (1995)
5	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	1	0	0	Copycat (1995)
6	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	Shanghai Triad (Yao a yao yao dao wa...
7	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	Top Gun (1986)

Línea 1, columna 1 100% Windows (CRLF) ANSI

Estructura de datos: **perfil de usuario**

Identificador de usuario

Información demográfica

- Información personal: edad, género, profesión

Modelo de preferencias del usuario

- Tres vectores con las preferencias
 - Demográficas
 - Basadas en contenido
 - Colaborativas

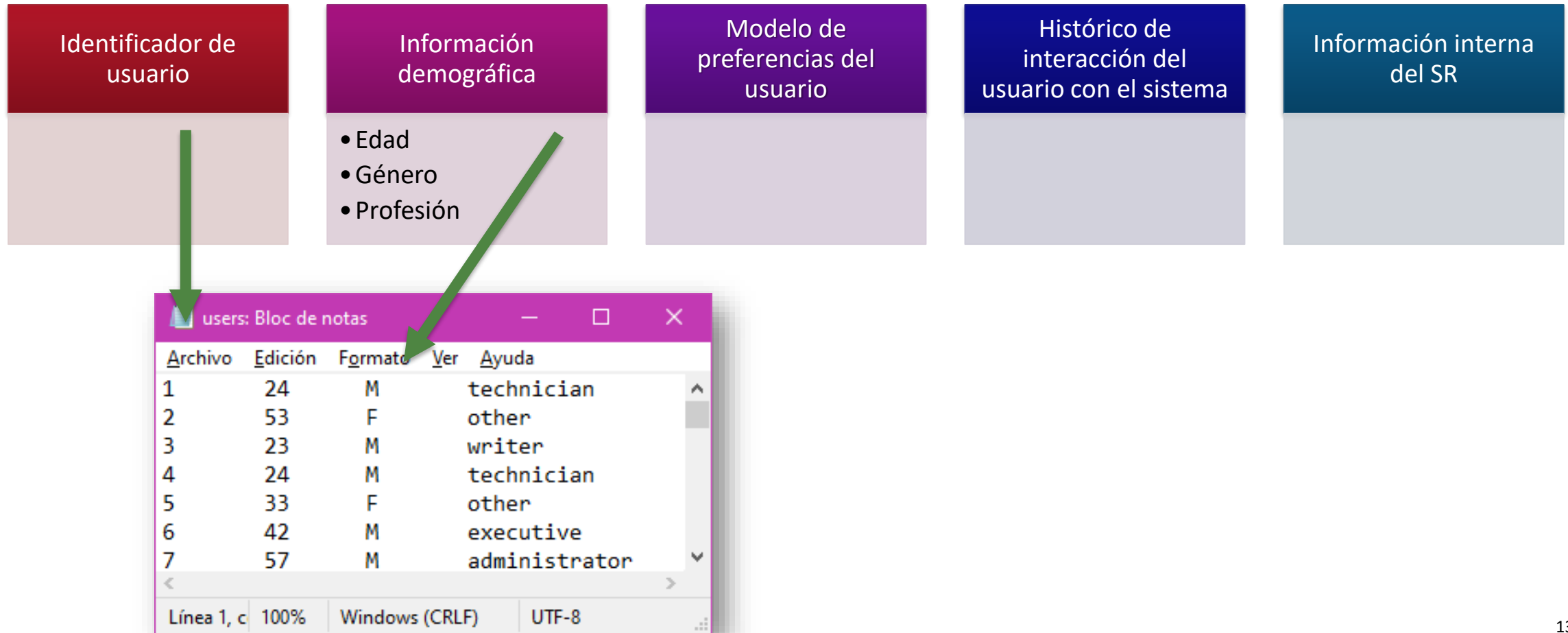
Histórico de interacción del usuario con el sistema

- Conjunto de películas vistas
- Ratio dado a la película

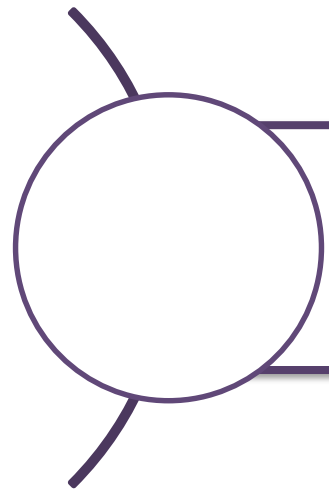
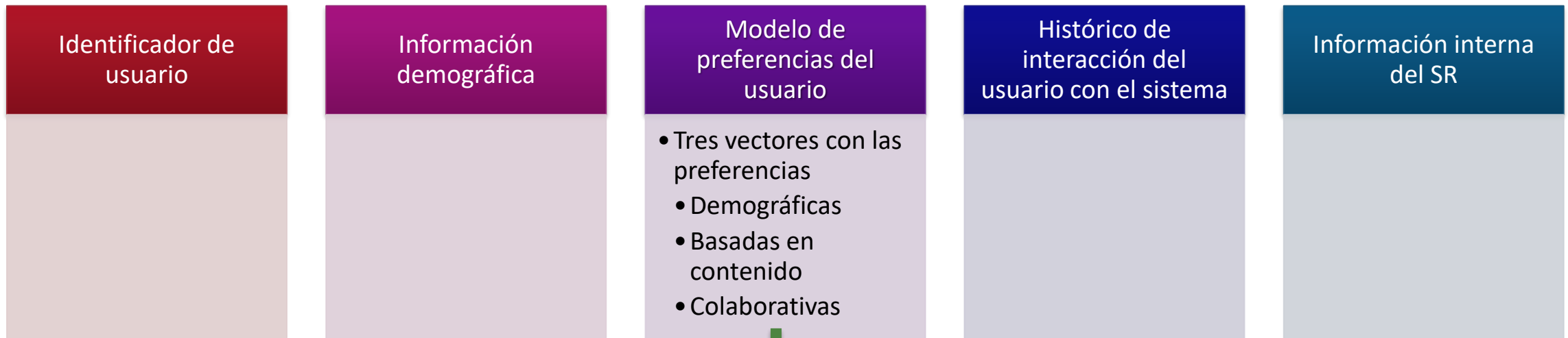
Información interna del SR

- Información que el SR calcula y que facilita el proceso de recomendación
- Clasificación del usuario, vecinos,...

Estructura de datos: **perfil de usuario**



Estructura de datos: **perfil de usuario**



Modelo de preferencias del usuario

- Predicción de lo que a cada usuario le interesará cada uno de los 19 géneros

Estructura de datos: **perfil de usuario**

Modelo de preferencias del usuario

- Habrá 3 modelos de preferencias (uno para cada técnica de recomendación básica o BRT):
 - Preferencias demográficas
 - Preferencias basadas en contenido
 - Preferencias colaborativas: depende de cómo se implemente la técnica colaborativa

```
int preferencias_demograficas[NUM_GENEROS];  
int preferencias_basadas_contenido[NUM_GENEROS];  
int preferencias_colaborativas[NUM_GENEROS];
```

Estructura de datos: perfil de usuario

Las preferencias pueden ser un vector, o cualquier estructura, donde, para cada género de película, se almacene el ratio de interés para el usuario en ese género (**0..100**)

Preferencias demográficas y basadas en contenido

- Se recomienda que de las 19 posiciones, **solo 5 o 6 tengan un valor**. El resto es aconsejable que estén a 0

Preferencias colaborativas

- Aconsejable que la mayor parte de las posiciones del vector tengan valor

0	20	55		90
0	1	2	...	18

genre: Bloc ...

Archivo Edición Formato Ver Ayuda

0	unknown
1	Action
2	Adventure
3	Animation
4	Children's
5	Comedy
6	Crime
7	Documentary
8	Drama
9	Fantasy
10	Film-Noir
11	Horror
12	Musical
13	Mystery
14	Romance
15	Sci-Fi
16	Thriller
17	War
18	Western

Estructura de datos: perfil de usuario

Ejemplo

El vector muestra las preferencias de un usuario

El usuario tiene un interés de **80** (sobre 100) en las películas de acción

Un interés de **30** en las películas de aventuras

Un interés de **60** en las películas de comedia

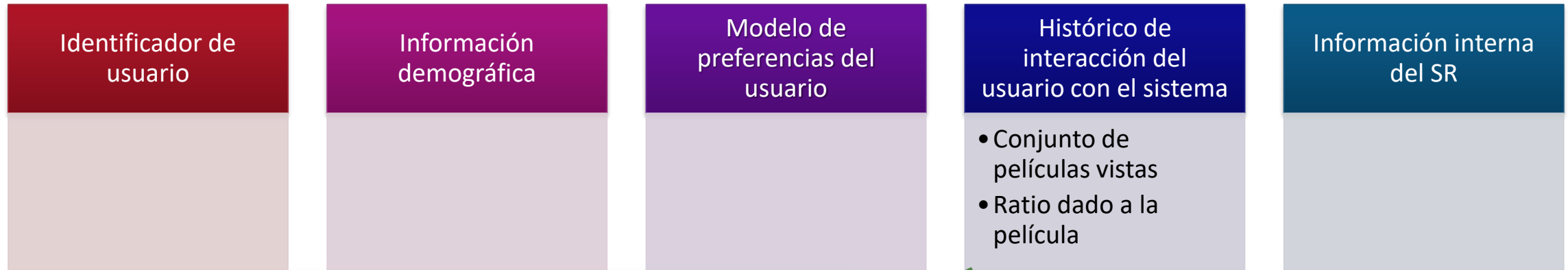
Un interés de **100** en las películas de crímenes

Un interés de **50** en las películas del oeste

0	80	30	0	0	60	100	0		50
0	1	2	3	4	5	6	7	...	18

genre: Bloc ...		—	□	×
Archivo	Edición	Formato	Ver	Ayuda
0	unknown			
1	Action			
2	Adventure			
3	Animation			
4	Children's			
5	Comedy			
6	Crime			
7	Documentary			
8	Drama			
9	Fantasy			
10	Film-Noir			
11	Horror			
12	Musical			
13	Mystery			
14	Romance			
15	Sci-Fi			
16	Thriller			
17	War			
18	Western			

Estructura de datos: **perfil de usuario**



u1_base.txt

Archivo	Editar	Ver
1	1	5
1	2	3
1	3	4
1	4	3
1	5	3
1	7	4
1	8	1
1	9	5

Ln 1, Col 1 | 100% | Windows (CRLF) | UTF-8

Estructura de datos: **perfil de usuario**

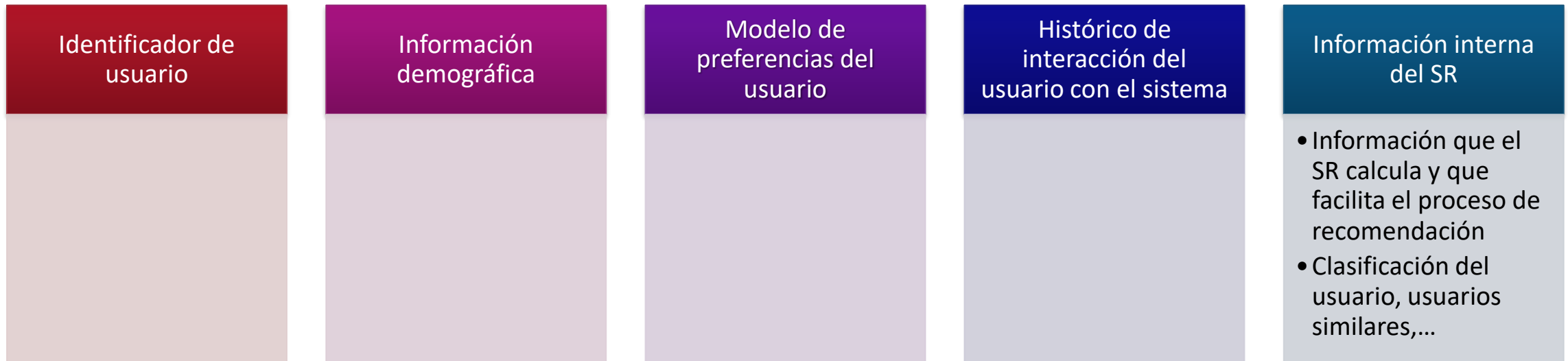
Histórico

- Contiene las películas vistas por el usuario y la puntuación dada a cada una de ellas
- Nº de películas vistas
- Para cada una de ellas:
 - Identificador
 - Puntuación dada
- La puntuación en el fichero está entre 1 y 5. Aconsejable convertirla en valores entre 1 y 100, por homogeneidad

```
// Item en el histórico del usuario  
typedef struct  
{  
    int id_item, ratio;  
} item_historico;
```

```
// Histórico de películas vistas  
int num_historico;  
item_historico historico[NUM_HISTORICO];
```

Estructura de datos: **perfil de usuario**



Información interna del SR

- Los SR necesitan almacenar información para posteriormente, calcular la recomendación
- Se va a utilizar en el SR Colaborativo
 - Contiene la información de los usuarios que son vecinos del usuario actual (usuarios con mayor grado de afinidad)

Estructura de datos: **perfil de usuario**

Información interna del SR Colaborativo: vecinos

Número de vecinos del usuario

Para cada vecino

Identificador


Grado de afinidad con el vecino (valor entre 0 y 100)

```
// Vecino
typedef struct
{
    int id_vecino, afinidad;
} vecino;
```



```
// Recomendación colaborativa
int num_vecinos;
vecino vecinos[MAX_VECINOS];
```

Lectura de datos



Leer los ficheros de datos y almacenarlos en las estructuras de datos creadas, para así poder comenzar la recomendación

Resultado de la recomendación

Lista de ítems recomendados

- Lista que contiene todas las películas que el usuario no ha visto (no están en su histórico)
- Para cada película contiene:
 - Identificador de la película
 - Ratio de interés del usuario en la película (calculado por la recomendación)
- La lista debe estar ordenada por ratio de interés (de mayor a menor) al finalizar el proceso de recomendación
- La interfaz de la aplicación mostrará por pantalla sólo los N primeros elementos de la lista (o lo que hayáis decidido)

```
// Item en la lista de items recomendados  
typedef struct  
{  
    int id_item, ratio;  
} item_recomendado;
```