UNIVERSIDADE DA CORUÑA

MUEI Mestrado Univer. en Enxeñaría Informática

RIWS - Recuperación de la Información

Recomendación con algoritmo de filtrado colaborativo

W. Joel Castro Reynoso

17/12/2013

Introducción

En este documento se describen y explican los pasos seguidos para la resolución de la práctica de algoritmos de recomendación realizada para la asignatura **Recuperación de la Información y Web Semántica**, en concreto para la parte de Recuperación de la Información. El objetivo principal de esta parte es dar a conocer los métodos y técnicas relacionadas con la estructura, análisis, organización, almacenamiento, búsqueda y recuperación de la información, entre los que se incluyen sistemas de búsqueda (***Search Engines***) y sistemas de recomendación (***Recommender Systems***), y proporcionar una visión de las posibilidades de utilización de dicho métodos en el ámbito empresarial.

La práctica se basa principalmente en la aplicación de un algoritmo de filtrado colaborativo (CF) para generar recomendaciones de películas previamente valoradas y con cuya valoración hemos generado nuestro conjuntos “Training” y “Test”.

El algoritmo implementado ha sido el algoritmo de filtrado colaborativo **kNN** (*k Nearest Neighbors*) y las pruebas se han basado en la ejecución del mismo con distintos parámetros y funciones de similitud (similarity func), entre las que se encuentran la función Coseno y la Correlación de Pearson, para luego evaluar el rendimiento de cada ejecución mediante el cálculo del **MAE** (*Mean Absolute Error*, o Error Medio Absoluto) y el **RMSE** (*Root Mean Squared Error*, o Raíz del error cuadrático de la media).

El error cuadrático medio (RMSE) nos da la media de las diferencias en promedio entre los valores pronosticados y los observados. El error medio absoluto (MAE) nos da una información similar. Ambos tienen la ventaja de que solo manejan los valores del error y no su signo, MSE usando valores absolutos -más interpretables- y RMSE usando el cuadrado para evitar los problemas de signo, con lo que se hace más cómodo su uso desde el punto de vista matemático y optimización estadística.

La implementación se ha realizado en Java, utilizando como base la estructura basada en comandos desarrollada para el resto de prácticas de la asignatura.

El comando para su ejecución es:
prompt$ java riws rate *fichero_excel_ratings knumber*

Argumentos:

- riws: ejecutable
- rate: comando específico para esta práctica. Para ver la lista completa de comandos ejecutar “help”
- fichero_excel_ratings: fichero excel con la información de las evaluaciones (training y test)
- knumber: parámetro opcional. Indica el número de vecinos (neighbors) tomados para la recomendación. En caso de omitir este parámetro, se tomarán como vecinos todos los datos del conjunto.

El código fuente, tanto de esta práctica como de toda las prácticas de la asignatura, se puede encontrar en Github: https://github.com/jdell/muei

Técnicas aplicadas

El algoritmo utilizado en esta práctica para las recomendaciones es el algoritmo de filtrado colaborativo kNN (k Nearest Neighbors) basado en la similitud entre usuarios. Los sistemas de recomendación basados en algoritmos de filtrado colaborativo utilizan las valoraciones de los usuarios sobre ciertos elementos del conjunto total para predecir valoraciones en el resto de los elementos y recomendar los de mayor valoración predicha. En nuestro caso, hemos partido de un conjunto de elementos ("Training") con la información proporcionada por los estudiantes y hemos utilizados estos datos para recomendar películas a los elementos indicados en el segundo conjunto ("Tests").

El algoritmo kNN es un algoritmo CF basado en memoria en el que se utiliza toda la base de datos de elementos y usuarios para generar predicciones. Una vez que se ha construido una lista de vecinos se combinan sus preferencias para generar una lista con los N elementos más recomendables para el usuario actual. Entre sus inconvenientes se encuentra la necesidad de disponer de un número míinimo de usuarios con un número mínimo de predicciones cada uno, incluido el usuario para el que se pretende realizar la recomendación (problema de falta de vecinos o vecinos poco fiables).

Para las funciones de similitud se han implementado:

- **Coseno**. En esta se considera cada elemento como un vector de un espacio vectorial de m dimensiones y se calcula la similitud como el coseno del ángulo que forman.

$$sim(u,v) = \sum_{\substack{i:r(u,i)\neq\emptyset \\ r(v,i)\neq\emptyset}} r(u,i)r(v,i) \Big/ \sqrt{\sum_{\substack{i:r(u,i)\neq\emptyset \\ r(v,i)\neq\emptyset}} r(u,i)^2 \sum_{\substack{i:r(u,i)\neq\emptyset \\ r(v,i)\neq\emptyset}} r(v,i)^2} \quad \in [0,1]$$

Similarity function: Cosine

- **Correlación de Pearson**. Esta se deriva de las fórmulas de regresión lineal, y asume que la relación entre elementos es lineal, los errores independientes y la distribución tiene varianza constante y media 0. Estas suposiciones normalmente no se producen realmente con lo que hay que valorar cómo afectan a la bondad de los resultados, pero en un gran número de casos el el rendimiento utilizando Pearson es apropiado.

$$sim(u,v) = \sum_{\substack{i:r(u,i)\neq\emptyset \\ r(v,i)\neq\emptyset}} (r(u,i) - \overline{r}_u)(r(v,i) - \overline{r}_v) \Big/ \sqrt{\sum_{\substack{i:r(u,i)\neq\emptyset \\ r(v,i)\neq\emptyset}} (r(u,i) - \overline{r}_u)^2 \sum_{\substack{i:r(u,i)\neq\emptyset \\ r(v,i)\neq\emptyset}} (r(v,i) - \overline{r}_v)^2}$$

$\in [-1,1]$

$\overline{r}_u$ ↑ Puntuación promedio de u

Similarity function: Pearson Correlation Coefficient

Resultados

Se han lanzado 8 ejecuciones con distintas configuraciones para el algoritmo kNN:

- Utilizando la versión básica (centered=false)
- Utilizando la versión centrada en la media (centered=true)
- Utilizando todo el conjunto de datos como vecinos (k=all)
- Utilizando un conjunto constante de datos como vecinos (k=15)

Y combinandolas con las funciones de similitud (Coseno y Corr. Pearson), se han obtenido los siguientes resultados (en azul y verde los mejores resultados):

```
KNN (centered: false, k:   all) - Sim:           Cosine - MAE:  0.846 - RMSE:  1.119
KNN (centered:  true, k:   all) - Sim:           Cosine - MAE:  0.795 - RMSE:  1.038
KNN (centered: false, k:    15) - Sim:           Cosine - MAE:  0.875 - RMSE:  1.161
KNN (centered:  true, k:    15) - Sim:           Cosine - MAE:  0.816 - RMSE:  1.073
KNN (centered: false, k:   all) - Sim: Pearson Correl. - MAE:  1.608 - RMSE:  2.180
KNN (centered:  true, k:   all) - Sim: Pearson Correl. - MAE:  0.783 - RMSE:  1.035
KNN (centered: false, k:    15) - Sim: Pearson Correl. - MAE:  1.793 - RMSE:  2.428
KNN (centered:  true, k:    15) - Sim: Pearson Correl. - MAE:  0.845 - RMSE:  1.131
Elapsed time: 1756 ms
```

La siguiente tabla muestra los datos de la primera ejecución (Cosine, false, all -232 estimaciones-):

0	10	4.00	3.81	8	5	4.00	4.27	17	5	5.00	4.23
0	32	4.00	4.48	8	10	4.50	3.82	17	26	5.00	1.80
0	34	3.00	3.01	8	27	4.00	3.72	17	34	2.00	3.13
0	36	3.50	3.30	8	28	2.00	2.71	17	36	4.00	3.30
0	42	4.00	3.59	8	29	3.50	3.16	17	59	4.00	3.75
0	43	3.50	3.54	8	45	4.00	3.70	17	60	3.00	3.59
0	45	4.50	3.67	8	51	3.00	3.60	17	81	4.00	2.59
0	47	3.00	3.63	8	75	2.50	3.86	18	0	4.00	3.97
0	49	4.50	4.30	9	0	4.00	3.91	18	36	3.00	3.37
0	52	4.00	4.09	9	18	4.00	4.34	18	63	4.00	3.16
0	65	1.00	3.97	9	28	3.00	2.69	19	23	2.00	3.04
0	67	4.50	4.11	9	30	2.00	3.34	19	28	3.00	2.63
0	76	3.50	3.40	9	34	2.00	3.08	19	70	2.50	3.74
0	82	4.00	1.75	9	67	3.00	4.10	19	75	4.00	3.70
0	86	4.00	3.79	10	4	3.00	3.00	20	15	4.00	4.52
1	21	4.00	3.50	10	8	4.00	3.23	20	19	3.50	4.15
1	24	3.00	3.09	10	20	5.00	4.34	20	36	2.50	3.32
1	40	4.00	3.11	10	23	5.00	3.08	20	37	4.50	4.48
1	49	5.00	4.32	10	28	4.00	2.68	20	39	4.00	3.42
1	63	4.00	3.14	10	34	3.50	3.10	20	43	4.50	3.56
1	66	3.00	3.43	10	47	1.00	3.71	20	50	3.50	3.93
1	69	4.00	3.46	10	50	4.00	3.93	20	59	4.50	3.69
1	70	5.00	3.67	10	99	4.00	2.45	20	60	3.00	3.72
1	83	3.00	2.43	11	8	4.00	3.32	20	67	4.00	4.11
1	95	3.00	2.82	11	9	4.00	4.24	20	89	2.50	2.25
2	21	3.50	3.51	11	29	4.00	3.16	20	95	2.50	2.86
2	30	3.00	3.32	11	49	3.50	4.32	21	3	5.00	3.99
2	38	2.50	3.25	11	51	4.00	3.59	21	10	3.00	3.75
2	53	2.50	3.66	11	52	4.00	4.08	21	15	1.00	4.51
2	54	3.50	4.00	11	67	4.00	4.10	21	19	1.00	4.05
2	63	5.00	3.15	11	68	3.50	3.56	21	21	1.00	3.50
2	65	5.00	4.02	11	76	5.00	3.38	21	23	4.00	3.12

3	15	4.00	4.51
3	42	4.00	3.60
3	70	4.00	3.76
4	3	3.00	3.96
4	5	5.00	4.24
4	10	3.00	3.81
4	21	4.00	3.54
4	24	3.00	3.10
4	27	2.00	3.72
4	29	1.00	3.18
4	36	3.00	3.29
4	40	2.00	3.23
4	43	2.00	3.59
4	57	3.00	3.50
4	58	2.00	3.54
4	67	4.00	4.08
4	68	2.00	3.66
4	76	3.00	3.40
4	88	3.00	3.00
5	25	4.00	4.03
5	54	4.00	4.01
5	75	3.00	3.81
5	76	5.00	3.46
6	23	2.50	3.07
6	32	4.00	4.44
6	36	3.50	3.30
6	50	4.00	3.85
6	60	4.00	3.71
6	65	3.00	3.90
6	85	3.50	1.00
6	95	3.50	2.81
6	99	3.50	2.42
7	4	2.00	2.97
7	19	4.00	4.07
7	22	5.00	4.04
7	24	1.00	3.04
7	31	5.00	3.54
7	35	2.00	3.26
7	40	4.00	3.08
7	43	3.00	3.53
7	44	4.00	2.92
7	58	1.50	3.53
7	67	5.00	4.08
7	68	2.00	3.51
7	70	4.50	3.69
7	75	3.00	3.77
11	80	3.50	3.45
11	89	3.50	2.23
12	0	4.00	3.94
12	36	3.00	3.36
12	37	5.00	4.51
12	52	4.00	4.12
12	61	3.50	3.32
12	70	3.00	3.71
13	25	5.00	4.11
13	37	5.00	4.50
13	63	5.00	3.17
14	20	5.00	4.39
14	22	4.00	4.09
14	29	3.00	3.18
14	39	5.00	3.46
14	41	4.00	3.47
14	50	3.00	3.93
14	57	4.00	3.55
14	59	5.00	3.79
14	77	5.00	3.36
15	9	2.50	4.21
15	16	3.00	2.45
15	18	2.50	4.40
15	20	2.50	4.35
15	24	1.00	3.11
15	32	1.50	4.44
15	35	1.00	3.30
15	41	4.00	3.52
15	46	4.00	3.87
15	48	3.00	3.50
15	60	3.00	3.66
15	71	1.50	5.00
15	80	1.00	3.58
15	81	5.00	2.77
15	85	1.00	1.00
15	87	1.00	3.49
15	93	5.00	3.49
15	96	2.50	1.00
16	5	5.00	4.25
16	9	4.00	4.21
16	19	5.00	4.11
16	27	4.00	3.73
16	33	4.50	4.50
16	63	3.50	3.13
16	74	4.00	3.50
16	75	4.00	3.80
21	40	5.00	3.17
21	41	1.00	3.52
21	47	2.00	3.67
21	54	5.00	3.99
21	56	5.00	3.73
21	61	2.00	3.25
21	63	1.00	3.09
21	76	3.00	3.38
21	77	2.00	3.39
21	86	4.00	3.84
21	92	2.00	3.48
21	97	1.00	1.00
22	18	4.00	4.38
22	24	2.00	3.04
22	35	4.00	3.28
22	36	4.00	3.30
22	38	3.00	3.11
22	40	3.00	3.17
22	50	3.00	3.91
22	63	4.00	3.17
22	64	3.00	3.93
22	70	3.00	3.70
22	75	3.00	3.78
22	86	4.00	3.78
23	0	5.00	3.95
23	3	4.00	3.99
23	18	5.00	4.38
23	24	4.00	3.13
23	28	5.00	2.65
23	29	3.00	3.18
23	32	5.00	4.47
23	45	3.00	3.68
23	51	3.00	3.53
23	53	3.00	3.70
23	72	3.00	2.26
23	80	4.00	3.44
23	93	5.00	3.52
24	20	5.00	4.34
24	23	3.00	3.04
24	24	3.00	3.11
24	28	4.00	2.68
24	38	4.00	3.20
24	70	4.00	3.71
24	77	4.00	3.38

Correspondiéndose con:

- columna 1: usuario objeto de recomendación
- columna 2: ítem película
- columna 3: valor del conjunto Test
- columna 4: valor recomendado

Como podemos apreciar, las ejecuciones que incluyen la variante centrada en la media dan mejores resultados. Y en nuestro ejemplo, dentro de esas ejecuciones las medidas de error (MAE y RMSE) son inversamente proporcionales al número de vecinos utilizados. Con un conjunto de datos más grande, podríamos utilizar variante basada en el Top de vecinos (similitud por encima del 0.75, por ejemplo) en vez de un conjunto de tamaño constante de vecinos (K constante) y evaluar nuevamente los algoritmos utilizados.