EVALUACIÓN DE UN MÉTODO DE PONDERACIÓN DE ATRIBUTOS MULTIVALUADOS EN SISTEMAS DE RECOMENDACIÓN BASADOS EN CONTENIDO

Manuel J. Barranco[1], **Jorge Castro**[1], **Luis Martínez**[1]

[1] *Universidad de Jaén, Campus Las Lagunillas, 23071, Jaén*, {barranco,jcastro,martin}@ujaen.es

Resumen

Los sistemas de recomendación basados en contenido (SRBC) junto con los sistemas de filtrado colaborativo son los más utilizados en el ámbito del comercio electrónico. Los SRBC, basándose en información histórica, construyen un perfil para cada usuario, que se compara con los productos para recomendar los que mejor se ajusten al perfil. En este proceso es de gran interés saber qué atributos o características descriptivas de los productos resultan más importantes para cada usuario, de manera que el sistema puede dar mayor peso a dichos atributos. El método TF-IDF es uno de los más utilizados para la ponderación de los atributos, sin embargo cuando éstos toman valores en dominios multivaluados dicho método podría mejorarse. En esta contribución se propone un método de ponderación basado en la entropía y los coeficientes de correlación y de contingencia con el fin de mejorar el filtrado basado en contenido en entornos con atributos de múltiples valores.

Palabras Clave: Atributos multivaluados, Sistemas de Recomendación basados en contenido, Ponderación de atributos.

1 INTRODUCCIÓN

Los sistemas de recomendación son herramientas que ayudan a los usuarios en situaciones en que pueden sentirse desbordados por la información, por lo que han sido utilizados ampliamente en el ámbito del comercio electrónico [11]. En particular, los sistemas de recomendación basados en contenido (SRBC) [1, 8, 9], usados tradicionalmente, utilizan información disponible sobre las elecciones realizadas por el usuario en el pasado, para construir su perfil que representa sus preferencias o necesidades.

Las funciones básicas de un SRBC consisten en (i) actualizar el perfil de cada usuario, (ii) comparar el perfil con los productos disponibles y (iii) recomendar los productos que mejor encajan en el perfil. En este proceso, al comparar el perfil con los productos, se debería tener en cuenta que no todos los atributos son igualmente importantes. Normalmente, cuando un usuario elige un producto se está fijando en algunos rasgos o atributos del mismo que son importantes para él/ella e ignorando otros que no le resultan de interés. Esta consideración representa una ponderación implícita sobre el conjunto de atributos de manera que cada atributo tendrá un peso subjetivo y diferente para cada usuario.

Típicamente, un SRBC trabaja con análisis textual de modo que los atributos son palabras *relevantes* que aparecen en la descripción de los productos. De esta manera cada producto va a estar descrito por una tupla de unos y ceros que indica si una palabra-atributo aparece o no en la descripción textual del producto. Sin embargo, en un caso más general, los atributos pueden estar definidos en diferentes dominios de distintos tipo: numérico, linguístico o nominal, etc.

El propósito de esta contribución es presentar un método novedoso para obtener los pesos de los atributos usando las valoraciones implícitas, que pueden obtenerse de las elecciones realizadas por los usuarios en el pasado, asignando pesos a los atributos según la ponderación del usuario. Esta ponderación sobre el conjunto de atributos se basará en dos medidas, *correlación* (para atributos cuantitativos) y *contingencia* (para atributos cualitativos). Además de estas dos medidas, también se utilizará la entropía o cantidad de información de cada atributo. Cuanto mayor sea la entropía de un atributo, más información ofrece al sistema y por tanto mayor ponderación en el proceso de filtrado.

En la siguiente sección revisaremos algunos conceptos preliminares en los que se basa el trabajo realizado. Tras ello presentaremos el método propuesto y finalmente se hará una evaluación del mismo, terminando con unas conclusiones.

2 PRELIMINARES

Esta sección revisa los SRBC y los métodos más utilizados para ponderar características o atributos.

2.1 SISTEMAS DE RECOMENDACIÓN BASADOS EN CONTENIDO

En un sistema de recomendación basado en contenido [1, 8, 9], el punto de partida es un conjunto de productos $A = \{a_i, i = 1 \dots n\}$ que pueden ser recomendados y un conjunto de características o atributos que los describen $C = \{c_j, j = 1 \dots m\}$ definidos cada uno en un dominio D_j. De este modo, cada producto a_i queda descrito por un vector $V_i = \left\{ v_j^i \in D_j, j = 1 \dots m \right\}$. El sistema almacena esta información en una base de datos mediante una tabla de doble entrada (ver Tabla 1).

Tabla 1: Datos de un SRBC

	c_1	$\dots$	c_j	$\dots$	c_m
a_1	v_1^1	$\dots$	v_j^1	$\dots$	v_m^1
$\dots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$
a_n	v_1^n	$\dots$	v_j^n	$\dots$	v_m^n

Para cada usuario, u, existe un subconjunto $A_u = \{a_{x_{ui}} \in A\,,\ 1 \leq x_{ui} \leq n\ ,\ i = 1 \dots nu\ ,\ nu \leq n\}$ de nu productos que hayan sido elegidos por él/ella, y para cada producto, $a_{x_{ui}}$, se asocia una valoración de preferencia, $r_i^u \in D_u$, siendo D_u el dominio de valoraciones del usuario (ver Tabla 2). Usando la información del usuario, el SRBC obtiene un perfil de usuario P_u que representa sus preferencias para cada atributo, y un vector de pesos W_u que incluye los pesos de cada atributo de acuerdo a la relevancia en las necesidades o preferencias del usuario (última fila en Tabla 2):

- $P_u = \left\{ p_j^u \in D_j, j = 1 \dots m \right\}$ es el perfil de usuario, es decir, los valores de cada atributo que mejor se ajustan a las preferencias del usuario. Pueden obtenerse de distintas maneras [1, 8, 9].
- $W_u = \left\{ w_j^u, j = 1 \dots m, 0 \leq w_j^u \leq 1 \right\}$ son los pesos que muestran la relevancia de cada atributo, de acuerdo a las necesidades del usuario.

Tabla 2: Datos de usuarios en un SRBC

	c_1	$\dots$	c_m	R_u
$a_{x_{u1}}$	$v_1^{x_{u1}}$	$\dots$	$v_m^{x_{u1}}$	r_1^u
$\dots$	$\dots$	$\dots$	$\dots$	$\dots$
$a_{x_{u,nu}}$	$v_1^{x_{u,nu}}$	$\dots$	$v_m^{x_{u,nu}}$	r_{nu}^u
P_u	p_1^u	$\dots$	p_m^u	
W_u	w_1^u	$\dots$	w_m^u	

Un SRBC se divide habitualmente en dos subsistemas:

- Subsistema off-line: Básicamente se encarga del mantenimiento de la base de datos:
  1. Actualización de la base de datos de productos. Un grupo de expertos participarán en la incorporación de la información que describe los elementos del sistema.
  2. Actualización de los perfiles de usuario. El sistema actualiza los perfiles de usuario basándose en la información implícita obtenida en el proceso on-line, mediante la observación de las interacciones del usuario con los productos.
- Subsistema on-line: Asiste al usuario filtrando y recomendando los productos que mejor se adapten a su perfil:
  1. Cálculo de similitudes. Para cada producto el sistema calcula su similitud con el perfil de usuario.
  2. Recomendación. El sistema selecciona los productos con mayor grado de similitud y, a continuación, se recomiendan al usuario los N mejores.

2.2 PONDERACIÓN DE ATRIBUTOS EN SRBC

En la literatura encontramos distintos métodos de ponderación de atributos en SRBC que trabajan con palabras, es decir, los atributos son palabras clave que describen los productos [9, 13]. A continuación se describe brevemente el funcionamiento de tales sistemas.

Inicialmente se construye un perfil de usuario mediante el uso de las valoraciones implícitas obtenidas a partir de los productos adquiridos o puntuados por el usuario. Los valores de los atributos dependerán de la aparición de ciertas palabras o términos linguísticos en la descripciones de los productos. De este modo, dado un atributo definido por una palabra y dado un producto, el perfil del mismo puede tomar dos valores: 1 si dicha palabra aparece en la descripción o 0 en caso contrario.

En el proceso de filtrado, el SRBC busca los productos más adecuados para el usuario, comparando el perfil del usuario y las descripciones de los productos. Una mejora consiste en dar mayor peso a aquellos atributos que se consideren más relevantes para el usuario. El método TF-IDF (Term-Fequency - Inverse Document Frequency) [2, 13] calcula el grado de relevancia de cada atributo c_j para un determinado usuario mediante la siguiente expresión:

$$W(u, c_j) = FF(u, c_j) * IUF(c_j) \tag{1}$$

La relevancia del atributo c_j para el usuario u se obtiene multiplicando dos factores: (i) Una medida de la similitud intra-usuario, FF (Feature Frecuency o frecuencia de atributo), que indica la frecuencia del atributo c_j para el usuario u y (ii) una medida de la disimilitud inter-usuarios, IUF

(Inverse User Frequency o frecuencia inversa de usuario), que ofrece un mayor peso a los atributos distintivos, es decir, a los que menos se repiten en el conjunto de usuarios.

Comunmente, el factor $FF(u,c_j)$, se obtiene sumando el número de veces que el atributo c_j aparece en los productos que el usuario u ha valorado positivamente. El segundo factor, de acuerdo al esquema TF-IDF [2], se obtiene como $IUF(c_j) = log\frac{|U|}{UF(c_j)}$ siendo $UF(c_j)$ el número de usuarios que han valorado positivamente cualquier producto que posea el atributo c_j, y $|U|$ el número total de usuarios registrados en el sistema. Este método de ponderación resulta muy adecuado en SRBC que trabajen con atributos binarios: palabras que aparecen o no en las descripciones textuales de los productos. Sin embargo, en aquellos sistemas que tratan con descripciones más complejas, con atributos multivaluados, el enfoque anterior no resulta apropiado.

El problema de ponderación de atributos multivaluados ha sido tratado en áreas como las de recuperación de información y aprendizaje automático [6, 7]. Sin embargo, este problema ha sido abordado escasamente en el área de los sistemas de recomendación. Nuestra intención es ofrecer una propuesta que trate este asunto de manera satisfactoria en SRBC.

3 PONDERACIÓN DE ATRIBUTOS BASADA EN LA ENTROPÍA Y MEDIDAS DE DEPENDENCIA

Nuestro objetivo en esta contribución es aplicar y evaluar un método de ponderación de atributos en SRBC, basado en [3], que trate con atributos multivaluados. En esta propuesta se calcula un peso para cada atributo atendiendo a:

1. Disimilitud inter-usuario: se calcula qué atributos resultan más informativos y por tanto más relevantes en el proceso de filtrado. Para este cálculo se propone el uso de la entropía. Un atributo con mayor entropía (mayor cantidad de información) resulta más útil como elemento distintivo de los gustos o preferencias de los usuarios.
2. Similitud intra-usuario: se calcula un coeficiente para medir la correlación entre las valoraciones realizadas por el usuario en el pasado y los valores de los atributos en el conjunto de productos. Para llevar a cabo este cálculo usamos dos coeficientes, dependiendo de la naturaleza de los atributos (cuantitativa o cualitativa).

El método propuesto usa la estructura de datos mostrada en la sección anterior (ver Tablas 1 y 2). Para nuestra propuesta vamos a considerar dos familias de vectores:

- Para cada producto $a_{x_{ui}}$ elegido por el usuario, $V^u_{i.} = \left\{v^{x_{ui}}_j, j = 1\ldots m\right\}$ proporciona su descripción, es decir, los valores de cada atributo para dicho producto.
- Para cada atributo c_j, $V^u_{.j} = \left\{v^{x_{ui}}_j, i = 1\ldots nu\right\}$ ofrece los valores de dicho atributo para cada uno de los productos que el usuario ha valorado en el pasado, $a_{x_{ui}}$.

El proceso completo de ponderación de atributos basado en la entropía y medidas de dependencia consta de las siguientes fases:

1. Cálculo de la disimilitud inter-usuario. Para cada atributo c_j, se cálcula la entropía H_j o cantidad de información que dicho atributo puede ofrecer.
2. Cálculo de la similitud intra-usuario. Para cada atributo c_j, dado el usuario u podemos calcular un coeficiente de dependencia, DC_{uj}, entre las valoraciones obtenidas por parte del usuario, $R_u = \{r^u_i, i = 1\ldots nu\}$, y los valores del atributo en los productos valorados, $V^u_{.j} = \left\{v^{x_{ui}}_j, i = 1\ldots nu\right\}$.
3. Cálculo de pesos. Finalmente se obtienen los pesos de los atributos como resultado de multiplicar la entropía y el grado de dependencia.

En las siguientes secciones se exponen brevemente cada una de las fases del método, que de forma esquemática puede verse en la figura 1. Para un mayor detalle puede consultarse [3].

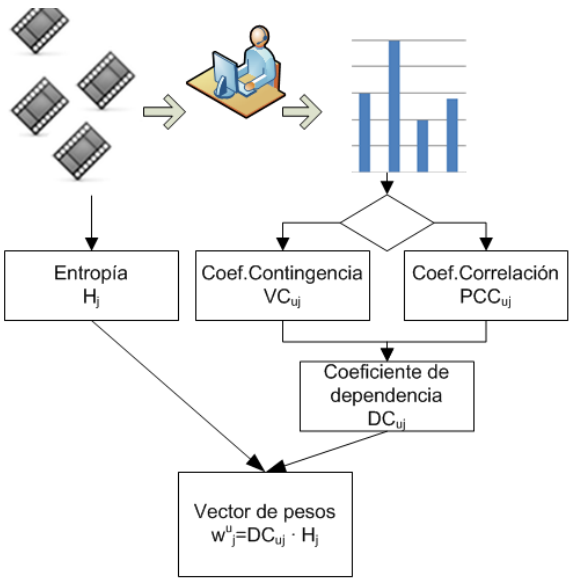

Figura 1: Ponderación de atributos basada en la entropía y medidas de dependencia

3.1 DISIMILITUD INTER-USUARIO

Para calcular la cantidad de información que proporciona cada atributo se propone el uso de la *entropía de la información* [5, 12]. Se define como la cantidad media de información, medida en bits, que contiene una variable aleatoria. Dada una variable aleatoria x, con una función de distribución de probabilidad $p(x)$, su entropía viene dada por:

$$H(x) = -\sum_i p(x_i) log_2(p(x_i)) \tag{2}$$

En el proceso de búsqueda de productos similares a un perfil de usuario dado, los atributos que tengan una entropía mayor van a resultar más interesantes y tendrán más peso en dicho proceso. Por ejemplo, en la tabla 3 tenemos dos atributos con distinta entropía, c_1 definido en el dominio $D_1 = \{A, B\}$ y c_2 en $D_2 = \{1,2,3,4,5,6\}$. Calculando las probabilidades de cada valor, dadas por su frecuencia relativa en el conjunto de datos conocidos, y aplicando la fórmula (2) tenemos que:

$$\begin{array}{l} H(c_1) = 1,\ dado\ que\ p(A) = p(B) = \frac{1}{2} \\ H(c_2) = 2.5,\ dado\ que\ p(1) = p(2) = p(3) = p(6) = \frac{1}{8} \\ \qquad\qquad y\ p(4) = p(5) = \frac{2}{8} \end{array}$$

Por tanto, el atributo c_2, con mayor entropía, está aportando más información al sistema y le daremos más peso en el proceso de filtrado.

Tabla 3: Ejemplo: dos atributos con distinta entropía

	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8
c_1	A	B	B	A	B	A	A	B
c_2	1	3	2	4	4	5	6	5

Así pues, para cada atributo c_j el sistema calcula la entropía H_j, y la entropía normalizada $H_j^* \in [0,1]$, como sigue:

$$\begin{array}{l} H_j = -\sum_{k_j}(f_{k_j}/n) Log_2(f_{k_j}/n) \\ H_j^* = \frac{H_j}{\sum_i H_i} \end{array} \tag{3}$$

siendo $\{k_j\}$ el conjunto de valores que el atributo c_j puede tomar, f_{k_j} la frecuencia del valor k_j en todo el conjunto de productos A y n el cardinal de A. Este cálculo considera log 0 = 0, de modo que los valores cuya frecuencia sea 0 no afecten el resultado.

3.2 SIMILITUD INTRA-USUARIO

Dado un conjunto de productos valorados por el usuario y dado un atributo, en esta fase se mide la dependencia entre dichas valoraciones realizadas por el usuario y los valores que toma dicho atributo en el conjunto de productos dado, dependiendo de su naturaleza. Si existe una dependencia entre estas variables, deduciremos que el atributo es importante para el usuario. Proponemos el uso de dos coeficientes de dependencia conocidos: el coeficiente de correlación de Pearson y el coeficiente de contingencia V de Cramer.

Coeficiente de correlación de Pearson [4]. Es un índice estadístico que mide la relación lineal entre dos variables dando una medida independiente de la escala empleada. Se obtiene dividiendo la covarianza por el producto de las desviaciones estándar de ambas variables. En nuestro caso lo aplicamos a las variables R_u y $V_{.j}^u$ obteniendo la siguiente fórmula:

$$PCC_{uj} = \frac{\sum_i r_i^u v_j^{x_{ui}} - \frac{\sum_i r_i^u \sum_i v_j^{x_{ui}}}{nu}}{\sqrt{\left(\sum_i (r_i^u)^2 - \frac{\left(\sum_i r_i^u\right)^2}{nu}\right)} \sqrt{\left(\sum_i \left(v_j^{x_{ui}}\right)^2 - \frac{\left(\sum_i v_j^{x_{ui}}\right)^2}{nu}\right)}} \tag{4}$$

Coeficiente V de Cramer [4]. Es uno de los ratios de contingencia más usados para medir la dependencia entre dos variables aleatorias, X e Y, donde al menos una de las dos es cualitativa. La fórmula para obtener dicho coeficiente sobre nuestros datos es la siguiente:

$$VC_{uj} = \sqrt{\frac{\sum_{k_u}\sum_{k_j} \frac{\left(f_{k_u,k_j} - \frac{f_{k_u} f_{k_j}}{n_u}\right)^2}{\frac{f_{k_u} f_{k_j}}{nu}}}{nu\ \ min\left(|D_u|, |D_j|\right)}} \tag{5}$$

donde k_u y k_j son índices en los conjuntos de datos R_u y $V_{.j}^u$ respectivamente, f_{k_u}, f_{k_j} son las frecuencias de los valores indexados por k_u y k_j respectivamente, y f_{k_u,k_j} es la frecuencia de la ocurrencia simultánea de los dos valores indexados por k_u y k_j.

Así pues, el coeficiente de dependencia, DC, entre las valoraciones realizadas por el usuario u sobre un conjunto de productos, y los valores del atributo j para cada uno de estos productos, viene dado por la siguiente expresión:

$$DC_{uj} = \begin{cases} |PCC_{uj}| & si\ c_j\ es\ cuantitativo \\ VC_{uj} & si\ c_j\ es\ cualitativo \end{cases}$$

3.3 CÁLCULO DE PESOS DE ATRIBUTOS

Una vez que los factores H_j^* y DC_{uj} han sido obtenidos, el sistema va a calcular el peso de cada atributo c_j como producto de ambos factores, de acuerdo a la fórmula (1).

$$w_j^u = DC_{uj} \cdot H_j^* \tag{6}$$

Para normalizar el vector de pesos $\{w_i\}$ debemos satisfacer la propiedad $\sum w_i = 1$, con lo que obtendremos:

$$W_u^* = \left\{ w_j^{*u} = \frac{w_j^u}{\sum_i w_i^u} \mid j = 1, \ldots, m \right\} \quad (7)$$

4 EVALUACIÓN

Para evaluar el método propuesto, se ha implementado y se han obtenido varias medidas usadas habitualmente en la evaluación de algoritmos de recuperación de información. Así mismo, dichas medidas se han aplicado a otras implementaciones de SRBC, con objeto de poder compararlas con nuestros resultados. Concretamente, las implementaciones realizadas han sido [1, 9]:

- Modelo booleano de SRBC sin ponderación de atributos usando la distancia euclídea como medida de similitud (CB-Euclídea)
- Modelo booleano de SRBC sin ponderación de atributos usando la función coseno como medida de similitud (CB-Coseno)
- Modelo booleano con ponderación de atributos TF-IDF.
- Modelo propuesto con ponderación de atributos multivaluados basado en la entropía y medidas de dependencia (PABED).

4.1 CASO DE ESTUDIO

El conjunto de datos utilizado para verificar la validez del método propuesto es el extraído del sistema *Movilens* (http://www.movielens.org) desarrollado por el grupo de investigación *GroupLens Research* de la Universidad de Minesota. Es un servicio libre que permite valorar películas para, a partir de dichas valoraciones, ofrecer recomendaciones a los usuarios siguiendo un modelo de filtrado colaborativo. En nuestro caso, solo hacemos uso de los datos para aplicarlos a un modelo de recomendación basado en contenido.

Para las pruebas descritas en este artículo, se han seleccionado los usuarios con 20 valoraciones o más, obteniendo así un conjunto de datos de 9464734 valoraciones hechas por 69878 usuarios sobre 9768 películas.

El conjunto de datos extraído consta de tuplas <usuario,película,valoración>, donde la valoración viene dada de 1 a 5, siendo 1 la peor puntuación y 5 la mejor. La información descriptiva de las películas la hemos obtenido de la base de datos *IMDB* (http://www.imdb.com) considerando los siguientes atributos: título, año, género, director y país.

Para evaluar la eficacia de las distintas implementaciones se han usado métricas habituales [10]: precisión, recall y f-medida.

$$precision = \frac{relevantes\ recomendadas}{recomendadas} \quad (8)$$

$$recall = \frac{relevantes\ recomendadas}{relevantes} \quad (9)$$

$$f - medida = (1 + \beta^2) * \frac{precision * recall}{\beta^2 * precision + recall} \quad (10)$$

Para el cálculo de la precisión y el recall, consideramos relevantes para un usuario las películas con una valoración de 4 ó 5. En el cálculo de la f-medida se ha utilizado $\beta = 1$, dando igual importancia a la precisión y al recall.

Para evaluar la eficiencia de las implementaciones a comparar, se ha usado el tiempo de ejecución medio. En este experimento, las ejecuciones se han realizadon en un servidor dedicado con una CPU de cuatro núcleos de 2.00GHz con 2MB de caché y 8GB de memoria RAM.

4.2 RESULTADOS

El experimento ha consistido en realizar cincuenta ejecuciones de cada implementación utilizando la técnica de validación cruzada (Cross Validation) con 5 particiones. Asimismo, se han probado diferentes tamaños de la lista de items recomendados (K=1, 5, 10, 50, 100) obteniendo diferentes resultados. Tal y como puede observarse en las Figuras 2, 3 y 4, el método propuesto (PABED) supera considerablemente al método básico, sin ponderación (CB-Euclidea y CB-Coseno), en las métricas de precisión, recall y f-medida. En cuanto a la comparación con el método TF-IDF, se obtienen resultados muy similares en las métricas mencionadas (incluso se observa una ligera mejora para valores altos de K) pero en la métrica de eficiencia (tiempo de ejecución) la mejora conseguida es mucho más acentuada (ver Figura 5).

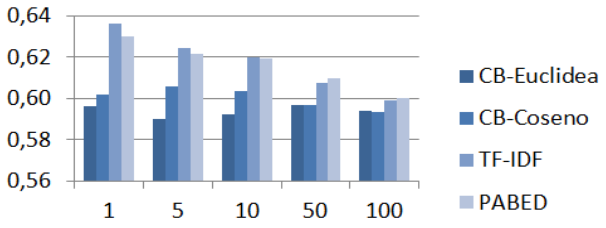

Figura 2: Precisión de cada sistema en función de k.

5 CONCLUSIONES

Los métodos de ponderación de atributos mejoran considerablemente los resultados en los sistemas de recomendación basados en contenido. TF-IDF es el método de ponderación más utilizado cuyo funcionamiento es bueno con

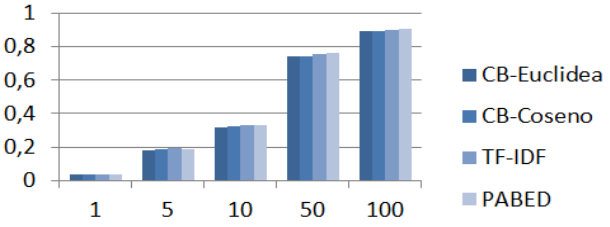

Figura 3: Recall de cada sistema en función de k.

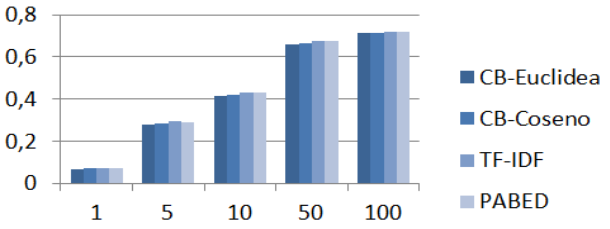

Figura 4: F1-medida de cada sistema en función de k.

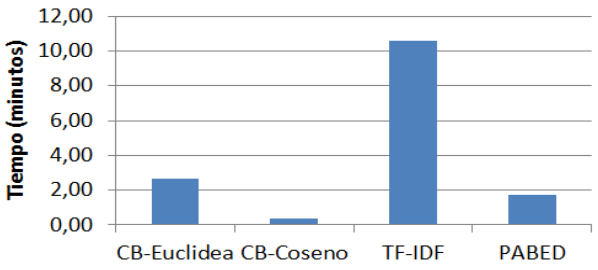

Figura 5: Tiempo de ejecución de cada sistema.

atributos booleanos. No obstante, cuando los atributos toman múltiples valores, dicho método puede mejorarse. En esta contribución hemos propuesto y evaluado un nuevo método para calcular pesos de atributos en sistemas de recomendación basados en contenido con atributos multivaluados que pueden ser tanto cuantitativos como cualitativos. El nuevo método está basado en dos factores: similitud intra-usuario y disimilitud inter-usuario. El primer factor se calcula con los coeficientes de correlación de Pearson, para atributos cuantitativos y de contingencia V de Cramer, para los cualitativos. Y para el segundo factor usamos la entropía que mide la cantidad de información aportada por cada atributo. El método ha sido evaluado obteniendo resultados satisfactorios tanto en eficacia como en eficiencia.

Agradecimientos

Esta contribución está parcialmente financiada por los proyectos de investigación TIN2009-08286, P08-TIC-03598, AGR-6581 y los fondos FEDER.

Referencias

[1] G. Adomavicius, A. Tuzhilin: Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions. IEEE Trans. on Knowledge and Data Engineering, vol 17, No. 6, June, pp.734-749, (2005)

[2] Akiko Aizawa: An information-theoretic perspective of TF-IDF measures. Information Processing and Managemente, 39:45-65 (2003)

[3] Manuel J. Barranco, Luis Martínez: A Method for Weighting Multi-valued Features in Content-Based Filtering. IEA/AIE 2010, Part III, LNAI 6098, Springer-Verlag, Berlin, pp. 409-418 (2010)

[4] Y.M.M.Bishop, S.E. Fienberg, P.W. Holland: Discrete Multivariate Analysis: Theory and Practice. The MIT Press, England (1995)

[5] T.M. Cover, J.A. Thomas: Elements of Information Theory. John Wiley & Sons, Inc. (1991)

[6] Tzung-Pei Hong, Jyh-Bin Chen: Finding relevant attributes and membership functions. Fuzzy Sets and Systems 103: 389-404 (1999)

[7] George H. John, Ron Kohavi, Karl Pfleger: Irrelevant features and the subset selection problem. Machine Learning: Proc. of the 11th int. conf., 121-129, Morgan Kaufmann Publishers, San Francisco, CA. (1994)

[8] L. Martínez, L.G. Pérez, M.J. Barranco: A Multigranular Linguistic Content-Based Recommendation Model. International Journal of Intelligent Systems, 22:5, pp.419-434 (2007)

[9] Michael J. Pazzani, Daniel Billsus: Content-Based Recommendation Systems, The Adaptive Web. Lecture Notes in Computer Science, Springer-Verlag, Vol. 4321, pp. 325-341 (2007)

[10] Sarwar, B., Karypis, G., Konstan, J., Riedl, J.: Application of dimensionality reduction in recommender systems-a case study. In ACM WebKDD Workshop (2000)

[11] J. Ben Schafer, Joseph A. Konstan, John Riedl: E-Commerce Recommendation Applications. Data Mining and Knowledge Discovery, 5, pp. 115-153, (2001)

[12] C.E. Shannon: A mathematical theory of communication. The Bell System Technical Journal, 27:379-423,623-656 (1948)

[13] P. Symeonidis, A. Nanopoulos, Y. Manolopoulos: Feature-weighted user model form recommender systems. Lecture Notes in Computer Science, Springer-Verlag, 4511:97-106 (2007)