Um Método de Filtragem Híbrida Baseado em Perfis Simbólicos Colaborativos

Byron Leite Dantas Bezerra, Francisco de Assis Tenório Carvalho, Valmir Macário Filho

Centro de Informática – Universidade Federal de Pernambuco (UFPE)
Av. Prof. Luiz Freire, s/n – Cidade Universitária – 52011-030 – Recife – PE – Brazil
{bldb,fatc,vmf2}@cin.ufpe.br

Abstract. *Recommendation Systems have become an important tool to cope with the information overload problem by acquiring information about the user behavior. In this paper we describe an approach to improve the recommendation quality in the first moments the user interacts with the system. The main idea is: (1) first of all, we describe the items with the general users opinion about them; and (2) after this, we use modal symbolic structures to save this content in the user profile. The proposed method outperforms, concerning the Find Good Items task measured by half-life utility metric, other approaches based on the following techniques: Cognitive Filtering, Social Filtering and hybrid methods.*

Resumo. *Os Sistemas de Recomendação têm se tornado uma ferramenta importante para lidar com o problema de sobrecarga de informação a partir da aquisição do perfil do usuário. Neste artigo nós descrevemos uma abordagem para melhorar a qualidade das recomendações nas primeiras interações do usuário com o sistema. A idéia básica é: (1) primeiramente, nós descrevemos os itens a partir das opiniões de todos os usuários sobre estes; e (2) após isso, utilizamos estruturas simbólicas modais para armazenar o conteúdo do perfil. O método proposto supera, segundo a métrica half-life utility sobre a tarefa Find Good Items, outras abordagens baseadas nas técnicas: Filtragem Cognitiva, Filtragem Social e outros métodos híbridos.*

1. Introdução

Sistemas de Recomendação (SR) permitem que sites de comércio eletrônico ofereçam produtos personalizados a seus clientes, provendo informações relevantes para ajudá-lo em suas compras. Para isso, há duas abordagens principais ([Herlocker, J. et al (2004)][Schafer, J.B. et al (2001)]). A primeira é a apresentação de uma lista de itens ordenados de acordo com sua relevância para o usuário ativo. A segunda é a estimação da relevância dessa lista baseada nas preferências do usuário ativo.

Independentemente da abordagem de recomendação, os SR necessitam das preferências dos usuários, que podem ser adquiridas implicitamente (escutando uma música na loja virtual, comprando um CD) ou explicitamente (avaliando algum artigo com uma nota em uma loja virtual) [Burke, R. (2002)][Claypool, M. et al (2001)][Hasenjäger, M. (2000)]. Naturalmente, quanto mais preferências do usuário são coletadas, melhores sugestões podem ser oferecidas. Mas um problema relevante

remanesce neste processo: o usuário não tem tempo suficiente para dispor informações sobre si próprio. Então, é necessário aprender como lidar com o mínimo de informação acerca do usuário. Nesse problema, o mais difícil de lidar é com a primeira utilização do sistema pelo usuário quando não existem informações sobre ele. Nesse caso é preciso uma estratégia apropriada para aquisição das preferências do usuário.

O próximo passo é filtrar informações relevantes. As soluções propostas para essa etapa podem ser classificadas em dois grupos principais concentrando o tipo de estratégia de filtragem, ou seja, FC – Filtragem Cognitiva ou Baseada em Conteúdo (se baseia na correlação entre o perfil do usuário e seu conteúdo) ou FS – Filtragem Social ou Colaborativa (se baseia em correlação entre usuários). Essas técnicas possuem limitações inerentes, tal como a impossibilidade de codificar algumas informações na primeira abordagem [Balanovic, M. and Shoham, Y. (1997)] e latência (ou problema do *cold-start*) na segunda. Entretanto, muitos trabalhos ([Balanovic, M. and Shoham, Y. (1997)],[Melville, P. et al (2002)],[Pazzani, M. (1999)],[Popescul, A. (2001)],[Sarwar, B. et al (1998)],[Yu, K. et al (2003)]) têm explorado sistemas híbridos para atenuar as limitações de cada uma das abordagens.

Nesse artigo nós descrevemos o sistema SMCF, um método inteligente para conseguir melhores listas de recomendações desde as primeiras utilizações do sistema. De acordo com [Burke, R. (2002)], sistemas híbridos geralmente combinam duas diferentes técnicas (por exemplo, FC e FS) para gerar uma simples saída de algum SR. No entanto, nós propomos uma abordagem híbrida diferenciada em que os dados colaborativos servem para construir o perfil do usuário que, por sua vez, é utilizado como entrada para um algoritmo colaborativo.

Basicamente, a idéia é descrever cada item no repositório por ações ou avaliações que a comunidade realiza através de variáveis simbólicas modais (seção 3). Depois, as descrições de itens avaliados são utilizadas para construir um perfil simbólico modal do usuário. Esse perfil é então comparado com outros perfis simbólicos com a finalidade de produzir recomendações num ambiente colaborativo.

A próxima seção introduz o estado da arte de sistemas de filtragem de informação (FI) híbrida. Apresentamos na seção 3 a solução proposta, que é avaliada empiricamente no domínio de recomendação de filmes na seção 4. Finalmente, algumas conclusões e possíveis direções são listadas na seção 5.

2. Abordagens de Filtragem de Informações Híbridas

A maioria dos trabalhos relacionados a FI híbrida classificam seus métodos em uma das três categorias descritas a seguir, onde cada categoria possui uma estratégia particular para combinar um algoritmo FC com um FS. A primeira abordagem é a medição do peso das predições geradas por ambos os algoritmos de FI com uma função apropriada para cálculo da predição final [Pazzani, M. (1999)].

A outra estratégia é estimar a pontuação de algum item desconhecido no repositório para um subconjunto de usuários através de um algoritmo FC e após essa etapa, utiliza-se um algoritmo de FS para predizer novos itens para o mesmo usuário. O sistema de pesquisa GroupLens [Sarwar, B. et al (1998)] desdobra alguns agentes automáticos que simulam o comportamento humano e calculam pontuações para novos

itens no repositório diminuindo o problema do *cold-start*. Além disso, o algoritmo FC [Melville, P. et al (2002)] é utilizado para converter matrizes esparsas em matrizes compactas; e depois utiliza FS para prover recomendações. Assim, essa técnica minimiza o problema da esparsidade e consegue bons resultados em alguns contextos.

A terceira categoria é caracterizada por algoritmos que constroem e mantêm um perfil de usuário baseado em uma descrição de conteúdo previamente avaliada pelo usuário. Esses perfis permitem medir a correlação entre usuários, que é importante para definir vizinhanças em recomendações colaborativas. Por exemplo, em [Adomavicius G., Tuzhilin A. (2005)] o perfil é baseado no conteúdo de páginas da internet avaliadas.

De acordo com [Burke, R. (2002)], existem outras possibilidades de SR baseados em combinações de muitos tipos de informações (demográficos, avaliações, características dos itens, etc) e técnicas (FS, FC, baseada em conhecimento, baseada em regras, etc). Uma categoria de SR discutida em [Burke, R. (2002)] é a *meta-level,* a qual toma o modelo aprendido por um classificador como entrada para o segundo classificador e, esse último, é responsável por gerar a saída final. Embora essa estratégia esteja relacionada ao sistema SMCF, [Burke, R. (2002)] não considera a possibilidade de combinação de classificadores que utilizem a mesma técnica na categoria *meta-level*.

3. SMCF :: Filtragem Híbrida Baseada em Perfis Simbólicos Colaborativos

Dados Simbólicos Modais (SM) foram introduzidos pela área de Análise de Dados Simbólicos (ADS). ADS fornece ferramentas apropriadas para gerenciar dados agregados descritos por variáveis multivaloradas [Bezerra, B. L. D. and De Carvalho, F.A.T. (2004)].

Em [De Carvalho, F.A.T. and Bezerra, B.L.D. (2002)], a primeira FI baseada em dados SM foi implementada. Naquele caso, esse tipo de dado foi utilizado para descrever os atributos dos itens, agregando avaliações dos itens em estruturas SM no perfil, que é comparado com cada item em uma abordagem de FC.

Além disso, também foi proposta em [Bezerra, B. L. D. and De Carvalho, F.A.T. (2004)] uma estratégia para recomendar itens com o mínimo de informação disponível, também baseada em dados SM. A diferença é que o segundo método define uma função apropriada para medir a distância entre perfis SM dos usuários, produzindo um algoritmo de FS, ao invés da FC usada em [De Carvalho, F.A.T. and Bezerra, B.L.D. (2002)].

No sistema SMCF, também adotamos uma estrutura simbólica para construir o perfil do usuário. Entretanto, nessa nova abordagem, a origem da informação utilizada para descrever um item deriva da comunidade, como uma alternativa aos atributos do item [De Carvalho, F.A.T. and Bezerra, B.L.D. (2002)] ou dados demográficos [Pazzani, M. (1999)]. Com o intento de exemplificar nossa solução, uma parte da matriz de avaliações do domínio de filme é apresentada na Tabela 1.

Tabela 1 – Parte da matriz de avaliações entre 1 a 5 dos usuários.

	Batman	Seven	Matrix	Titanic	Monsters
Auana	5	∅	2	5	∅
Bricia	3	2	2	∅	5
Elaine	1	4	5	3	5

Vanessa	4	∅	4	∅	5

A figura a seguir representa o histograma das taxas obtidas pelos usuários da Tabela 1 para o filme "Matrix". Nesse exemplo, existem cinco possíveis notas coletadas explicitamente, mas o mesmo conceito pode ser aplicado no caso de obtenção implícita das preferências dos usuários (nós só precisamos ordenar todas as possíveis ações do usuário em uma lista de prioridades). Independente do processo de aquisição, histogramas representam a opinião geral da comunidade sobre um item.

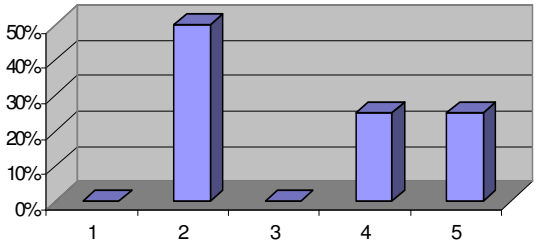

Figure 1. Histograma das notas dos usuários da Tabela 1 para o filme "Matrix".

3.1. Construindo o Perfil Simbólico Modal do Usuário

Como delineado em [Bezerra, B. L. D. and De Carvalho, F.A.T. (2004)] e [De Carvalho, F.A.T. and Bezerra, B.L.D. (2002)], a construção das descrições SM do perfil do usuário envolve duas etapas: (a) pré-processamento e (b) generalização. A idéia geral é (a) construir uma descrição SM para cada item avaliado pelo usuário e (b) depois agregar essas descrições em algumas descrições SM onde cada uma representa o interesse de um usuário.

Pré-Processamento. O pré-processamento é necessário para ambas as construções de conjuntos de descrições simbólicas, utilizadas para representar perfil do usuário e comparar o perfil do usuário com um novo item ou com outro perfil do usuário.

Seja $D = \{g_1,g_2,...,g_n\}$ o conjunto de níveis possíveis de interesse de que um usuário possa ter por algum item do repositório. Na Figura 1, esse conjunto é equivalente ao intervalo de notas, ou seja, $D = \{g_1=1,g_2=2,g_3=3,g_4=4,g_5=5\}$. Seja $m_i^k = (\mathcal{I}_i^k, g_k)$, onde $\mathcal{I}_i^k \in \mathrm{P}(U)$ contém todos os usuários que mostraram algum interesse g_k no item i. Então a descrição de um item $\boldsymbol{i}$ (i=1, ..., n) é definida como $x_i = (X_i, C(i))$, onde $X_i = \{m_i^1, m_i^2, ..., m_i^n\} \subseteq (\mathrm{P}(U) \times D)$ e $C(i) \in D$ se refere ao nível de interesse de um usuário alvo para o item i. Observe o exemplo na Tabela 2.

Tabela 2 – Descrição do filmes "Matrix" de acordo com Tabela 1.

X_i	m_i^1	({}, 1)
	m_i^2	({"Auana", "Bricia"}, 2)
	m_i^3	({}, 3)
	m_i^4	({"Vanessa"}, 4)
	m_i^5	({"Elaine"}, 5)
C(i)		*<Taxa do usuário alvo>*

Para cada categoria $m_i^k \in X_i$, nós podemos associar o seguinte peso:

$$w(m_i^k) = \frac{\left|\Im_i^k\right|}{\sum_{k=1}^{n}\left|\Im_i^k\right|}$$

onde $|\Im_i^k|$ é a cardinalidade de $\Im_i^k$. Então, a descrição SM de um item *i* é dada por $\tilde{x}_i = (\tilde{X}_i, C(i))$, onde $\tilde{X}_i = (S(i), q(i))$ é uma variável SM, e $S(i) = \{g_k \mid m_i^k \in X_i\ ,\ k=1,...,n\}$ é o suporte da distribuição de peso $q(i)$. Observe o exemplo na Tabela 3.

Tabela 3 – Descrições Simbólicas Modais dos filmes listados na Tabela 1.

Batman	(({1,2,3,4,5}, {0.25,0,0.25,0.25,0.25}), *C*(*i*))
Seven	(({1,2,3,4,5}, {0,0.50,0,0.50,0}), *C*(*i*))
Matrix	(({1,2,3,4,5}, {0,0.50,0,0.25,0.25}), *C*(*i*))
Titanic	(({1,2,3,4,5}, {0,0,0.50,0,0.50}), *C*(*i*))
Monsters	(({1,2,3,4,5}, {0,0,0,0,1.0}), *C*(*i*))

Generalização. Como desenvolvida em [Bezerra, B. L. D. and De Carvalho, F.A.T. (2004)] e [De Carvalho, F.A.T. and Bezerra, B.L.D. (2002)], a etapa de generalização permite construir uma descrição SM apropriada do perfil do usuário. Em nossa abordagem, cada usuário é formado por um conjunto de sub perfis. Cada sub perfil é modelado por uma descrição SM que contém um sumário do corpo de todas as informações do conjunto de itens que o usuário tenha avaliado com a mesma nota.

Formalmente, seja u_g o sub perfil do usuário *u* o qual é formado por conjuntos de ítens que foram avaliados com a nota *g*. Seja $y(u_g) = (S(u_g), q(u_g))$ a descrição SM do sub perfil u_g, com $S(u_g)$ sendo o suporte da distribuição de pesos $q(u_g)$. Se $\tilde{x}_i = (\tilde{X}_i, C(i))$ é a descrição do item *i* pertencente a u_g,o suporte $S(u_g)$ de $q(u_g)$ é definido como:

$$S(u_g) = \bigcup_{i \in u_g} S(i)$$

Seja $g_k \in S(u_g)$ uma categoria pertencente a *D* e $|u_g|$ o número de elementos pertencentes ao conjunto u_g. Então, o peso $W(u_g, g_k) \in q(u_g)$ da categoria g_k é:

$$W(u_g, g_k) = \frac{1}{|u_g|}\sum_{i \in u_g}\delta(i, g_k)\ ,\ \text{onde a função } \delta \text{ é } \delta(i, g_k) = \begin{cases} w(m_i^k) \in q(i), & if \quad g_k \in S(i) \\ 0, & otherwise \end{cases}$$

De acordo com as equações anteriores, o perfil SM de Brícia é:

Tabela 4 – Perfil simbólico modal de Brícia.

$y(Bricia_1)$	({1,2,3,4,5} , {0,0,0,0,0})
$y(Bricia_2)$	({1,2,3,4,5} , {0,0.50,0,0.375,0.125})
$y(Bricia_3)$	({1,2,3,4,5} , {0.25,0,0.25,0.25,0.25})
$y(Bricia_4)$	({1,2,3,4,5} , {0,0,0,0,0})
$y(Bricia_5)$	({1,2,3,4,5} , {0,0,0,0,1.0})

3.2 Colaboração Através de Perfis Simbólicos Modais

Nessa seção, nós descrevemos como o perfil SM apresentado na seção 3.1 pode ser utilizado para gerar listas de recomendações personalizadas. Basicamente, as etapas abaixo devem ser seguidas:

1. Construção de um perfil de usuário SM. Essa etapa pode ser realizada de forma incremental sem degradar o uso da memória como discutido anteriormente.
2. Medir a similaridade de todos os usuários com o usuário alvo. Similaridade entre os usuários é medida através de uma função que compara as descrições SM de cada usuário.
3. Selecionar os *h* usuários mais próximos ao usuário alvo. O quão perto o usuário se encontra é definido pela similaridade de algum vizinho candidato e o usuário alvo.
4. Geração de uma lista ordenada de itens após o cálculo de predições a partir de uma combinação de avaliações dos vizinhos selecionados.

Embora, as etapas 2-4 sejam padrões em algoritmos colaborativos, essa segunda é especialmente diferente porque é baseada em funções apropriadas que comparam perfis modais simbólicos de usuários [Bezerra, B. L. D. and De Carvalho, F.A.T. (2004)]. Entretanto, essa função é utilizada para definir a vizinhança do usuário alvo.

Seja $y(u_g) = (S(u_g), q(u_g))$ a descrição SM do sub perfil u_g do usuário alvo. Também, seja $y(v_g) = (S(v_g), q(v_g))$ a descrição SM do sub perfil v_g de um candidato a vizinho do usuário alvo. A comparação entre o usuário *u* e o candidato a vizinho *v* é obtida através da função de similaridade:

$$\psi(u,v) = \frac{1}{|D|} \sum_{g \in D} \left(1 - \phi(y_{u_g}, y_{v_g})\right)$$

onde *D* é o conjunto de todos os níveis possíveis de interesse que o usuário pode possuir por algum item do repositório, e |*D*| é sua cardinalidade.

A função de dissimilaridade ϕ compara duas descrições simbólicas modais, levando em conta diferenças entre as distribuições de pesos para cada elemento no suporte. Uma versão adaptada da distância *euclidiana* é utilizada para definir a função de dissimilaridade ϕ, como apresentada na equação a seguir:

$$\phi(y_{u_g}, y_{v_g}) = \sqrt{\sum_{g_k \in S(u_g) \cap S(v_g)} \left(W(u_g, g_k) - W(v_g, g_k)\right)^2}$$

Agora que nós estamos aptos a computar a similaridade entre um usuário alvo *u* com cada usuário na base de dados, nós podemos executar a 3ª de uma maneira direta.

Finalmente, o objetivo da 4ª etapa é gerar uma lista ordenada de itens de acordo com as necessidades do usuário. Para conseguir esse objetivo, nós precisamos calcular predições para cada item desconhecido no repositório utilizando a vizinhança encontrada na etapa 3. Assim, considerando que a função ρ que mede a relevância de um item *i* para algum usuário *u*:

$$\rho(u,i) = \bar{r}_u + \frac{\sum_{v=1}^{h} \left(r_{v,i} - \bar{r}_v\right) * \psi(u,v)}{\sum_{v=1}^{h} \psi(u,v)}$$

onde *h* é o tamanho da vizinhança. Finalmente, ordenamos uma lista de itens de acordo com os valores produzidos pela equação anterior e apresentamos essa lista ao usuário.

4. Validação Experimental

Nosso estudo de caso é o domínio de recomendação de filmes. Nós utilizamos o conjunto de dados Movielens (*http://movielens.umn.edu*) em conjunto com uma base de conteúdo do *site* IMDB (*http://www.imdb.com*). O conjunto de dados definitivo contém 91190 avaliações de 1466 filmes no intervalo de 1 (a pior nota) a 5 (a melhor nota) de 943 diferentes usuários. Três questões são importantes para nossos experimentos:

1. Quanto o sistema deve saber acerca de um usuário para prover recomendações?
2. Que tarefas de recomendação nós temos interesse em avaliar?
3. Como podemos avaliar essa(s) tarefa(s) e comparar o desempenho de vários sistemas?

A respeito da primeira questão, nesse artigo nós estamos interessados na avaliação da utilidade de um sistema de recomendação nas primeiras interações do usuário. Nesse cenário, o usuário não prove muitas informações sobre ele mesmo, primeiro porque não há muito tempo para fazer isso, e segundo nãoé uma boa estratégia para sistemas de informação perguntar muitas questões ao usuário, podendo acarretar que ele/ela possa deixar o sistema e não voltar mais. Assim, achamos razoável que o usuário avalie de 5 a 10 itens no primeiro contato com o sistema.

Sobre a segunda questão, é solicitado ao nosso sistema fornecer algumas listas ordenadas (tarefa *Find Good Items* [Herlocker, J. et al (2004)]). Essa decisão foi motivada pela hipótese que num ambiente de comércio eletrônico essa tarefa seja mais útil que as demais funcionalidades disponíveis em SR [Schafer, J.B. et al (2001)].

Finalmente, a respeito da nossa terceira questão, de acordo com [Herlocker, J. et al (2004)], uma métrica de ordem apropriada foi proposta por [Breese, J. et al (1998)] chamada de *half-life utility.* A avaliação dessa métrica foi especialmente designada para tarefas como *Finf Good Items*. A maior vantagem dessa métrica é que ela mede a utilidade de uma lista ordenada levando em conta que o usuário geralmente observa os primeiros resultados da lista. Depois, ela assume a hipótese que cada item sucessivo numa lista é menos interessante para o usuário de acordo com um decremento exponencial. Consulte [Breese, J. et al (1998)],[Burke, R. (2002)] e [Herlocker, J. et al (2004)] para maiores detalhes da métrica *half-life utility.*

Nesse ponto nós podemos descrever a metodologia utilizada nos experimentos. Primeiro, selecionamos usuários que tenham avaliado ao menos 100 itens dos 1466 filmes disponíveis. Esses usuários foram utilizados no conjunto de teste para executar quatro experimentos diferentes com os números m={5,10} de itens obtidos no conjunto de treinamento para cada usuário de vizinhança h={30,50}.

Além disso, uma versão adaptada da metodologia estratificada 10 *fold cross-validation* foi executada [Witten, I.H. and Frank, E. (2000)] (pages 125:127). Essa adaptação consiste em arranjar o conjunto de treinamento e o conjunto de testes, respectivamente, em proporções 1/10 e 9/10, em vez de 9/10 e 1/10 como é realizado normalmente. Isso é compatível com o fato de cada usuário não dispor de informações suficientes no primeiro contato com o sistema.

Os seguintes algoritmos foram executados em nossos testes:

1. (MSA) – FC baseado em conteúdo SM;
2. (CFA) – kNN-CS baseado em correlação de Pearson;
3. (CnMCF) – FS baseada em conteúdo SM;
4. (SMCF) – Sistema colaborativo baseado em perfil SM.

Na Tabela 5 nós podemos ver a média (μ) e o desvio padrão (σ) da métrica *half-life utility* para todos os algoritmos agrupados por h={30,50} e m={5,10}. Essa tabela apresenta ainda valores observados para a *t-statistic* [Witten, I.H. and Frank, E. (2000)] a respeito de dois testes detalhados em 9 graus de liberdade entre o comportamento do método $SMCF_{h=30,m=5}$ e a média do comportamento dos outros métodos (MSA, CFA e CnMCF).

Tabela 5 – Resultados dos experimentos agrupados por *h* e *m* de acordo com a métrica *half-life utility*.

		MSA			CFA			CnMCF			SMCF	
		μ	σ	t	μ	σ	t	μ	σ	t	μ	σ
h=30	m=5	34.34	0.78	46.58	40.21	4.32	11.84	44.67	11.87	5.25	**63.84**	2.25
	m=10	31.8	1.11	64.77	58.09	1.99	6.23	53.49	9.71	3.49	63.54	2.01
h=50	m=5	34.34	0.78	46.58	40.21	4.32	11.84	63.87	2.21	-1.18	63.54	2.25
	m=10	31.84	1.11	64.77	58.09	1.99	6.23	63.57	2.00	2.65	63.54	2.01

Como esperado, o algoritmo MSA apresenta o desempenho independente do valor de h. Assim como, nós podemos ver que o mesmo acontece com os algoritmos CFA e SMCF. De fato, como apontado em [Herlocker, J. et al (2004)], a qualidade das recomendações executadas por sistemas colaborativos não variam significante quando o número de vizinhos é maior que 30. Entretanto SMCF não é um método colaborativo padrão, a entrada da função de correlação pessoa a pessoa é altamente baseada em dados sociais.

Podemos dizer a partir da Tabela 5 que o método MSA apresenta o pior desempenho quando existem apenas 5 itens no perfil do usuário. Isso acontece porque apenas a informação cognitiva é utilizada para produzir recomendações. A abordagem MSA é fundamentalmente uma filtragem baseada em conteúdo e esse tipo de sistema tradicionalmente apresenta piores resultados que os métodos colaborativos.

O valor 50 é determinado para melhorar a qualidade do sistema CnMCF. Além disso, estima-se que possuir 50 vizinhos e apenas 5 itens no perfil do usuário é suficiente para o CnMCF conseguir melhor precisão que os algoritmos CFA e MSA (com um c. l. de 0.1%). Conseguir boas recomendações com apenas 5 itens pode ajudar sistemas a obter consumidores leais e a conquistar novos clientes, principalmente porque não é necessário adquirir muita informação sobre o usuário durante as primeiras interações. Entretanto, CnMCF possue o pior desvio padrão.

O desvio padrão está relacionado com a estabilidade do sistema e também com a capacidade do sistema em armazenar informações equivalentes independente da variabilidade dos perfis. Consequentemente, um baixo desvio padrão significa que o sistema possui desempenho similar para todos os usuários, então é estável. É esperado que o sistema alcance uma grande estabilidade à medida que mais informações se tornam disponíveis. Nós vemos que o CnMCF é muito instável quando h é 30, entretanto é estável quando h é 50.

O sistema SMCF aumenta a precisão de forma semelhante com o sistema CnMCF, entretanto com uma estabilidade melhor e requerendo menos informações no perfil do usuário. De fato, o resultado mais importante em nossos experimentos está relacionado com o desempenho de SMCF. De acordo com a Tabela 5, é possível prover

melhores resultados personalizados utilizando 5 itens para construir o perfil do usuário assim como o CnMCF, mas requerendo apenas 30 vizinhos na etapa colaborativa. Além disso, o sistema SMCF é pelo menos 10 vezes mais rápido que o sistema CnMCF, porque a informação de conteúdo utilizada no perfil do usuário e as equações que medem a correlação de pessoa para pessoa são mais simples no sistema SMCF do que nas abordagens simbólicas modais anteriores [Bezerra, B. L. D. and De Carvalho, F.A.T. (2004)], [De Carvalho, F.A.T. and Bezerra, B.L.D. (2002)].

Os experimentos apresentados demonstraram que os métodos de filtragem de informação híbridos baseados em dados simbólicos, o CnMCF e o SMCF, são aptos a aprender mais acerca do usuário. A explicação para o melhor desempenho de ambos é que estes utilizam informação de conteúdo (conteúdo descritivo em CnMCF, informação social em SMCF) a fim de encontrar a vizinhança do usuário, a qual é mais rica que a CS, que utiliza apenas identificadores dos itens. Adicionalmente, o sistema SMCF é mais estável, rápido e requer menos informações que o sistema CnMCF.

5. Conclusões

Nós apresentamos nesse trabalho a proposta de uma estratégia adequada para minimizar o problema da aprendizagem do perfil do usuário na primeira utilização do sistema. Nossa estratégia é baseada em ambos: um método apropriado para de aquisição de preferência dos usuários e num novo método híbrido de FI que utiliza a idéia fundamental da abordagem de Filtragem Baseada em Objetos Modais Simbólicos [De Carvalho, F.A.T. and Bezerra, B.L.D. (2002)] para construir perfis enriquecidos de usuários. Adicionalmente, foi desenvolvido uma função de similaridade que torna possível selecionar os melhores vizinhos de um usuário ativo com a finalidade de executar o algoritmo *kNN-CF*.

Verificou-se que o desempenho superior do método proposto, na tarefa *Find Good Items*, quando existem poucas informações sobre o usuário. Supondo que o usuário avalia 5 itens no primeiro contato, o que é bastante aceitável na vida real, o sistema está apto a oferecer boas recomendações, o que pode motivar o usuário a retornar ao sistema, tornando-o fiel. Todavia, a vantagem mais importante de nossa técnica é que ela não necessita de informação do conteúdo dos itens. Consequentemente, nós podemos recomendar itens independentemente da sua natureza e complexidade, pois não precisamos descrevê-lo. Por exemplo, o sistema SMCF está apto a lidar com sentimentos especiais, como o “gosto do vinho” ou “cheiro do perfume”.

Pelo menos dois pontos podem ser citados como trabalhos futuros:

1. Avaliar outras funções para medir a correlação entre perfis simbólicos modais;
2. Combinar CnMCF e SMCF numa estratégia única.

Agradecimentos. Os autores gostariam de agradecer ao CNPq pelo suporte financeiro.

Referências Bibliográficas

Adomavicius G., Tuzhilin A. Toward the next generation of recommender systems: a survey of the state-of-the-art. Knowledge and Data Engineering, IEEE, 2005.

Balanovic, M. and Shoham, Y.: Fab: Content-based, collaborative recommendation. Communications of the ACM, Vol. 40 (1997) 88-89.

Bezerra, B. L. D. and De Carvalho, F. de A. T.. A Symbolic Hybrid Approach to Face the New User Problem in Recommender Systems. In: Australian Joint Conference on Artificial Intelligence - AI2004, Cairns. Proceedings of the 17th Australian Joint Conference on Artificial Intelligence. Berlin (Alemanha): Springer-Verlag, 2004.

Bock, H.H. and Diday, E.: Analysis of Symbolic Data. Springer, Heidelberg (2000).

Breese, J., Heckerman, D., and Kadie, C.: Empirical analysis of predictive algorithms for collaborative filtering. In Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence (1998) 43-52.

Burke, R.. Hybrid recommender systems: survey and experiments. User Modeling and User-Adapted Interaction. November, 2002.

Claypool, M., Brown, D., Phong Le, Waseda M. Inferring User Interests. IEEE Internet Computing, Vol. 5 (2001), 32-39.

De Carvalho, F.A.T. and Bezerra, B.L.D.: Information Filtering based on Modal Symbolic Objects. Proceedings of the 26th Annual Conference of the Gesellschaft für Klassifikation (GfKl), Springer (2002) 395-404.

Hasenjäger, M. Active Data Selection in Supervised and Unsupervised Learning. PhD thesis, Technische Fakultät der Universität Bielefeld (2000).

Herlocker, J.L., Konstan, J.A., Terveen, L.G., and Riedl, J.: Evaluating Collaborative Filtering Recommender Systems. ACM Transactions on Information Systems, Vol. 22, Issue 1 (2004) 5-53.

Melville, P., Mooney, R.J., and Nagarajan, R.: Content-Boosted Collaborative Filtering for Improved Recommendations. Proceedings of the Eighteenth National Conference on Artificial Intelligence (2002) 187-192.

Pazzani, M.: A framework for collaborative, content-based and demographic filtering. Artificial Intelligence Review, Vol. 13 (5-6), (1999) 393-408.

Popescul, A., Ungar, L.H., Pennock, D.M., and Lawrence, S.: Probabilistic Models for Unified Collaborative and Content-Based Recommendation in Sparse-Data Environments. 17th Conference on Uncertainty in Artificial Intelligence (2001).

Sarwar, B., Konstan, J., Borchers, A., Herlocker, J., Miller, B. and Riedl, J.: Using Filtering Agents to Improve Prediction Quality in the Grouplens Research Collaborative Filtering System. In Proceedings of the ACM Conference on Computer Supported Cooperative Work (1998) 345-354.

Schafer, J.B., Konstan, J.A., and Riedl, J.: E-Commerce Recommendation Applications. Data Mining and Knowledge Discovery, Vol. 5. (2001) 115-153.

Witten, I.H. and Frank, E.: Data Mining - Practical Machine Learning Tools and Techniques with Java Implementations. San Diego, CA: Morgan Kaufmann, (2000).

Yu, K., Schwaighofer, A., Tresp, V., Ma, W.-Y., and Zhang, H.: Collaborative ensemble learning: Combining collaborative and content-based information filtering via hierarchical Bayes. In C. Meek and U. Kjærulff, editors, Proceedings of UAI 2003, Morgan Kaufmann, (2003) 616-623.