Universidade Federal de Pernambuco
Centro de Informática

Trabalho de Graduação em Inteligência Artificial

ESTUDO DE ALGORITMOS DE FILTRAGEM DE INFORMAÇÃO BASEADOS EM CONTEÚDO

Autor: Byron Leite Dantas Bezerra
Orientador: Francisco de Assis Tenório Carvalho
Co- Orientador: Geber Lisboa Ramalho

Recife, 2002

RESUMO

Sistemas de Recomendação têm o objetivo de fornecer recomendações baseadas em informações registradas sobre as preferências dos usuários. Para isso, tais sistemas utilizam técnicas de Filtragem de Informação para processar informações e prover ao usuário itens potencialmente mais relevantes. Uma das abordagens mais difundidas em Filtragem de Informação é a Filtragem Baseada em Conteúdo. Geralmente essa abordagem baseia-se na comparação entre a descrição do perfil do usuário e as descrições dos itens a serem recomendados. Ou seja, o sistema seleciona os itens que atingem as melhores posições em uma seqüência definida por um critério de ordenação conveniente. O presente trabalho descreve uma abordagem onde cada usuário é modelado por um meta - protótipo e a comparação entre um item e o perfil é baseada em uma função de comparação composta por dois componentes: posição e conteúdo.

ABSTRACT

Recommender Systems aims to furnish automatic recommendations based on information recorded about user preferences and uses Information Filtering techniques to manage this information and provide the user with options which will present greater possibility to satisfy the user. Content based Filtering is one of the most useful approaches used for Information Filtering. Often in this approach, the recommendation is based on a comparison between the description of a user profile and the descriptions of the items. Therefore, the system selects those items that score the best according to a particular criterion. This work presents an approach where each user profile is modeled by a meta-prototype and the comparison between an item and a profile is based on a suitable matching function which possess two components: position and content.

AGRADECIMENTOS

Em primeiro lugar agradeço a Deus por me iluminar sempre em todos os momentos difíceis da minha vida. Agradeço sobretudo a minha família - meus pais, minha irmã e minha noiva - por sempre compreenderem os momentos ausentes em vista das atividades exigidas para o êxito do presente trabalho.

Agradeço aos meus orientadores, ou melhor, amigos, Francisco Carvalho e Geber Ramalho. Graças a eles obtive a direção correta para as pesquisas científicas realizadas nesse trabalho.

Por fim, agradeço ao CESAR e a todo seu corpo de colaboradores, por propiciarem um ambiente agradável e produtivo para o desenvolvimento de boa parte desse trabalho.

ÍNDICE

1 INTRODUÇÃO 1

1.1 MOTIVAÇÃO 2

1.2 O TRABALHO DE GRADUAÇÃO 3

2 O ESTADO DA ARTE 4

2.1 SISTEMAS DE RECOMENDAÇÃO 4

3 K VIZINHOS MAIS PRÓXIMOS - KNN 6

4 FILTRAGEM BASEADA EM META-PROTÓTIPO - FMP 8

4.1 CONCEITOS 8

4.1.1 Dados Simbólicos 8

4.2 CONSTRUÇÃO DO META-PROTÓTIPO 9

4.2.1 Pré-processamento 10

4.2.2 Generalização 10

4.3 A COMPARAÇÃO ENTRE META-PROTÓTIPOS 11

5 DEFINIÇÃO DO AMBIENTE EXPERIMENTAL 13

5.1 BASE DE DADOS 13

5.2 QUALIDADE DAS RECOMENDAÇÕES 13

5.2.1 Métrica 14

6 KNN NO DOMÍNIO DE FILMES 16

6.1 RELEVÂNCIA DOS ATRIBUTOS 16

6.1.1 Pesos dos Atributos 18

6.2 A COMPARAÇÃO ENTRE ATRIBUTOS 19

6.2.1 O Atributo Textual 20

7 FMP NO DOMÍNIO DE FILMES 22

7.1 REPRESENTAÇÃO 22

7.1.1 Representação do Perfil do Usuário 22

7.2 COMPARAÇÃO 23

7.3 APRIMORANDO A FMP 24

7.3.1 Composição do Perfil do Usuário 24

7.3.2 As Avaliações do Usuário como Heurística 25

8 EXPERIMENTOS E RESULTADOS 31

8.1 DESEMPENHO 31

9 CONCLUSÕES 35

10 TRABALHOS FUTUROS 36

11 REFERÊNCIAS BIBLIOGRÁFICAS 37

12 APÊNDICE 40

12.1 GUIA DE ENTRETENIMENTO PERSONALIZADO – GEP 40

1 INTRODUÇÃO

A explosão de documentos eletrônicos nos diversos tipos de rede fez surgir o conceito de repositórios de informação digital. Basicamente, esses repositórios consistem em agrupamentos lógicos de documentos eletrônicos com precária organização estrutural da informação. No contexto da Internet pode-se dizer que grandes portais são exemplos claros de repositórios de informação digital. Da mesma forma, as redes internas de corporações de médio e grande porte constituem repositórios de informação digital, visto que nesses ambientes surge uma quantidade enorme de documentos eletrônicos. A figura abaixo ilustra o conceito de repositórios digitais.

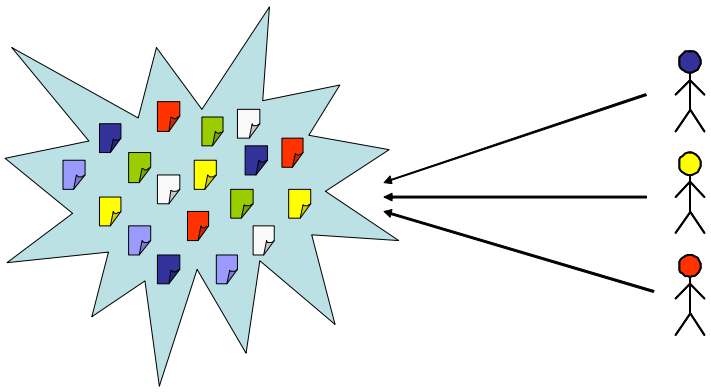

Ilustração 1- O repositório digital possui uma ampla e diversa quantidade de documentos. Ao mesmo tempo, existem usuários acessando os documentos do repositório com necessidades de informações diferentes e perfis distintos.

Um dos grandes problemas que surgem com a disseminação dos repositórios de informação digital é a dificuldade em se encontrar informação relevante em tempo hábil, dada a grande disponibilidade de informação minimamente organizada. Isso é o que chamamos de *sobrecarga de informação*.

Diante do cenário exposto justifica-se a necessidade de ferramentas de auxílio à filtragem de informações relevantes de forma eficaz e de fácil uso. Além disso, pelo fato dos repositórios digitais estarem disponíveis a uma grande diversidade de usuários com necessidades de informações diferentes e em momentos distintos, mostra-se bastante conveniente que tais

ferramentas forneçam personalização no nível de conteúdo tomando como base o perfil do usuário.

Sistemas de Informação que filtram informações relevantes para um determinado usuário baseado em seu perfil são conhecidos como *Sistemas de Recomendação*. Esses sistemas utilizam em geral duas abordagens de filtragem de informação: *Filtragem Baseada em Conteúdo* (Content-Based Filtering) e *Filtragem Colaborativa* (Collaborative Filtering) [1]. A primeira analisa a correlação do conteúdo dos itens com o perfil do usuário a fim de sugerir os itens relevantes e descartar os itens insignificantes. O segundo método baseia-se na correlação entre os perfis de usuários. A idéia básica é selecionar os itens preferidos pelos usuários cujas preferências mais se assemelham ao gosto do usuário alvo. Comprova-se facilmente que essas duas abordagens são complementares [2]. Nesse sentido, vários estudos sugerem formas de combinar as técnicas de filtragem baseada em conteúdo com as técnicas de filtragem colaborativa, com o propósito de resolver suas limitações intrínsecas [3][4][5].

1.1 MOTIVAÇÃO

A motivação do presente trabalho surgiu dos seguintes pressupostos:

i. a filtragem colaborativa não é suficiente para todos os cenários de uso de um sistema de recomendação, devido aos problemas inerentes a esse tipo de solução [2];
ii. a filtragem baseada em conteúdo é necessária em sistemas de recomendação mas as abordagens atuais sofrem de problemas de desempenho que são de grande importância nesse tipo de sistema;
iii. as características *i* e *ii* foram observadas em um sistema de informação comercial, o *GEP* (anexo 12.1).

Dessa forma, a proposta inicial desse trabalho é a investigação dos algoritmos de filtragem de informação baseada em conteúdo passíveis de serem implementados em sistemas de recomendação comerciais.

1.2 O TRABALHO DE GRADUAÇÃO

O presente trabalho foi realizado no quadro do projeto Smart-es, fruto do acordo CAPES-COFECUB entre o LIP6 (Universidade de Paris VI, França) e o CIn (Centro de Informática da UFPE). Nessa monografia é apresentada uma nova abordagem de filtragem baseada em conteúdo, chamada de Filtragem Baseada em Meta-Protótipos (FMP), que melhora significativamente o desempenho de sistemas de recomendação mantendo a qualidade da resposta equivalente à alcançada pelo método dos *k Vizinhos Mais Próximos* [6]. O método de FMP foi aplicado com sucesso pela primeira vez na classificação de um tipo especial de imagens SAR simuladas [10] e é originário das pesquisas básicas realizadas na abordagem simbólica em classificação e métodos afins [16]. O presente trabalho representa uma contribuição efetiva por ser o primeiro a aplicar o método FMP no domínio de filmes de uma forma bastante original e, principalmente, por melhorar o desempenho de sistemas de recomendação sem comprometer a exatidão da resposta.

2 O ESTADO DA ARTE

Esta seção apresenta uma visão geral de sistemas de recomendação bem como da área de filtragem de informação.

2.1 SISTEMAS DE RECOMENDAÇÃO

A implementação dos chamados sistemas de recomendação pressupõe a resolução de dois subproblemas:

i. a coleta do perfil do usuário;
ii. a exploração adequada desse perfil.

O perfil do usuário pode ser coletado algumas vezes implicitamente e outras apenas explicitamente. Uma aquisição implícita poderia ser pensada em um cenário de um sítio de comércio eletrônico. Nesse caso a seguinte política aplicar-se-ia: cada vez que um usuário compre algum produto o sistema registraria essa informação no perfil do usuário de forma positiva. Esse modelo implícito de aquisição do perfil do usuário é adotado pela Amazon Book Store [23].

Não obstante, algumas vezes é inviável a aquisição implícita do perfil do usuário. Considere por exemplo um sítio de recomendação de filmes. Assumindo que tal sistema não comercializa filmes, não seria possível aplicar o modelo de aquisição implícita de forma adequada. Sendo assim, o sistema necessita uma avaliação do usuário com relação aos filmes apresentados, dando origem a um perfil de usuário constituído dos filmes avaliados e suas respectivas notas atribuídas por esse usuário. Esse modelo explícito de aquisição de perfil é utilizado no sítio PTV [24].

O segundo problema mencionado anteriormente tem sido bastante estudado nos últimos anos. Embora as soluções propostas sejam diferentes elas podem ser classificadas em duas categorias principais com relação ao tipo de filtragem adotada. Algumas dessas soluções baseiam-se na análise da correlação entre o conteúdo dos itens com o perfil do usuário para recomendar itens relevantes e descartar os itens não pertinentes. Essa técnica é chamada de Filtragem Baseada em Conteúdo. Uma outra

abordagem, chamada de Filtragem Colaborativa, é baseada na correlação entre perfis de usuários.

Na próxima seção será discutida uma abordagem de filtragem de informação baseada em conteúdo largamente utilizada em sistemas de recomendação, o método dos *k Vizinhos Mais Próximos*.

3 K VIZINHOS MAIS PRÓXIMOS - KNN

No algoritmo dos *k Vizinhos mais* Próximos, originalmente proposto em [11], e outros algoritmos de *aprendizagem baseada em instância* [12], os exemplos são instâncias originais do conjunto de treinamento (itens do perfil do usuário). Durante a generalização, esses algoritmos usam uma função de distância para determinar quão próximo um novo vetor de entrada *y* está a cada instância da memória, e utiliza as instâncias mais próximas para inferir a classe de saída de *y*. O algoritmo KNN clássico é mostrado logo abaixo.

Fase de Treinamento

Para cada instância **i** avaliada pelo usuário **U** (classe $\omega_{\mathbf{Ui}}$), adicione o exemplo de treinamento **(i,**$\omega_{\mathbf{Ui}}$**)** no perfil do usuário **P**.

Fase de Classificação

Dado uma nova instância **j** no conjunto de exemplos a serem classificados (conjunto de perguntas **Q**):

Seja $\mathbf{i_0}$**, ...,** $\mathbf{i_k}$ as k instâncias do perfil do usuário que são mais similares (próximas) a **j**, a classificação dessa instância com relação a **P** é:

$$\omega_{U_j} = \frac{\sum_{i=0}^{k} \omega_{U_i} * s_{i,j}}{\sum_{i=0}^{k} s_{i,j}}$$	Equação 1 - Equação do KNN. $s_{i,j}$ é a função de similaridade definida na Equação 2.

$$s_{\alpha,\beta} = \frac{\sum_{i=0}^{r} \mu_i * \Gamma_i(\alpha_i, \beta_i)}{\sum_{i=0}^{r} \mu_i}$$	Equação 2 - Equação da função de similaridade. μ_i é o peso do atributo *i*. $\Gamma_i(\alpha_i,\beta_i)$ define a similaridade entre os elementos α e β com relação ao atributo *i*.

A função Γ será discutida mais profundamente na seção 6.2 visto que ela é muito dependente do tipo de atributo considerado.

Alguns problemas são identificados na abordagem KNN. O primeiro é que $s_{i,j}$, a *função de similaridade* (Equação 2), é computacionalmente complexa, ou seja, demanda muito processamento de CPU. Essa complexidade computacional é ainda mais relevante se considerarmos um cenário de um sistema WEB onde existem milhões de usuários e milhares de filmes a serem classificados. Além disso, todas as instâncias no perfil do usuário devem ser comparadas com cada item no conjunto **Q** para confirmar a relevância do mesmo para esse usuário. Suponha que se queira determinar os melhores filmes a partir de um conjunto de perguntas de tamanho **S** para um dado usuário com um conjunto de treinamento (perfil) de tamanho **M**. Nesse cenário, o sistema deverá processar a função Γ, **S*M*R** vezes, onde **R** é o número de atributos de uma instância. Considerando um sistema WEB acessado por milhões de usuários ao mesmo tempo, esse número passa a ser **N*S*M*R**, onde **N** é o número de usuários.

4 FILTRAGEM BASEADA EM META- PROTÓTIPO - FMP

4.1 CONCEITOS

Dados simbólicos são mais complexos que os dados usuais, tendo em vista que os primeiros apresentam variação interna e são estruturados. Diversas são as fontes dos dados simbólicos. Por exemplo, eles podem ser extraídos a partir de interrogações de grandes bancos de dados relacionais ou podem ser expressos a partir do conhecimento formulado por especialistas.. A necessidade de novas ferramentas de análise de dados simbólicos vem crescendo e, nesse sentido, é introduzida a *Análise de Dados Simbólicos* (*Symbolic Data Analysis - SDA*) [16], que estende a *Análise de Dados* em *Análise de Conhecimento* (*Knowledge Analysis*).

SDA é um novo domínio no que se refere à *Descoberta de Conhecimento* (*knowledge discovery*), que tem relação com: *análise multivalorada*, *reconhecimento de padrões*, *banco de dados* e *inteligência artificial*. SDA provê ferramentas específicas para se trabalhar com dados complexos, agregados, relacionais e de alto nível, descritos por variáveis multivaloradas onde as entradas de tabela de dados são conjuntos de categorias ou de números, intervalos ou distribuições de probabilidade associadas a regras e taxonomias.

Os métodos SDA generalizam os métodos clássicos de análise exploratória de dados, como técnicas fatoriais, árvores de decisão, discriminação, regressão, métodos neurais, escalonamento multidimensional, classificação supervisionada, agrupamentos e reticulados conceituais.

4.1.1 Dados Simbólicos

Em *Análise de Dados* clássica, a entrada é uma tabela de dados onde as linhas são as descrições dos indivíduos, e as colunas são as variáveis.

Uma célula dessa tabela contém ou um valor quantitativo simples ou uma categoria.

Não obstante, algumas vezes no mundo real a informação registrada é muito complexa para ser descrita por dados usuais. Esse é o motivo pelo qual diferentes tipos de variáveis simbólicas e dados simbólicos têm sido introduzidos [16]. Por exemplo, uma variável de intervalo corresponde a um intervalo do seu domínio, dado um objeto. Ao mesmo tempo, uma variável categórica multivalorada é definida por um subconjunto do seu domínio. Por fim, uma variável modal, dado um objeto, é representada por uma medida não-negativa (uma freqüência, ou uma distribuição de probabilidade, ou um sistema de pesos).

Uma descrição simbólica de um item é um vetor cujos descritores (atributos) são variáveis simbólicas. Esses descritores simbólicos são coletados em uma *Tabela de Dados Simbólicos*, que são mais complexos do que os dados usuais, visto que cada célula contém não somente um valor único, mas um conjunto de valores, ou um intervalo, ou uma distribuição, etc.

Exemplo. Um segmento de uma imagem (conjunto de pixels) pode ter como descrição simbólica um vetor de dados simbólicos d_s = ([70,80]0.7;[90,120]0.3), onde a característica nível de cinza, representada por valores em [70,80] e [90,120], é modelada por uma variável histograma (ou modal) cujas freqüências relativas dos dois intervalos de valores são 0.7 e 0.3, respectivamente.

Na abordagem FMP, o perfil do usuário é um vetor cujos descritores são variáveis simbólicas modais (ou meta-protótipo). A comparação entre o perfil do usuário e um item a ser recomendado decorre de uma *função de dissimilaridade* apropriada, que é composta por dois componentes: *o componete de posição* e *o componente de conteúdo*.

4.2 Construção do Meta-Protótipo

A construção do perfil do usuário é baseada em um conjunto de itens já avaliados pelo usuário de forma positiva ou não. Com a finalidade de

construir esse perfil, deve-se considerar dois passos no processo de aprendizagem: *pré-processamento* e *generalização*.

4.2.1 Pré-processamento

Cada instância de exemplo é descrita por um vetor de atributos. Essa descrição pode incluir vários tipos de atributos:

i. valor quantitativo simples (contínuo ou discreto),
ii. valor qualitativo simples (nominal ou ordinal),
iii. qualitativo multivalorado (ordenado ou não),
iv. textual.

O objetivo dessa fase de pré-processamento é representar cada item como uma descrição simbólica modal, ou seja, um vetor de vetores de pares (valor, peso). Os itens formam a entrada para a fase de aprendizagem.

Os pares (valor, peso) são formados de acordo com o tipo dos descritores:

i. se o descritor é um valor qualitativo simples, ou qualitativo multivalorado, ou quantitativo simples e discreto, então cada valor é ponderado pelo inverso do cardinal do conjunto de valores do domínio desse indivíduo;
ii. se o descritor é um valor quantitativo simples e contínuo, então ele é mapeado em um histograma comum;
iii. se o descritor é textual então é possível utilizar alguns métodos largamente difundidos em *Recuperação de Informação* como *Centróides* e *TFIDF* [17].

4.2.2 Generalização

Esse passo tem como objetivo a representação do perfil do usuário como um objeto simbólico modal (meta-protótipo). A descrição simbólica de cada perfil de usuário é a generalização da descrição simbólica modal dos seus segmentos.

O meta-protótipo que representa o perfil do usuário é também um vetor de vetores de pares (valor, peso). Os valores, que estão presentes na

descrição de pelo menos um item avaliado pelo usuário no passado, estão também presentes na descrição do perfil do usuário (meta-protótipo). O peso correspondente é a média dos pesos dos respectivos valores presentes nas descrições de cada item isoladamente.

4.3 A COMPARAÇÃO ENTRE META-PROTÓTIPOS

A recomendação de um item para um usuário baseia-se em uma função que compara a descrição simbólica do item com a descrição simbólica de um usuário. A função de comparação mede a diferença em conteúdo e em posição entre um item e a descrição do usuário. A diferença em posição é medida por um componente livre de contexto, enquanto a diferença em conteúdo é medida por um componente dependente de contexto.

Seja $x = (x_1,\ldots,x_p)$ e $u = (u_1,\ldots,up)$ as descrições simbólicas modais de um item e do perfil do usuário, respectivamente, onde $x_j = ((x_{j1},w_{j1}), \ldots, (x_{jk(j)},w_{jk(j)}))$ e $u_j = ((u_{j1},W_{j1}), \ldots, (u_{jm(j)},W_{jm(j)}))$, j = 1, …, p.

A comparação entre o item *x* e o usuário *u* é definida pela seguinte função de comparação:

$\phi(x,u) = \sum_{j=1}^{p} \phi_{cf}(x_j,u_j) + \phi_{cd}(x_j,u_j)$	Equação 3 - A função de comparação.

O componente *livre de contexto* (ϕ_{cf}) da função de comparação é definido por:

$\phi_{cf}(x_j,u_j) = \frac{\left\lvert \overline{X}_j \cap \overline{U}_j \cap (X_j \oplus U_j) \right\rvert}{\left\lvert X_j \oplus U_j \right\rvert}$	Equação 4 - O componente *livre de contexto* (ϕ_{cf}) da função de comparação. $X_j = \{x_{j1}, \ldots, x_{jk(j)}\}$, $U_j = \{u_{j1}, \ldots, u_{jk(j)}\}$ e $X_j \oplus U_j$ é o operador de junção definido por *Ichino* [19].

O componente *dependente de contexto* (ϕ_{cd}) da função de comparação é definido por:

$\phi_{cd}(x_j, u_j) = \frac{1}{2}\left[\sum_{k\,/\,x_k \in X_j \cap \overline{U}_j} w_k + \sum_{m\,/\,u_m \in \overline{X}_j \cap U_j} W_m\right]$	Equação 5 - O componente *dependente de contexto* (ϕ_{cd}) da função de comparação.

5 DEFINIÇÃO DO AMBIENTE EXPERIMENTAL

Durante as adaptações efetuadas nos algoritmos de filtragem de informação para o domínio de filmes foram executados diversos experimentos com objetivo de se identificar a melhor configuração para efeito de comparação entre esses algoritmos. Nesse sentido, é necessária a definição do ambiente experimental utilizado nesses experimentos.

5.1 BASE DE DADOS

A base de dados EachMovie [21] consiste de 72.916 usuários com 2.811.983 avaliações para 1.628 filmes. Os seguintes experimentos são baseados em um subconjunto desse banco de dados que consiste em 22.867 usuários e 1.572.965 com escore de 1 a 5 (1:péssimo, 2:ruim, 3:regular, 4:bom, 5:ótimo) para 638 filmes. A tabela de filmes originais foi combinada com uma segunda tabela contendo as descrições completas dos filmes no idioma Português, pois o banco de dados original do EachMovie não contém qualquer descrição dos filmes e por essa razão não seria possível experimentar a filtragem baseada em conteúdo. Por fim, 100 usuários com no mínimo 150 avaliações foram escolhidos randomicamente para todos os experimentos efetuados nesse trabalho.

5.2 QUALIDADE DAS RECOMENDAÇÕES

A avaliação de Sistemas de Recomendação é ainda uma questão aberta no estudo desses sistemas. Isso se deve principalmente a dois fatores. Primeiro o desenvolvimento de Sistemas de Recomendação é uma inovação recente e ainda não há um consenso sobre uma métrica para avaliar esse tipo de sistema. O outro motivo está associado ao entendimento de qual é o objetivo de um Sistema de Recomendação, ou seja, que tipo de recomendação é usada e quando pode ser considerada uma boa recomendação.

No presente trabalho definimos que o objetivo principal de um sistema de recomendação é gerar uma lista de itens ordenados segundo sua

relevância para o usuário. Nesse caso o usuário deseja navegar pela lista de itens para observar sua ordem de relevância. Por exemplo, quando o usuário busca o melhor item dentre as escolhas, ele busca esse tipo de recomendação.

Depois de definido o objetivo do sistema a ser avaliado é necessário escolher-se uma métrica apropriada para medir a qualidade do sistema.

5.2.1 Métrica

A métrica criada por Breese et al. [22] é usada para calcular a utilidade para o usuário de uma lista ordenada por um sistema de filtragem. A utilidade esperada de uma lista é a probabilidade do usuário ver o item recomendado vezes a sua utilidade.

Em um sistema de filtragem de informação onde o resultado de uma filtragem é retornado em uma lista de itens ordenados pela relevância, é presumível que o usuário irá investigar apenas os primeiros elementos da lista na sua busca por itens relevantes. A métrica Breese tem como principal vantagem levar em consideração essa observação no seu cálculo. A métrica possui uma constante α chamada de *half-life*, que define a posição do item na ordem gerada pelo sistema em que o usuário tem 50% de chance de observar. A utilidade de cada item será a diferença entre a avaliação dada pelo usuário e o valor médio *d* no intervalo de avaliação. A utilidade de uma lista ordenada para o usuário *a* é então:

$R_a = \sum_j \frac{\max(r_{a,j} - d, 0)}{2^{(j-1)/(\alpha-1)}}$	Equação 6 - A utilidade de uma lista ordenada para o usuário *a*.

Na Equação 6 *j* é a posição do item na lista ordenada pelo sistema e $r_{a,j}$ é a avaliação feita pelo usuário ao item que ocupa a posição *j*. Breese utiliza um *half-life* α igual a 5 e acrescenta que a utilização de um valor igual a 10 apresentou pouca diferença nos resultados. Para esse trabalho resolveu-se utilizar um *half-life* igual a 5. O valor final da métrica para o conjunto de usuários testados é calculado como segue:

$R = 100 \frac{\sum_a R_a}{\sum_a R_{\max}}$	Equação 7 - O valor final da métrica Breese para o conjunto de usuários testados. Note que R ∈ [0,100].

Na Equação 7 R_{max} é a utilidade máxima obtida quando todos os itens são ordenados de acordo com a avaliação do usuário. Essa última equação garante que o resultado não dependerá da distribuição nem do número das avaliações dos usuários escolhidos para um experimento.

Resumindo, a métrica proposta por Breese mede a capacidade do sistema ordenar os itens de acordo com sua relevância para o usuário. Além disso, ela dá mais ênfase à precisão dos primeiros itens na ordenação do sistema, os quais terão maior chance de serem apreciados pelo usuário. Dessa forma, a métrica Breese é adeqüada para o objetivo do sistema de recomendação definido para esse trabalho.

Além da métrica apresentada aqui existem outras métricas que poderiam ser utilizadas para medir a capacidade de um sistema de recomendação dependendo do objetivo definido para esse sistema. Métricas como precisão e cobertura, correlacões de valores como Pearson ou de ordens como a Spearsman e Kendall Tau, são outras opções de uma vasta seleção de métricas já usadas para avaliar sistemas de filtragem de informação. Todas as métricas possuem vantagens e significados diferentes. No entanto, Breese é a mais apropriada para o objetivo do sistema de recomendação previamente definido e portanto será adotada para efeito de comparação. Um estudo mais aprofundado das métricas que serviram como base para essa escolha encontra-se em [22].

6 KNN NO DOMÍNIO DE FILMES

O domínio de filmes possui algumas particularidades que precisam ser modeladas nos sistemas de recomendação a fim de que os algoritmos de filtragem de informação atinjam melhores resultados. Precisamos identificar, por exemplo, que tipo de informação o usuário leva em consideração normalmente para avaliar um filme como bom. Alguém pode dizer que o diretor e o elenco são informações importantes para o usuário. Mas não é tão simples assim, precisamos também identificar o quão relevante é um atributo em relação ao outro. A seguir descrevemos alguns experimentos efetuados com objetivo de identificar a relevância dos atributos no domínio de filmes. Nesses experimentos utilizamos o algoritmo KNN descrito na seção 1.

6.1 RELEVÂNCIA DOS ATRIBUTOS

Alguns possíveis atributos do domínio de filmes são: *nome*, *elenco*, *diretor*, *ano*, *país*, *gênero* e *sinopse*. Desses atributos o *nome* é o único que é claramente irrelevante em filtragem de informação baseada em conteúdo, pois são raros os casos que existem filmes com nomes semelhantes como, por exemplo, *Rambo I* e *Rambo II*. Dessa forma, esse atributo acaba diminuindo a qualidade da recomendação em vez de aumentá-la. Por esse motivo esse atributo é ignorado nesse trabalho.

Para medir a relevância dos atributos no domínio de filmes efetuamos o seguinte experimento:

- Para cada usuário foi separado 20% do seu conjunto de itens avaliados para o conjunto de perguntas (teste) e 80% para o conjunto de treinamento (ou seja, o conjunto que compõe efetivamente o perfil do usuário).
- Executou-se o algoritmo KNN variando-se o k em 5 e 11. O valor 5 foi escolhido pois normalmente usa-se um número primo pequeno [13]. O valor 11 foi escolhido para observar a diferença

nos resultados de precisão e de desempenho com um valor para k pequeno e com um valor um pouco maior, no caso 11.

- Utilizou-se a métrica Breese descrita na seção 5.2.1 para medir a qualidade das recomendações.
- A primeira bateria de experimentos (Gráfico 1) consiste em considerar ou não um atributo. Dessa forma um experimento considera todos os atributos enquanto um outro experimento exclui o atributo *elenco*, por exemplo. Os experimentos são definidos abaixo:
    A. Considera todos os atributos
    B. Ignora elenco
    C. Ignora diretor
    D. Ignora gênero
    E. Ignora ano
    F. Ignora país
    G. Ignora sinopse

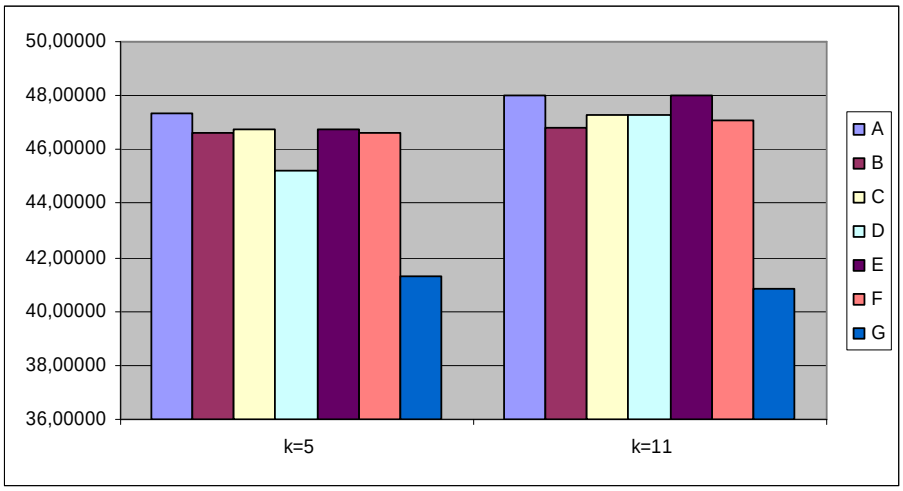

Gráfico 1 – Compara a qualidade do sistema com o algoritmo KNN através da métrica Breese variando os atributos do domínio de filmes.

De acordo com o Gráfico 1 observamos que ignorando o atributo *ano* o KNN, para k igual a 11, mantém um resultado muito pouco inferior ao experimento padrão (experimento A). Para k igual a 5 esse resultado ainda é um pouco menor, cerca de um ponto percentual. Logo, fixaremos os demais experimentos para utilizar todos os atributos exceto o *nome*.

6.1.1 Pesos dos Atributos

Uma outra questão pertinente em se tratando dos atributos é o peso de cada um para se definir uma ordem dos melhores filmes para um determinado usuário. Nesse sentido, utilizamos o procedimento descrito em [25] para se calcular a importância de um atributo baseado em sua entropia. A idéia básica é calcular o nível de generalização de um atributo (objeto simbólico) a partir da sua cobertura no espaço das descrições. Para isso é definido um coeficiente de afinidade. Nesse trabalho utilizamos dois coeficientes de afinidade, um baseado no LOG (Equação 8) e outro baseado na RAIZ (Equação 9).

$w_j = -\sum_{k=1}^{m_j} p_{jk} * \log_{m_j} p_{jk}$	Equação 8 – Coeficiente de afinidade baseado no LOG [27]. w_j é o peso do atributo j, m_j é o número máximo de categorias do atributo j, p_{jk} é o peso da k-ésima categoria do atributo j.

$w_j = \frac{1}{\sqrt{m_j}} \sum_{k=1}^{m_j} \sqrt{p_{jk}}$	Equação 9 – Coeficiente de afinidade baseado na RAIZ [25]. w_j é o peso do atributo j, m_j é o número máximo de categorias do atributo j, p_{jk} é o peso da k-ésima categoria do atributo j.

Dessa forma, os pesos obtidos a partir do coeficiente de afinidade baseado no LOG e na RAIZ são mostrados na tabela abaixo.

Método	**País**	**Gênero**	**Diretor**	**Elenco**	**Ano***	**Sinopse***
LOG	0,339	0,830	0,845	0,921	0,007	0,800
RAIZ	0,416	0,848	0,533	0,824	0,007	0,800
Especialista	0,300	1,000	0,500	0,800	0,300	1,000

Tabela 1 – Possíveis pesos para os atributos. *Como os atributos *ano* e *sinopse* não se aplicam aos experimentos do LOG e da RAIZ seus pesos foram fixados para esses casos.

Após o ajuste dos pesos repetimos o experimento A descrito na seção 6.1 variando os pesos dos atributos com:

A. Pesos equivalentes;
B. Os pesos obtidos pelo LOG;
C. Os pesos obtidos pela RAIZ;
D. Pesos definidos por um especialista do domínio.

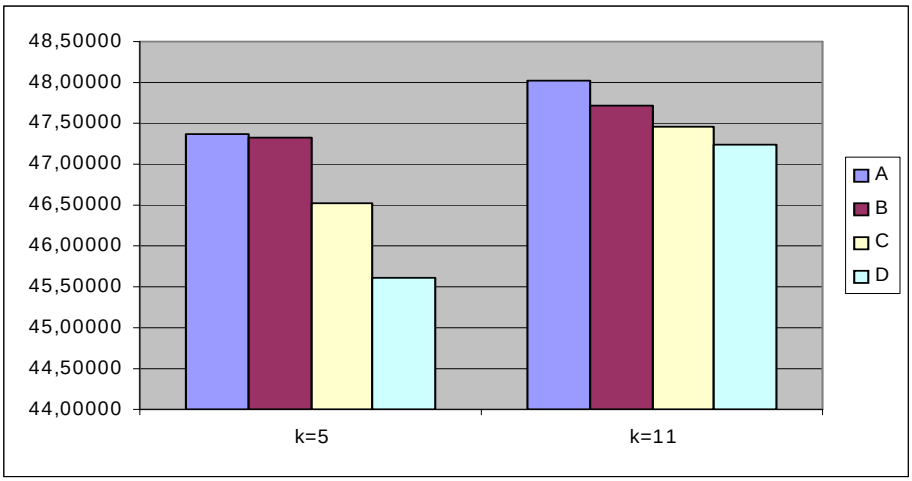

Gráfico 2 - Compara a qualidade do sistema com o algoritmo KNN através da métrica Breese variando os pesos dos atributos do domínio de filmes.

O Gráfico 2 mostra o resultado desses experimentos comparando-os com o experimento A da seção 6.1. A conclusão é que é conveniente manter os pesos dos atributos equivalentes.

Um resumo com os atributos relevantes no domínio de filmes a partir dos experimentos efetuados é mostrado na Tabela 2.

Atributo	Tipo	Exemplo
Gênero	Qualitativo multivalorado	Drama
País	Valor simples qualitativo nominal	EUA
Diretor	Valor simples qualitativo nominal	Steven Spielberg
Elenco	Qualitativo multivalorado	Tom Hanks, David Morse, Bonnie Hunt, Michael Clarke, James Cromwell
Ano	Valor simples qualitativo nominal	1999
Sinopse	Textual	O governo dos EUA oferece um milhão de dólares por alguma informação sobre um perigoso terrorista. John tentará encontrá-lo antes que o FBI o mate.

Tabela 2. Atributos do domínio de filmes.

6.2 A Comparação entre Atributos

A comparação entre os atributos de dois itens depende substancialmente do tipo desses atributos. A tabela abaixo descreve a função

Γ (Equação 2) especificando as regras de comparação para cada tipo de atributo.

Atributo	Tipo	Regra
Gênero, Elenco	Qualitativo multivalorado	$\Gamma_i(\alpha_i, \beta_i) \leftarrow \frac{range(\alpha_i \cap \beta_i)}{range(\alpha_i \cup \beta_i)}$
País, Diretor	Valor simples qualitativo nominal	$\Gamma_i(\alpha_i, \beta_i) \leftarrow \begin{cases} 1 \bullet \alpha_i = \beta_i \\ 0 \bullet \alpha_i \neq \beta_i \end{cases}$
Ano	Valor simples qualitativo nominal	$\Gamma_i(\alpha_i, \beta_i) \leftarrow \max\begin{cases} \rho_i(\alpha_i, \beta_i) \\ 0 \end{cases}$, onde $\rho_i(\alpha_i, \beta_i) \leftarrow 1 - \frac{2 * \lvert \delta(\alpha_i) - \delta(\beta_i) \rvert - 10}{100}$, e $\delta(\mathbf{x})$ é a década do ano **x**.

Tabela 3. Função Γ para o domínio de filmes.

6.2.1 O Atributo Textual

O atributo *sinopse* requer uma atenção especial visto que se trata de um texto repleto de palavras e, portanto, não se aplicam os métodos de comparação utilizados para os atributos anteriores. Dessa forma é necessário utilizarmos algumas técnicas bem definidas de *Recuperação de Informação* [17], como *Centróides* e *TFIDF*.

Basicamente, o conceito de um centróide aproxima-se muito do atributo qualitativo multivalorado. Ou seja, um centróide é definido como um vetor de pares *(palavra, peso)*, onde a *palavra* ou pertence ao texto a partir do qual foi extraído o centróide, ou é o radical de uma ou mais palavras do texto. A segunda opção é uma técnica aplicada para construção de centróides conhecida como *Stemming* [18]. Nesse trabalho assumiremos, para efeito de simplificação, que as palavras do centróide pertencem ao texto a partir do qual foi extraído o mesmo.

Um recurso bastante utilizado na construção do centróide é a *Stoplist* [17], que se trata de um conjunto de palavras consideradas irrelevantes para efeito de cálculo da similaridade entre dois textos e, portanto, tais palavras não são adicionadas ao centróide. Exemplos de palavras irrelevantes são artigos, preposições, adjetivos e pronomes.

O peso de uma palavra em um centróide corresponde à freqüência que essa palavra aparece no respectivo texto. No entanto, é conveniente considerar-se o peso dessa palavra na base de documentos como um todo. O peso de uma palavra, de acordo com *TFIDF*, é definido segundo a Equação 10.

$$p_w^D \leftarrow f_w^D * IDF(w)$$ onde $$IDF(w) \leftarrow \log_2\left(\frac{range(B)}{\varsigma_w^B}\right)$$	Equação 10 - O peso da palavra *w* no texto *D*. *f* corresponde a freqüência de *w* em *D*. *range(B)* é a cardinalidade do banco de textos do sistema. ζ é o total de documentos em *B* que contém a palavra *w*. O termo *IDF* significa *Independent Document Frequency*.

A comparação entre dois centróides é dada pelo co-seno do ângulo formado entre os dois vetores que representam esses centróides. Ou seja, a similaridade entre dois centróides é definida pelo produto interno dividido pelo produto entre a norma de ambos. Dessa forma, a função Γ para o atributo textual é definida segundo a Equação 11.

$$\Gamma(\alpha,\beta) \leftarrow \begin{cases} \dfrac{\sum_{j=1}^{k} p_{w_j}^{\alpha} * p_{w_j}^{\beta}}{\varpi(\alpha,\beta)} \bullet se\ \varpi(\alpha,\beta) \neq 0 \\ 0 \bullet se\ \varpi(\alpha,\beta) = 0 \end{cases}$$ onde $$\varpi(\alpha,\beta) \leftarrow \sum_{j=1}^{k} (p_{w_j}^{\alpha})^2 * \sum_{j=1}^{k} (p_{w_j}^{\beta})^2$$	Equação 11 – A função Γ para o atributo textual.

7 FMP NO DOMÍNIO DE FILMES

Agora que foram introduzidas as características básicas do domínio de filmes é necessário apresentarmos os aspectos desse domínio para a filtragem baseada em meta-protótipo. É importante frisar que a configuração de atributos e pesos definida na seção 6.1 continuará sendo utilizada na abordagem FMP.

7.1 REPRESENTAÇÃO

Como especificado na seção 4.2, cada filme é representado por um vetor de vetores de pares (valor, peso). O meta-protótipo correspondente ao filme da Tabela 2 é mostrado na Tabela 4.

Atributo	Meta-Protótipo (x)
Gênero	(1.0 Drama)
País	(1.0 EUA)
Diretor	(1.0 Steven Spielberg)
Elenco	(0.2 Tom Hanks, 0.2 David Morse, 0.2 Bonnie Hunt, 0.2 Michael Clarke, 0.2 James Cromwell)
Ano	(1.0 1999)
Sinopse	(IDF(eua)∗0.07 eua, IDF(governo)∗0.07 governo, IDF(oferece)∗0.07 oferece, IDF(milhão)∗0.07 milhão, IDF(dólares)∗0.07 dólares, IDF(informação)∗0.07 informação, IDF(perigoso)∗0.07 perigoso, IDF(terrorista)∗0.07 terrorista, IDF(john)∗0.07 john, IDF(tentará)∗0.07 tentará, IDF(encontrá-lo)∗0.07 encontrá-lo, IDF(fbi)∗0.07 fbi, IDF(mate)∗0.07 mate)

Tabela 4 – Meta-protótipo correspondente ao filme mostrado na Tabela 2.

Observe o atributo *Sinopse* na Tabela 4. Como vimos na seção 6.2.1 um centróide pode ser criado a partir do atributo textual e dessa forma pode ser visto como um atributo qualitativo multivalorado como, por exemplo, o *Elenco*.

7.1.1 Representação do Perfil do Usuário

Suponha que os filmes descritos na Tabela 5 tenham sido avaliados por determinado usuário.

Atributo	Filme 1	Filme 2
Gênero	Drama, Guerra	Guerra
País	EUA	EUA
Diretor	Steven Spielberg	Francis Ford Copolla

Elenco	Tom Hanks, Michael Clarke, James Cromwell, Ben Kingsley, Ralph Fiennes	Caroline Goodall, Jonathan Sagall, Liam Neeson, Michael Clarke
Ano	1999	1993
Sinopse	O governo dos EUA oferece um milhão de dólares por alguma informação sobre um perigoso terrorista. John tentará encontrá-lo antes que o FBI o mate.	O FBI tentará matar um perigoso ladrão de banco nesse filme de ação.

Tabela 5 - Exemplos de filmes avaliados por determinado usuário.

De acordo com a seção 4.2.2, o peso correspondente a descrição simbólica de um meta-protótipo de um usuário é a média dos pesos dos respectivos valores presentes nas descrições de cada item isoladamente. A Tabela 6 mostra o meta-protótipo do perfil do usuário exemplificado na Tabela 5.

Atributo	**Meta-Protótipo (*u*)**
Gênero	((0.5 Drama, (0.5+1.0) Guerra) ∗0.5)
País	(((1.0+1.0) EUA) ∗0.5)
Diretor	((0.5 Steven Spielberg, 0.5 Francis Ford Copolla) ∗0.5)
Elenco	((0.2 Tom Hanks, (0.2+0.25) Michael Clarke, 0.2 James Cromwell, 0.2 Ben Kingsley, 0.2 Ralph Fiennes, 0.25 Caroline Goodall, 0.25 Jonathan Sagall, 0.25 Liam Neeson)∗0.5)
Ano	((1.0 1999, 1.0 1993)∗0.5)
Sinopse	((IDF(fbi)∗(0.12+0.07) fbi, IDF(tentará)∗(0.12+0.07) tentará, IDF(matar)∗(0.12+0.07) matar, IDF(perigoso)∗(0.12+0.07) perigoso, IDF(banco)∗0.12 banco, IDF(ladrão)∗0.12 ladrão, IDF(ação)∗0.12 ação, IDF(filme)∗0.12 filme, IDF(eua)∗0.07 eua, IDF(governo)∗0.07 governo, IDF(oferece)∗0.07 oferece, IDF(milhão)∗0.07 milhão, IDF(dólares)∗0.07 dólares, IDF(informação)∗0.07 informação, IDF(terrorista)∗0.07 terrorista, IDF(john)∗0.07 john)∗0.5)

Tabela 6 - O meta-protótipo do perfil do usuário exemplificado na Tabela 5.

7.2 COMPARAÇÃO

A comparação de meta-protótipos no domínio de filmes obedece às regras descritas na seção 4.3. Todavia, algumas adaptações podem melhorar o desempenho e a precisão da resposta em sistemas de recomendação de filmes.

Não é difícil notar que a generalização e a construção de meta-protótipos podem ser feitas de forma incremental, melhorando o desempenho de classificação. Para isso, basta mantermos na memória o número de exemplos usados para construir *u* (*Cardinalidade(u)*) em vez de

calcularmos a média ponderada dos exemplos no passo de generalização. Então, divida ϕ por *Cardinalidade(u)* na comparação com o meta-protótipo de um filme; ou divida ϕ por *Cardinalidade(u)* e também por *Cardinalidade(u')*, onde *u'* é o meta-protótipo de um outro usuário. Além disso, o meta-protótipo do usuário não precisa ser recriado caso um novo item seja avaliado.

7.3 APRIMORANDO A FMP

Nesta seção serão discutidas algumas heurísticas na aplicação da abordagem FMP para o domínio de filmes.

7.3.1 Composição do Perfil do Usuário

Agora que sabemos como representar os itens e o perfil do usuário com meta-protótipo no domínio de filmes, e sabemos como compará-los podemos efetuar experimentos para comparação com o KNN. Todavia precisamos definir quais os itens que entrarão no perfil do usuário. Será que é interessante, por exemplo, construirmos o meta-protótipo com os itens avaliados com notas 1 e 2 pelo usuário?

Para investigar esse problema o seguinte experimento foi executado:

- Para cada usuário foi separado 20% do seu conjunto de itens avaliados para o conjunto de perguntas (teste) e 80% para o conjunto de treinamento (ou seja, o conjunto que compõe efetivamente o perfil do usuário).
- Utilizou-se a métrica Breese descrita na seção 5.2.1 para medir a qualidade das recomendações.
- Executou-se o algoritmo KNN com k igual a 11 para efeito de comparação
- Executou-se a FMP para cada uma das possíveis composições do protótipo do usuário:
    A. O meta-protótipo do usuário é composto por todos os itens avaliados
    B. O meta-protótipo do usuário é composto apenas pelos itens avaliados com nota maior ou igual a 3

C. O meta-protótipo do usuário é composto apenas pelos itens avaliados com nota maior que 3

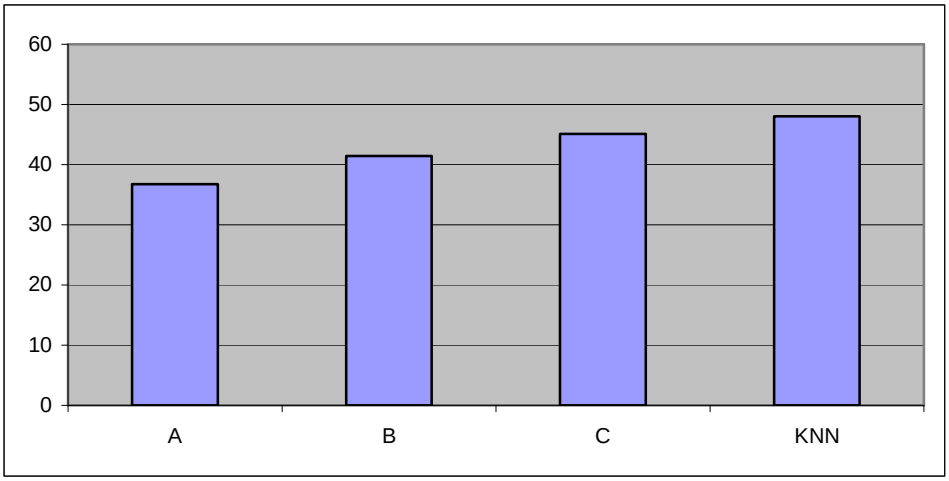

Gráfico 3 – Compara a qualidade do sistema com FMP de acordo com a métrica Breese variando os itens que compõe o perfil do usuário. Mostra também a melhor configuração obtida pelo KNN (seção 1).

De acordo com o Gráfico 3 é mais adequado inserir no meta-protótipo do usuário apenas os itens que foram avaliados com nota acima de 3. No entanto, o resultado obtido está muito aquém do resultado alcançado pela filtragem com KNN.

7.3.2 As Avaliações do Usuário como Heurística

O domínio de filmes possui algumas informações que ainda não foram consideradas, como, por exemplo, as avaliações dos usuários. Dessa forma, questiona-se como utilizar adequadamente as avaliações negativas do usuário, ou seja, os filmes que receberam notas 1, 2 e (talvez) 3. Uma maneira de se empregar a informação trazida por tais avaliações é a construção de um novo meta-protótipo. Portanto, o perfil do usuário passa a ser constituído de dois meta-protótipos: um meta-protótipo positivo ($\boldsymbol{u}^+$) e um meta-protótipo negativo ($\boldsymbol{u}^-$). O processo continua o mesmo, entretanto o meta-protótipo que abrigará um novo item será determinado pela nota dada ao mesmo. Logo, sugere-se a seguinte combinação: itens com notas 1 e 2 pertencem a $\boldsymbol{u}^-$, e itens com notas 4 e 5 pertencem a $\boldsymbol{u}^+$. Já os itens com avaliação 3 possuem três opções:

A. eles não são adicionados em nenhum meta-protótipo pois o usuário não possui qualquer opinião sobre o filme (*don't care*);
B. eles serão incorporados em $\boldsymbol{u}^-$, pois o usuário não gosta do filme e não desejaria assistir a um outro similar;
C. eles serão adicionados a $\boldsymbol{u}^+$, pois o usuário não gosta muito do filme mas ainda assim aceitaria assistir a um outro semelhante.

Dessa forma, a função de comparação torna-se:

$\Phi(x,u) = \frac{\phi(x,u^+) + (1 - \phi(x,u^-))}{2}$	Equação 12 - A função de comparação Φ para dois meta-protótipos, onde ϕ é definida na Equação 3.

As avaliações dos usuários trazem uma outra hipótese que pode ser ainda mais forte do que aquela discutida anteriormente. É fato que uma nota 5 vale muito mais que uma nota 4, e que uma nota 1 é muito ruim enquanto uma nota 2 não é tão ruim assim. Uma forma de modelar esse comportamento do mundo real na abordagem FMP é: i) um item com nota 5 tem uma proporção maior em $\boldsymbol{u}^+$ do que um item avaliado com nota 4, e, equivalentemente, ii) um item com nota 1 tem uma proporção maior em $\boldsymbol{u}^-$ do que um item avaliado com nota 2.

Para investigar as heurísticas discutidas anteriormente, ou seja, o uso de dois meta-protótipos para definição do perfil do usuário e a replicação dos itens de acordo com a nota dada pelo usuário ao item, o seguinte experimento foi executado:

- Para cada usuário foi separado 20% do seu conjunto de itens avaliados para o conjunto de perguntas e 80% para o conjunto de treinamento (ou seja, o conjunto que compõe efetivamente o perfil do usuário).
- Utilizou-se a métrica Breese descrita na seção 5.2.1 para medir a qualidade das recomendações.
- Os itens avaliados com nota 1 serão replicados 3 vezes em $\boldsymbol{u}^-$

- Os itens avaliados com nota 2 serão replicados 2 vezes em $\boldsymbol{u}^-$
- Os itens avaliados com nota 4 serão replicados 2 vezes em $\boldsymbol{u}^+$
- Os itens avaliados com nota 5 serão replicados 3 vezes em $\boldsymbol{u}^+$
- Executou-se a FMP para cada uma das possíveis situações A, B e C discutidas anteriormente.

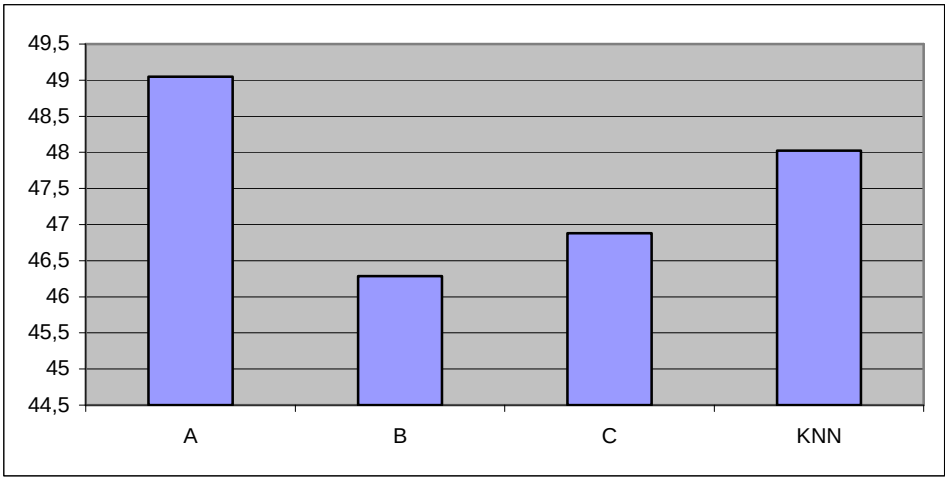

Gráfico 4 - Compara a qualidade do sistema com FMP de acordo com a métrica Breese. No entanto, o perfil do usuário é composto por dois meta-protótipos e ocorre replicação dos itens avaliados. Mostra também a melhor configuração obtida pelo KNN (seção 1).

De acordo com o Gráfico 4 os itens avaliados com nota 3 pelo usuário não são significativos para se determinar a ordem de relevância dos elementos. Isso significa, portanto, que os usuários geralmente não dão importância a esses itens (don´t care), visto que eles são popularmente avaliados como "*mais ou menos*". Dessa forma a configuração utilizada nos próximos experimentos não consideram os itens avaliados com nota 3 para composição do perfil do usuário.

O Gráfico 4 também mostra o quanto melhorou a qualidade da precisão do sistema em comparação a FMP original. Dois foram os motivos dessa melhora: o primeiro foi o uso de dois meta-protótipos e o segundo foi a replicação de itens de acordo com a nota dada pelo usuário. Assim, surgem ainda duas outras questões:

A. Se aumentarmos a proporção dos itens replicados nos meta-protótipos haverá uma melhoria na qualidade da resposta?
B. Qual a importância de cada meta-protótipo? Será que $\boldsymbol{u}^+$ **"**vale mais" do que em $\boldsymbol{u}^-$?

7.3.2.1 A Proporção das Replicações

Para respondermos a questão A efetuamos o seguinte experimento:

- Para cada usuário foi separado 20% do seu conjunto de itens avaliados para o conjunto de perguntas e 80% para o conjunto de treinamento (ou seja, o conjunto que compõe efetivamente o perfil do usuário).
- Utilizou-se a métrica Breese descrita na seção 5.2.1 para medir a qualidade das recomendações.
- Os itens avaliados com nota 1 serão replicados 10 vezes em $\boldsymbol{u}^-$
- Os itens avaliados com nota 2 serão replicados 6 vezes em $\boldsymbol{u}^-$
- Os itens avaliados com nota 4 serão replicados 6 vezes em $\boldsymbol{u}^+$
- Os itens avaliados com nota 5 serão replicados 10 vezes em $\boldsymbol{u}^+$
- Executou-se a FMP

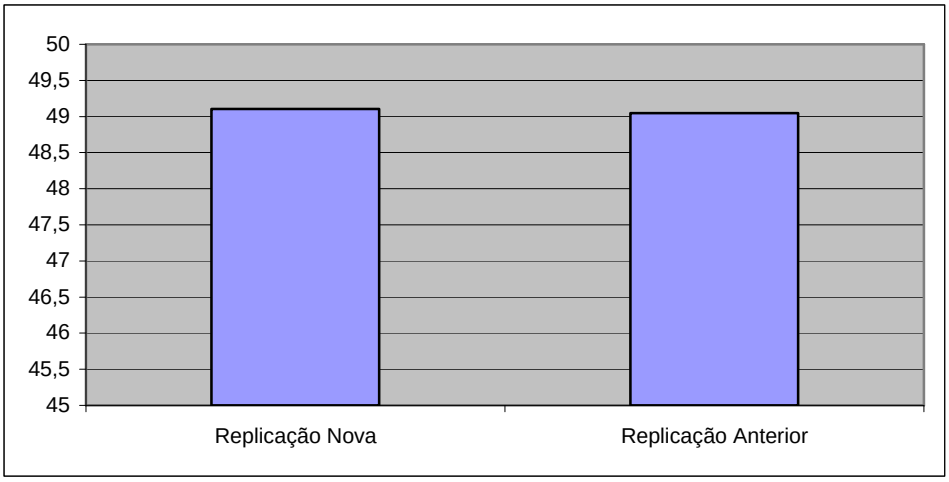

Gráfico 5 – Compara a qualidade do sistema com FMP através da métrica Breese variando a proporção das replicações. A Replicação Anterior está definida na seção 7.3.2.

Ao compararmos os resultados desse experimento com os resultados do experimento da seção 7.3.2 observamos que há uma melhora insignificante quando aumentamos a proporção de inclusão dos itens nos meta-protótipos. Dessa forma, continuaremos com a configuração anterior visto que ela não compromete a velocidade do sistema tanto quanto a configuração desse último experimento.

7.3.2.2 Ponderação dos Meta-Protótipos

A questão B da seção 7.3.2 está diretamente associada a Equação 12. Se observarmos essa equação é fácil verificar que o peso dado a cada meta-protótipo é equivalente. Para verificarmos a importância de cada meta-protótipo para a qualidade de resposta do sistema ponderamos os meta-protótipos em proporções cuja soma é equivalente a 1. Nesse sentido a Equação 12 é redefinida segundo a equação mostrada abaixo.

$\Phi(x,u) = w * \phi(x,u^+) + (1-w)(1-\phi(x,u^-))$	Equação 13 - A função de comparação Φ para dois meta-protótipos, onde $w \in [0, 1]$.

Assim, o seguinte experimento é proposto:

- Para cada usuário foi separado 20% do seu conjunto de itens avaliados para o conjunto de perguntas (teste) e 80% para o conjunto de treinamento (ou seja, o conjunto que compõe efetivamente o perfil do usuário).
- Utilizou-se a métrica Breese descrita na seção 5.2.1 para medir a qualidade das recomendações.
- Executou-se a FMP para os seguintes valores de *w*: 0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9 e 1.0.

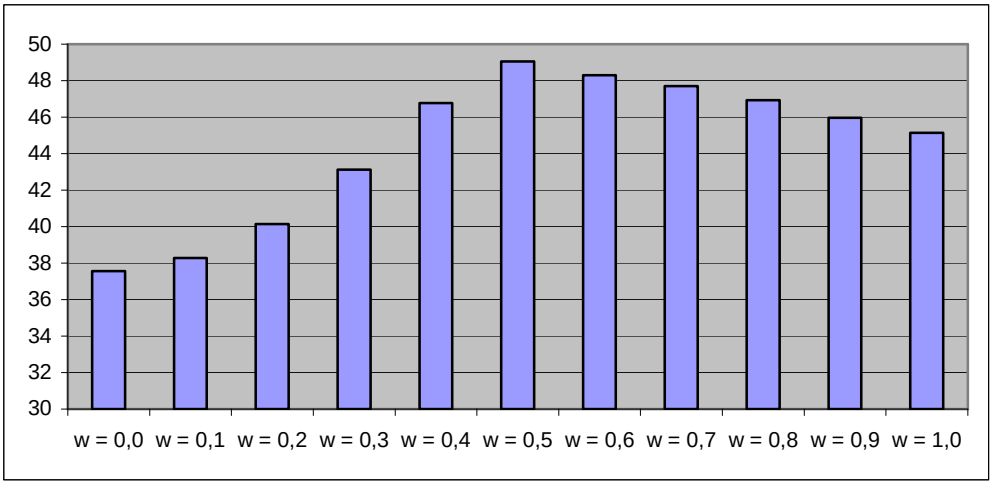

Gráfico 6 – Compara a relevância dos meta-protótipos positivo e negativo para o perfil do usuário.

O Gráfico 6 mostra que é mais conveniente considerar um peso equivalente para os dois meta-protótipos, ou seja, as avaliações positivas são tão importantes quanto as avaliações negativas do usuário para se ordenar os itens mais relevantes para esse usuário. Além disso, observa-se que melhores resultados são obtidos ao se dar um peso maior ao meta-protótipo positivo (*w* igual a 0.6 ou 0.7) do que ao meta-protótipo negativo (*w* igual a 0.4 ou 0.3).

8 EXPERIMENTOS E RESULTADOS

Até o momento efetuamos diversos experimentos com objetivo de melhorar a qualidade da resposta de ambas as abordagens de filtragem discutidas nesse trabalho. A tabela abaixo mostra o resultado, de acordo com a métrica *Breese*, das melhores configurações obtidas para os algoritmos KNN e FMP.

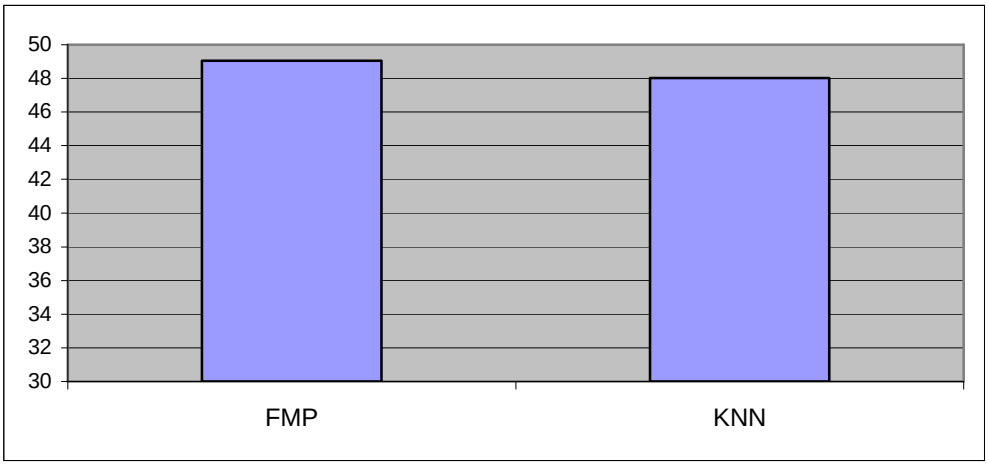

Gráfico 7 – Qualidade dos sistemas com FMP e com KNN de acordo com Breese.

De fato, observamos que a abordagem FMP alcançou melhores resultados na qualidade da resposta. Todavia, a diferença não é tão significativa assim (1 ponto percentual), de forma que, em um sistema de recomendação de filmes real, os algoritmos possam ser considerados equivalentes. No entanto, o próximo experimento mostra que a abordagem FMP é extremamente interessante para esses sistemas de recomendação.

8.1 DESEMPENHO

O desempenho de sistemas de recomendação pode ser melhorado com métodos de indexação no conjunto de exemplos, tais como *K-D Trees* [7]. *K-D Trees* são estruturas de dados dinâmicas e adaptáveis que são muito similares a árvores binárias mas dividem o espaço geométrico de forma adequada para solucionar problemas como busca em intervalos (range searching).

A idéia básica de *K-D Trees* é construir árvores de busca binária com instâncias do conjunto de treinamento como nós, utilizando os atributos das instâncias como chaves em uma seqüência alternada conveniente.

O algoritmo *K-D Tree* descrito em [7] foi adaptado para ser usado em conjunto com o algoritmo KNN apresentado nesse trabalho. Dessa forma, o seguinte experimento tem o objetivo de comparar o desempenho dos algoritmos KNN, K-D Tree e FMP no que diz respeito a velocidade em se gerar as recomendações:

- Foram separados 50 usuários com no mínimo 150 itens avaliados.
- Para cada usuário separou-se os 100 primeiros itens da sua lista de avaliações para formar o conjunto de treinamento. No entanto, variou-se o tamanho do conjunto de treinamentos em 5, 10, 20, 40, 60, 80 e 100 itens.
- Para cada usuário separou-se os 50 últimos itens da sua lista de itens avaliados para o conjunto de perguntas.
- Utilizou-se a métrica Breese descrita na seção 5.2.1 para medir a qualidade das recomendações.
- Utilizou-se 5 e 11 para os valores de k, ou seja, a quantidade de vizinhos mais próximos.
- Foram executados os algoritmos KNN, K-D Tree e FMP e mediu-se o tempo de geração das recomendações em segundos para cada usuário. Esse tempo inclui a construção do perfil do usuário que é próxima de zero no caso do KNN e um pouco mais do que isso no caso de FMP e K-D Tree.

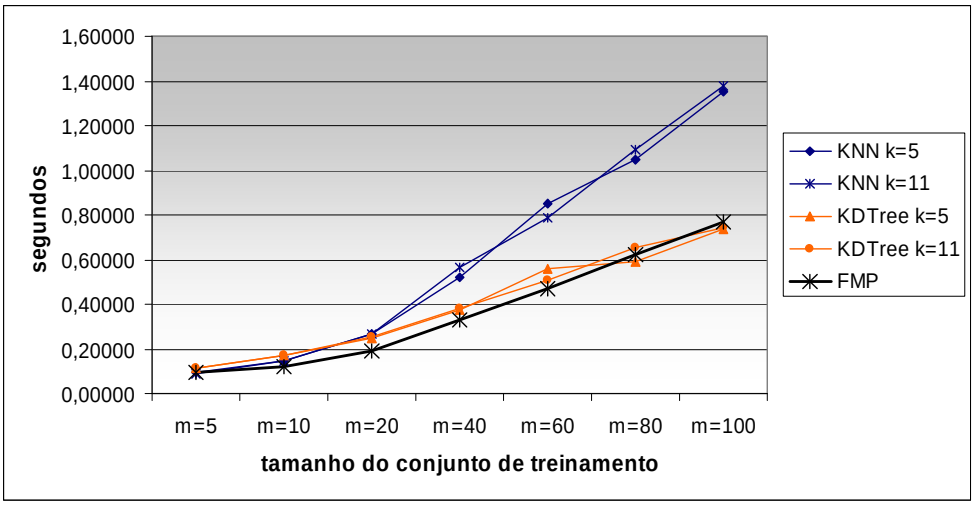

Gráfico 8 – Compara o desempenho dos sistemas de recomendação com KNN, K-D Tree e FMP.

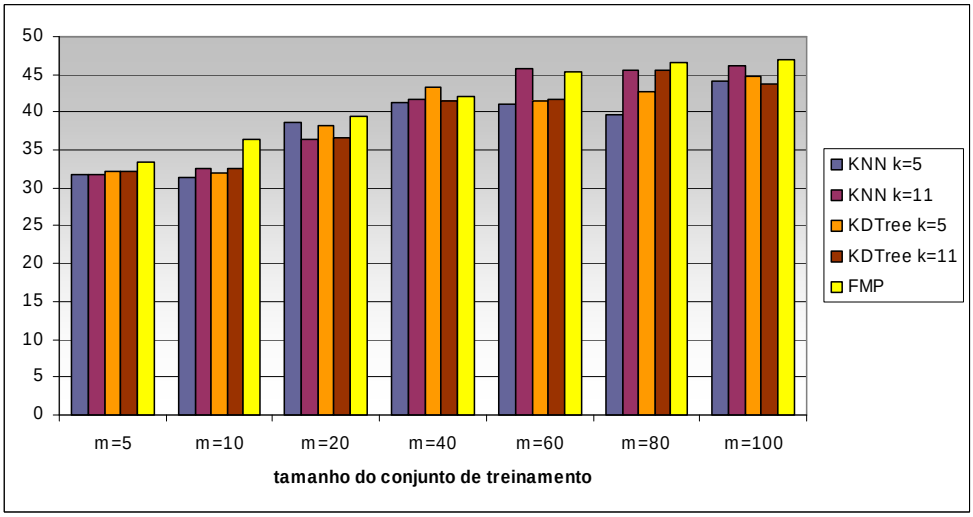

Gráfico 9 – Compara a precisão dos sistemas de recomendação com KNN, K-D Tree e FMP, de acordo com a métrica Breese.

O Gráfico 8 mostra a média dos tempos gastos por cada algoritmo para geração das recomendações. O Gráfico 9 mostra os valores calculados a partir de Breese para os sistemas de recomendação com diferentes algoritmos de filtragem de informação baseada em conteúdo.

Observando os gráficos, verificamos que a classificação de um item do conjunto de perguntas (teste) requer N comparações com o KNN e $N*log_2N$ comparações com o K-D Tree, onde N é o número de itens no perfil do usuário.

Além disso, observa-se que o custo de classificação para o FMP cresce na mesma taxa que o K-D Tree. Isso ocorre porque, na abordagem FMP, o conjunto de treinamento do perfil do usuário é utilizado por completo para efeito de comparação. Como visto anteriormente, o que ocorre na verdade é um condensamento do perfil do usuário devido à mudança de representação do mesmo. Por esse motivo identifica-se uma precisão relativamente maior do FMP com relação ao KNN e ao K-D Tree.

Por manter uma precisão equiparável ao KNN e por alcançar um nível de desempenho acima do KNN e equivalente ao K-D Tree, a abordagem FMP torna-se tão interessante para sistemas de recomendação reais.

9 CONCLUSÕES

Algumas técnicas de filtragem de informação baseada em conteúdo, que são comumente usadas em sistemas de recomendação, possuem problemas de desempenho inerentes. A abordagem dos *k Vizinhos mais Próximos*, por exemplo, possui um custo computacional alto pois o conjunto de treinamento que compõe o perfil de um determinado usuário é composto por todas as instâncias avaliadas pelo mesmo.

Há uma diversidade de trabalhos propondo soluções para esses problemas, mas nenhum deles contempla os requisitos de: i) melhorar o desempenho sem degradar a precisão da resposta; e, ii) ser aplicável ao domínio de filmes, lidando com todos os tipos de atributos desse domínio.

A abordagem FMP, que obteve bons resultados na classificação de um tipo especial de imagens SAR, melhora significativamente o desempenho de sistemas de recomendação de filmes mantendo a precisão da resposta equiparável à precisão do KNN.

10 TRABALHOS FUTUROS

Há uma série de estudos que abordam o problema de desempenho do algoritmo KNN, reduzindo o conjunto de treinamento, tais como *Drop* [13] e *RISE* [14]. Em especial, *Drop* atingiu uma redução da memória de até 12% em alguns domínios. Além disso, é possível reduzir o conjunto de exemplos utilizando *Drop*, seguido por uma mudança de representação com FMP no conjunto de exemplos reduzido. Dessa forma um trabalho futuro bastante interessante é o estudo dessas propostas de melhoria de desempenho combinadas com a abordagem FMP proposta nesse trabalho.

O trabalho [26] propõe um algoritmo de seleção de exemplos do conjunto de treinamento do usuário através de técnicas de aprendizagem ativa. Um trabalho bastante interessante de ser feito é utilizar os exemplos selecionados pelo algoritmo proposto em [26] para compor o meta-protótipo do usuário.

Uma outra extensão desse trabalho é a investigação de novas heurísticas no domínio de filmes na tentativa de melhorar ainda mais a FMP. Adicionalmente, algumas técnicas utilizadas em *Recuperação de Informação* com sucesso como, por exemplo, *Stemming* [18] e *Sinônimos* [17], podem ser implementadas.

Para finalizar, uma outra possibilidade de trabalho futuro é a adaptação da FMP em outros domínios de informação e o estudo comparativo com outras abordagens já utilizadas com sucesso nesses novos domínios.

11 REFERÊNCIAS BIBLIOGRÁFICAS

[1] Shardanand, U. and Maes, P. (1995). Social Information Filtering: Algorithms for Automating 'Word of Mouth'. In Proceedings of CHI'95. Denver, CO.

[2] M. Claypool, A. Gokhale, T. Miranda, P. Murnikov, D. Netes, and M. Sartin. Combining Content-Based and Collaborative Filters in an Online Newspaper. In Proceedings of ACM SIGIR Workshop on Recommender Systems, August 19 1999.

[3] Joshua Alspector, Aleksander Kolcz, and Nachimuthu Karunanithi. Comparing Feature-Based and Clique-Based User Models for Movie Selection. In Proceedings of the Third ACM Conference on Digital Libraries, pages 11-18, 1998.

[4] Marko Balanovic and Yoav Shoham. Content-based, collaborative recommendation. Communications of the ACM, 40(3), March 1997.

[5] Smyth, B. & Cotter, P. (1999) Surfing the Digital Wave: Generating Personalised TV Listings using Collaborative, Case-Based Recommendation. Proceedings of the 3rd International Conference on Case-Based Reasoning, Munich, Germany, 561-571.

[6] Arya S.: Nearest Neighbor Searching and Applications, Ph.D. thesis, University of Maryland, College Park, MD, 1995.

[7] Bentley J., "Multidimensional binary search trees used for associative searching", Communications of the ACM, Vol.18, pp. 509-517, 1975.

[8] Friedman J., Bentley J., Finkel R., "An algorithm for finding best matches in logarithmic expected time", ACM Transactions on Mathematical Software, Vol.3, pp. 209-226, 1977.

[9] Sproull R.L., "Refinements to nearest-neighbor searching", Algorithmica, Vol.6, pp. 579-589, 1991.

[10] De CARVALHO, F.A.T., Souza, R.M.C.M. and Verde, R. (submitted): A Modal Symbolic Pattern Classier.

[11] Cover, T. M., and P. E. Hart (1967). Nearest Neighbor Classifiers. IEEE Transactions on Computers, 23-11, November, 1974, pp. 1179-1184.

[12] Aha, David W., Dennis Kibler, Marc K. Albert (1991). Instance-Based Learning Algorithms. Machine Learning, 6, pp. 37-66.

[13] D. R. Wilson and T. R. Martinez. Reduction techniques for exemplar-based learning algorithms. Machine Learning, 38(3):257-268, 2000.

[14] Domingos, Pedro (1995). Rule Induction and Instance-Based Learning: A Unified Approach. to appear in The 1995 International Joint Conference on Artificial Intelligence (IJCAI-95).

[15] "Ranger - Nearest Neighbor Search in Higher Dimensions" from Stony Brook State University of New York. http://www.cs.sunysb.edu/algorith/implement/ranger/implement.shtml

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

[17] Baeza, Y. and Ribeiro, N. Modern Information Retrieval.

[18] Orengo, V. and Huyck, C. A Stemming Algorithm for the Portuguese Language.

[19] Ichino, M. and Yaguchi, H. (1994): Generalized Minkowsky Metrics for Mixed Feature Type Data Analysis. IEEE Transactions system, Man and Cybernetics, 24, 698-708

[20] Verde, R., De Carvalho, F.A.T. and Lechevallier, Y. (2000): A Dynamical Clustering Algorithm for Symbolic Data, in 25th Annual Conference of the Germany Classification Society, Munich (Germany), 59-72

[21] McJones, P. (1997). EachMovie collaborative filtering data set. DEC Systems Research Center. http://www.research.digital.com/SRC/eachmovie/

[22] Herlocker, Jonathan Lee. Understanding and Improving Automated Collaborative Filtering Systems, cp 3.

[23] Amazon Book Store. http://www.amazon.com, acessado em 22 de Março de 2001.

[24] PTV. http://www.ptvplus.com, acessado em 22 de Março de 2001.

[25] Brito, P. and De Carvalho, F. A. T. (in press) Symbolic Clustering of Constrained Probabilistic Data, in Otto Opitz and Manfred Schwaiger (Eds.), Exploratory Data Analysis in Empirical Research, Springer.

[26] Teixeira, Ivan. Active CP: Um Método de Aprendizagem Ativa em Sistemas de Filtragem Colaborativa, in Master Thesis, Centro de Informática, UFPE (2002).

[27] R. Michalski, J. G. Carbonell, T. M. Mitchel (eds.) (1986). Machine Learning vol II, Los Altos, California: Morgan Kaufmann.

12 APÊNDICE

12.1 GUIA DE ENTRETENIMENTO PERSONALIZADO – GEP

O sistema GEP foi desenvolvido no CESAR (Centro de Estudos e Sistemas Avançados do Recife) em parceria com o CIn-UFPE com o objetivo de tornar-se um protótipo de demonstração da tecnologia de personalização aplicada a um sistema WEB de recomendação de eventos de entretenimento.

O GEP foi idealizado como um sistema capaz de ajudar um usuário a escolher uma opção de entretenimento dentre uma quantidade grande de opções disponíveis. Para isso o GEP é capaz de aprender sobre o perfil do usuário através de avaliações explícitas dos itens, e utilizando o conhecimento que possui sobre ele gerar um guia com as opções de entretenimento mais interessantes. O sistema é, portanto, um sistema de informação sobre eventos de entretenimento, com um cadastro dos eventos disponíveis e a data e hora de sua realização, onde o usuário pode tanto recuperar e consultar os eventos pelo local, data e categoria, como também terá a opção de avaliar os eventos e receber sugestões de eventos segundo o seu perfil.