Universidade Federal de Campina Grande
Centro de Engenharia Elétrica e Informática
Coordenação de Pós-Graduação em Ciência da Computação

Recomendação no Domínio de TV Digital: Uma Arquitetura Baseada na Análise de Descritivos Textuais

Felipe Barbosa Araújo Ramos

Dissertação submetida à Coordenação do Curso de Pós-Graduação em Ciência da Computação da Universidade Federal de Campina Grande - Campus I como parte dos requisitos necessários para obtenção do grau de Mestre em Ciência da Computação.

Área de Concentração: Ciência da Computação
Linha de Pesquisa: Metodologia e Técnicas da Computação

Hyggo Almeida e Angelo Perkusich
(Orientadores)

Campina Grande, Paraíba, Brasil

FICHA CATALOGRÁFICA ELABORADA PELA BIBLIOTECA CENTRAL DA UFCG

R175r Ramos, Felipe Barbosa Araújo.

Recomendação no domínio de TV digital : uma arquitetura baseada na análise de descritivos textuais / Felipe Barbosa Araújo Ramos. – Campina Grande, 2014.

73 f. : il. color.

Dissertação (Mestrado em Ciência da Computação) – Universidade Federal de Campina Grande, Centro de Engenharia Elétrica e Informática. 2014.

"Orientação: Prof. Dr. Hyggo Oliveira de Almeida, Prof. Dr. Angelo Perkusich".

Referências.

1. TV Digital. 2. Sistema de Remendação. 3. Arquitetura de Recomendação. I. Almeida, Hyggo Oliveira de. II. Perkusich, Angelo. III. Título.

CDU 004:621.397.13(043)

"RECOMENDAÇÃO NO DOMÍNIO DE TV DIGITAL: UMA ARQUITETURA BASEADA NA ANÁLISE DE DESCRITIVOS TEXTUAIS"

FELIPE BARBOSA ARAUJO RAMOS

DISSERTAÇÃO APROVADA EM 30/06/2014

HYGGO OLIVEIRA DE ALMEIDA, D.Sc, UFCG
Orientador(a)

ANGELO PERKUSICH, D.Sc, UFCG
Orientador(a)

LEANDRO BALBY MARINHO, Dr., UFCG
Examinador(a)

MARCOS RICARDO ALCÂNTARA MORAIS, D.Sc, UFCG
Examinador(a)

CAMPINA GRANDE - PB

Resumo

Os Sistemas de Recomendação vêm sendo utilizados em diversos domínios de aplicação, mais recentemente, no domínio de TV (TV Digital, *Smart* TV, etc). Várias abordagens podem ser utilizadas para recomendar itens ou *tags*, baseadas principalmente no *feedback* dos usuários. Porém, no domínio de TV Digital a obtenção de *feedback* explícito é feita usualmente por meio do controle remoto, que deve ser evitado para maximizar a experiência do usuário ao ver TV. Além disso, como no contexto de *Smart* TV vários tipos de itens podem ser recomendados (filmes, músicas, livros, etc) a recomendação deve ser genérica o suficiente para se adequar a diferentes conteúdos. Portanto, para contornar o problema de obtenção de *feedback* e gerar recomendações que possam ser usadas por diferentes aplicações de *Smart* TV, neste trabalho é proposta uma arquitetura de recomendação baseada na extração e classificação de termos por meio da análise de descritivos textuais de programas de TV presentes nos guias de programação. A fim de validar a solução proposta, um protótipo usando um conjunto de dados real foi desenvolvido, mostrando que a partir dos termos recomendados é possível gerar recomendações finais para diferentes aplicações de *Smart* TV.

Abstract

Recommendation systems have been used in several application domains, most recently for TV (Digital TV, Smart TV, etc). Several existing approaches can be used to recommend items or tags, mainly based on user feedback. However, in the Digital TV domain, user feedback has to be done generally by using the remote control, which should be avoided to improve user experience. Moreover, in the Smart TV environment several types of items can be recommended (movies, music, books, etc). Thus, the recommendation should be generic enough to suit to different content. Therefore, to solve the problem of acquiring feedback and still generate personalized recommendations to be used by different Smart TV applications, this work proposes a recommendation architecture based on the extraction and classification of terms by analyzing the textual descriptions of TV programs present on electronic programming guides. In order to validate the proposed solution, a prototype using a real dataset has been developed, showing that from the recommended terms is possible to generate final recommendations for different Smart TV applications.

Agradecimentos

Agradeço primeiramente a Deus por estar sempre ao meu lado e guiar meus caminhos, dando-me paz, saúde e todas as condições possíveis para exercer as tarefas do dia a dia.

Aos meus pais Adeziva Barbosa e Fernando Ramos e ao meu irmão Fernando, os quais me dão apoio e incentivo em todos os momentos da minha vida.

À minha namorada Monik, pelo apoio e incentivo fornecido, principalmente, durante a finalização deste documento de dissertação.

Aos orientadores Hyggo Almeida e Angelo Perkusich, que me auxiliaram durante todo o período do mestrado.

Aos professores e funcionários do Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Campina Grande.

Aos amigos e colegas do mestrado e do laboratório Embedded pelo apoio constante durante toda essa trajetória, em especial Antonio Alexandre Moura Costa e Reudismam Rolim de Sousa.

À Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) pelos investimentos na pós-graduação.

Por fim, a todos que de alguma forma contribuíram para a realização desse trabalho.

Conteúdo

Lista de Símbolos

BC - *Recomendação Baseada em Conteúdo*

TVD - *TV Digital*

EPG - *Guia Eletrônico de Programação*

FC - *Recomendação Baseada em Filtragem Colaborativa*

FM - *Fatoração de Matriz*

IMBHN - *Modelo Indutivo Baseado em Redes Bipartidas Heterogêneas*

IPTV - *Televisão sobre Protocolo de Internet*

SR - *Sistema de Recomendação*

SVD - *Decomposição em Valores Singulares*

TF-IDF - *Frequência do Termo - Frequência Inversa no Documento*

Lista de Figuras

Lista de Tabelas

Capítulo 1

Introdução

Neste capítulo são apresentados o contexto geral do trabalho acerca da utilização de sistemas de recomendação (SRs), o escopo do problema que motivou a pesquisa, os objetivos almejados, a relevância do trabalho e, por fim, a estrutura do documento de dissertação.

1.1 Contextualização

Com o crescimento relevante nos últimos anos dos meios de comunicação, como Internet e televisão, o acesso à informação tornou-se cada vez mais fácil. Abriu-se então um leque de opções para os usuários, que podem realizar diversos tipos de transações, como compras na Internet [11], buscas por músicas [39], por notícias [37], etc. Porém, dada a grande quantidade de opções de conteúdos disponíveis, tornou-se difícil para os usuários encontrar informações relevantes [1]. Nesse contexto, os sistemas de recomendação apresentam-se como ferramentas importantes, pois auxiliam os usuários na escolha de itens ou conteúdos adequados a suas preferências.

Os sistemas de recomendação trabalham com o conceito de itens e usuários, em que "item" é usado para denotar o que é recomendado para o "usuário" [45]. Para gerar recomendações acuradas, em geral, os SRs identificam as principais características dos usuários e dos itens para criar seus perfis. Essas informações são coletadas de fontes de dados e podem ser textuais, visuais, etc.

Os sistemas de recomendação podem ser utilizados em diferentes domínios de aplicação, auxiliando os usuários em suas escolhas [45]. Em TV Digital (TVD), por exemplo, os SRs

estão ganhando cada vez mais importância [10], uma vez que existe uma quantidade elevada de canais e programas ofertados [40]. Além disso, os programas apresentam conteúdos diversificados, dificultando a tomada de decisão dos usuários. Em TVD, em geral os perfis dos usuários são gerados por meio da análise de seus históricos, já os perfis dos itens são usualmente gerados por meio da extração das informações contidas no Guia Eletrônico de Programação (EPG) [49], que fornece detalhes acerca dos conteúdos dos programas (Figura 1.1)[1]. O EPG, geralmente, provê uma lista de programas e canais de TV disponíveis por um período de pelo menos 36 horas [3], ajudando a fazer a TVD mais flexível e fornecendo uma forma mais fácil de acesso às informações desse domínio [28]. Dentre as informações dos programas disponibilizadas pelo EPG, têm-se: título, descrição, categorias, dentre outras [17].

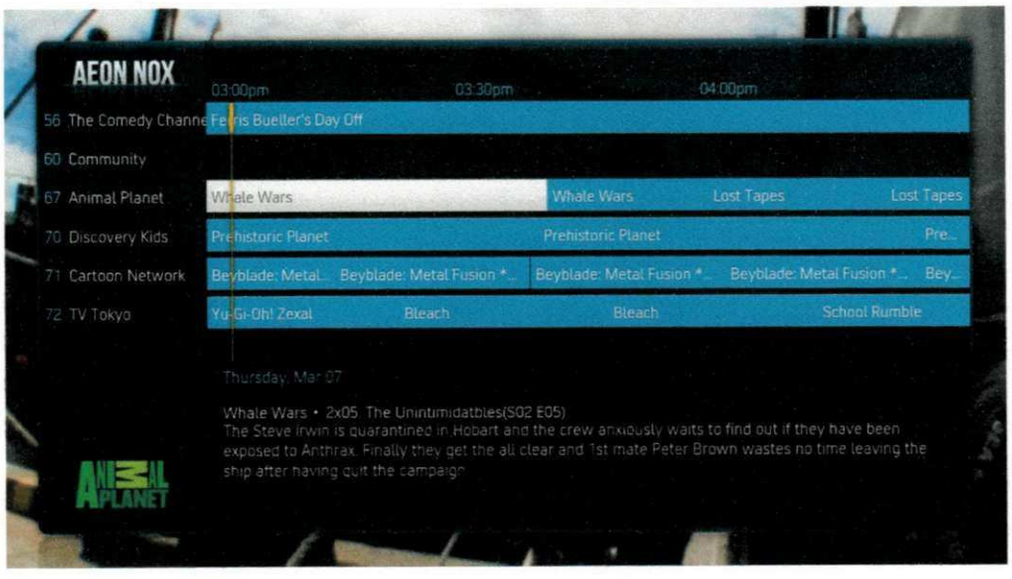


Figura 1.1: Exemplo de guia eletrônico de programação.

Além da diversidade de conteúdo da TV Digital, com o advento das *Smart* TVs, que integram características da Internet e Web 2.0 em aparelhos de TVD [10], a gama de opções para usuários se tornou ainda maior. Por meio das aplicações de *Smart* TV, os usuários passaram também a ter acesso as informações de provedores de conteúdo baseados na Internet, como *Netflix*[2], *YouTube*[3], etc [10]. Assim, como as *Smart* TVs estão se tornando cada vez

[1]Imagem retirada da Internet

[2]www.netflix.com

[3]www.youtube.com

mais populares [35] e oferecem aos usuários uma diversidade de conteúdo multimídia além do conteúdo de TV Digital especificado no EPG, os sistemas de recomendação se mostram ainda mais relevantes nesse contexto, pois diante de tantas escolhas possíveis os usuários podem sentir dificuldade em encontrar conteúdo interessante ao seu perfil [10].

1.2 Problemática

Sistemas de recomendação em geral são voltados para a recomendação de itens específicos, por exemplo, no domínio de *e-commerce* a recomendação final é um produto à venda [36], no domínio de serviços de *streaming* é um filme [6], etc. Assim, abordagens de recomendação de itens ou predição de notas são indicadas nesses casos. Entretanto, existem domínios em que os itens recomendados são oriundos de diferentes aplicações, como nas *Smart* TVs (Figura 1.2[4]), que permitem a conexão com a Internet e a instalação de aplicações de diferentes contextos [16], tornando necessária a geração de recomendações para cada aplicação, já que seus itens finais diferem entre si.

Figura 1.2: Exemplo do domínio de *Smart* TV com aplicações de conteúdos diferentes.

[4]Adaptação de imagem retirada da Internet

Em *Smart* TVs diferentes aplicações podem recomendar itens de tipos distintos e com características diferentes, como recomendação de usuários em redes sociais, de filmes, de notícias, de produtos em promoção, de canais e programas de TV, de vídeos de esportes, de receitas de comida, de músicas, de concursos, de ofertas de compras coletivas, entre outros. Logo, dada a heterogeneidade das aplicações, fazer uso de abordagens de recomendação de itens no domínio de *Smart* TV implica no desenvolvimento de sistemas de recomendação para diferentes contextos ou módulos desses SRs, baseando-se nas características dos itens. Além disso, para diferentes contextos são criados diferentes perfis dos usuários. Por exemplo, na Figura 1.3 observa-se o surgimento de uma nova aplicação de TV que necessita de recomendações. Assim, na ausência de uma abordagem que gere recomendações independentes de contexto, é necessário o desenvolvimento de SRs para aplicações de contextos diferentes, baseando-se em suas especificações, e acarretando os seguintes problemas:

- Há um desvio de foco do desenvolvimento da aplicação para a geração do SR correspondente;
- Tarefa repetitiva e custosa. Por exemplo, componentes dos SRs como os geradores de perfis dos usuários são criados para cada aplicação baseando-se na interação dos usuários com os itens específicos dessas aplicações. Porém, todos os usuários de *Smart TV* interagem com itens em comum, programas de TV, independente das aplicações que utilizam. Além disso, pode ser necessário o estudo de formas diferentes de obtenção de *feedback* implícito para os diferentes tipos de itens.
- São necessários conhecimentos específicos para a criação de SRs. Por exemplo, um SR baseado em conteúdo tem suas raízes na recuperação de informação e na filtragem de informação [1]. Além disso, a arquitetura de um SR é composta de vários componentes. Por exemplo, em um SR baseado em conteúdo têm-se: analisador de conteúdo, que pré-processa os dados para extrair informação relevante [38]; aprendiz de perfil, que coleta informação representativa para gerar os perfis dos usuários [38]; componente de filtragem, que explora o perfil do usuário para gerar sugestões de itens [38]. Logo, a geração de um SR não se mostra uma tarefa simples, tendo em vista todos os conhecimentos e processos citados. Portanto, o auxílio de um especialista da área pode ser necessário, caso contrário, pode-se levar muito tempo na fase de desenvolvimento

do SR e consequentemente na finalização e inserção da aplicação no mercado (*time to market*);

- Caso não haja tempo ou conhecimentos suficientes para o desenvolvimento do SR e o uso de recomendação seja um requisito importante, a ausência desta funcionalidade pode comprometer o objetivo principal da aplicação. Por exemplo, no desenvolvimento de uma aplicação que sugere filmes para usuários, o uso de SR se mostra essencial. Portanto, caso o desenvolvedor não tenha conhecimentos suficientes para gerar o SR de forma correta, a aplicação perde seu objetivo principal.

- Para cada aplicação será utilizada uma base de usuários diferente. Assim, para uma aplicação recém-criada é necessário um período de uso para que as informações acerca dos usuários sejam suficientes para a geração das recomendações a partir de seus perfis.

Assim, o domínio de *Smart* TV requer uma abordagem que garanta interoperabilidade das recomendações, característica de algumas abordagens de recomendação de *tags*, que trabalham com a sugestão de palavras-chave que melhor descrevem os itens ou usuários [50]. Porém, em TV Digital a atribuição explícita de *tags* ou *feedback* (modelo baseado no usuário [50]) é restrita por requisitos de experiência do usuário, que demandam o uso do controle remoto o mínimo possível [3].

Logo, a investigação de formas implícitas de aquisição de informação mostra-se importante em sistemas de recomendação aplicados à TV Digital/*Smart* TV, pois a obtenção de *feedback* explícito por parte dos usuários é restrita por requisitos do domínio [26]. Um exemplo de aquisição implícita consiste na extração de termos dos descritivos textuais dos itens (programas de TV), em que um termo representa uma palavra-chave do item e pode ser utilizado na geração de recomendações para as diversas aplicações de TV Digital/*Smart* TV.

Porém, apenas a extração dos termos não é suficiente para representar os itens de forma adequada, é preciso também atribuir semântica para os termos de forma que eles sejam mais significativos. Além disso, para que as recomendações sejam utilizadas por aplicações diferentes é preciso personalizá-las, por exemplo, classificando em categorias predefinidas os termos extraídos e recomendados, de modo que cada aplicação possa definir suas categorias relacionadas e termos diferentes sejam recomendados.

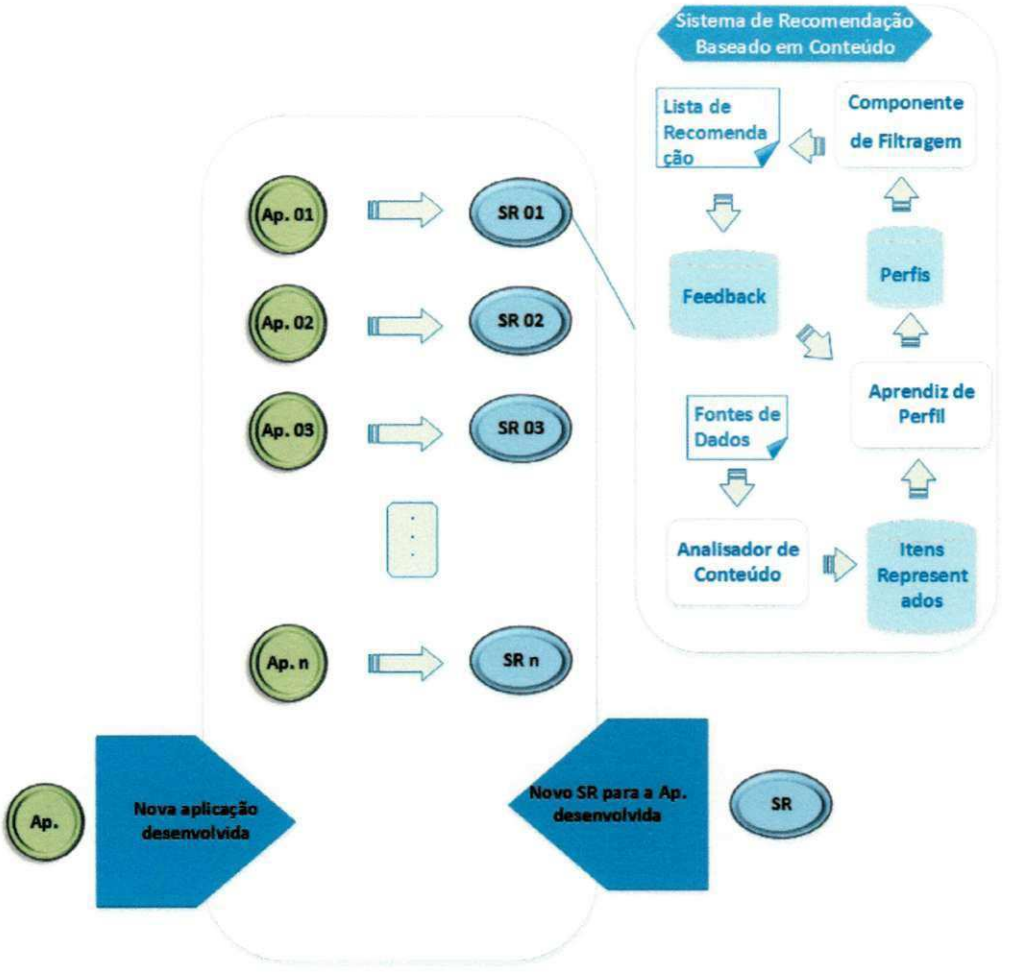


Figura 1.3: Exemplo do desenvolvimento de uma nova aplicação no domínio de *Smart* TV em uma arquitetura baseada na recomendação de itens (para cada aplicação é necessário o desenvolvimento de um sistema de recomendação diferente).

1.3 Objetivos

Neste trabalho, tem-se como principal objetivo propor uma arquitetura de recomendação de termos para o domínio de *Smart* TV, em que as recomendações geradas sejam utilizadas por aplicações de diferentes contextos.

A fim de alcançar o objetivo principal, alguns objetivos específicos foram listados:

1. Identificar forma de obtenção de *feedback* implícito para o domínio de TV Digital;
2. Identificar forma de mineração de informações textuais do EPG;

3. Propor abordagem de recomendação independente de contexto (*cross-domain recommendation*)

4. Facilitar o processo de geração de recomendação final para diferentes tipos de aplicações de *Smart* TV;

5. Avaliar da solução proposta.

Visando avaliar a arquitetura proposta, um protótipo foi desenvolvido utilizando uma base de dados real que constatou a viabilidade da utilização da recomendação de termos por aplicações com itens diferentes, no caso, filmes e livros.

1.4 Relevância do Trabalho

O trabalho colabora na área de sistemas de recomendação aplicados ao domínio de TV Digital/*Smart* TV, e sendo essa uma área de atuação que ainda carece de estudos mais elaborados [28], a solução apresenta relevância significativa, pois propõe uma arquitetura de recomendação para TVD.

Além disso, diferente dos trabalhos encontrados na literatura [2, 3, 10, 32, 40], que trabalham geralmente com a recomendação de programas de TV, a arquitetura proposta visa contornar restrições encontradas no domínio de TV Digital para gerar recomendações de termos utilizáveis por aplicações diversas de *Smart* TV, sem a necessidade de um *feedback* explícito do usuário.

Por fim, este trabalho tem relevância para o avanço nas pesquisas do Laboratório de Sistemas Embarcados e Computação Pervasiva (Embedded) no domínio de TV Digital/*Smart* TV.

1.5 Estrutura da Dissertação

Os capítulos restantes que compõem este documento estão estruturados da seguinte forma:

Capítulo 2: Fundamentação Teórica. Apresenta definições gerais dos temas abordados nessa dissertação, que servem para dar embasamento teórico aos leitores acerca de sistemas de recomendação em geral e aplicados ao domínio de TV Digital, abordagens de obtenção

de *feedback* para geração de perfis de usuários, mineração de texto e redes bipartidas heterogêneas que são usadas na fase de classificação dos termos;

Capítulo 3: Trabalhos Relacionados. Discute os trabalhos relacionados na área de sistemas de recomendação e categorização de texto;

Capítulo 4: Solução Proposta. Apresenta as fases realizadas para a obtenção da arquitetura proposta para extração, classificação e recomendação de termos extraídos dos descritivos textuais de programas de TV;

Capítulo 5: Validação do Trabalho. Detalha a forma de coleta de informações para gerar a base de dados utilizada na validação, as ferramentas utilizadas ao longo do trabalho e o protótipo desenvolvido para validar a solução proposta;

Capítulo 6: Considerações Finais. Apresenta as conclusões e contribuições do trabalho, como também, as limitações encontradas e os trabalhos futuros que podem ser desenvolvidos a partir desta pesquisa.

Capítulo 2

Fundamentação Teórica

Neste capítulo são apresentados os conceitos gerais necessários para o entendimento do trabalho. Nas próximas seções são apresentadas considerações acerca de: sistemas de recomendação em geral e aplicados ao domínio de TV Digital, mineração de texto, formas de obtenção de *feedback* implícito e explícito e redes bipartidas heterogêneas.

2.1 Sistemas de Recomendação

Os sistemas de recomendação representam uma área de estudo interessante [1], pois existem inúmeras aplicações e sistemas que lidam com sobrecarga de informação e necessitam ajudar seus usuários em suas escolhas. Um exemplo do uso de sistemas de recomendação é o site *Amazon*[1] [36], que faz uso dessas ferramentas para proporcionar recomendações personalizadas de produtos para seus clientes com o intuito de maximizar suas vendas.

SRs representam uma sub-área de aprendizagem de máquina e podem ser classificados em três categorias: recomendação baseada em conteúdo, recomendação baseada em filtragem colaborativa e abordagens híbridas [1].

Nos sistemas de recomendação baseados em conteúdo (BC) as recomendações são geradas de acordo com as preferências dos usuários no passado. Então, a relevância de um item i para um usuário u é avaliada baseando-se na relevância de itens similares a i que foram avaliados anteriormente pelo usuário u [1]. Na Figura 2.1 [38] é apresentada uma arquitetura de alto nível de um sistema de recomendação baseado em conteúdo.

[1] http://www.amazon.com/

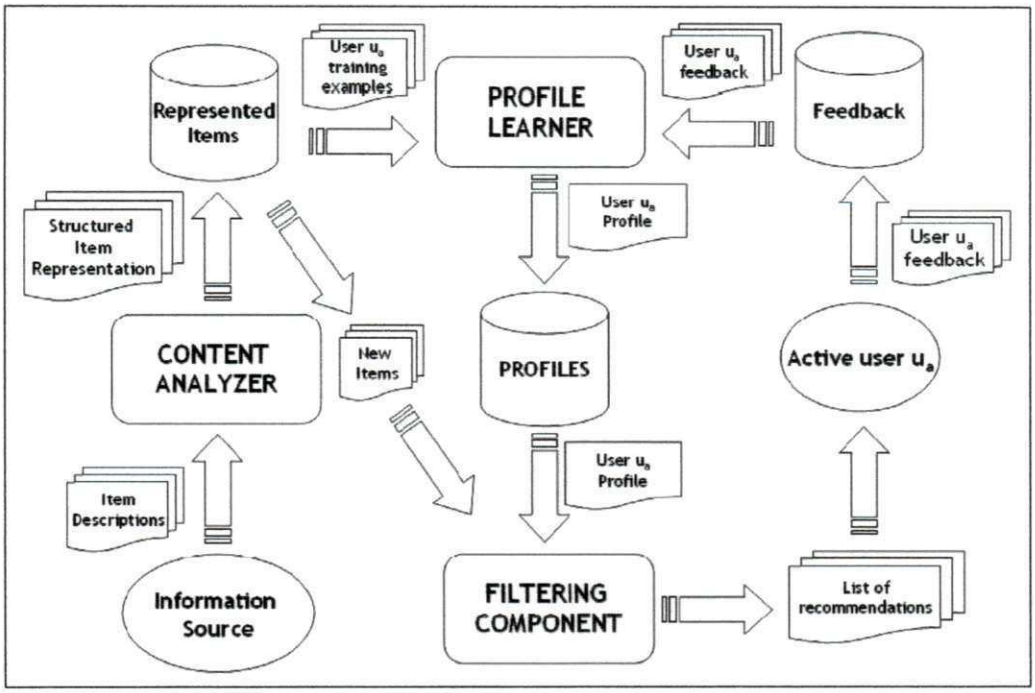


Figura 2.1: Arquitetura de alto nível de um sistema de recomendação baseado em conteúdo.

Sistemas de recomendação baseados em conteúdo têm relação com áreas como recuperação de informação e filtragem de informação [1], pois o processo básico dessa abordagem de recomendação consiste em combinar atributos dos itens e dos perfis dos usuários [38], que são extraídos dos conteúdos disponibilizados, por exemplo a sinopse de um filme, descrição de um programa de TV, etc. Apesar de ser uma técnica conhecida, ao utilizá-la isoladamente alguns problemas são encontrados, como:

- **Superespecialização**, que consiste em recomendar apenas conteúdos parecidos com o perfil do usuário, não havendo "novidades" [1];
- **Novo Usuário**, que consiste no surgimento de um novo usuário na base que ainda não interagiu com itens o suficiente para que seu perfil seja conhecido e recomendações confiáveis sejam geradas [1].

Em contrapartida, nos sistemas de recomendação baseados em filtragem colaborativa as recomendações são geradas baseando-se na suposição que usuários com gostos similares avaliam itens similarmente [51]. Por isso, um passo importante na abordagem colaborativa é identificar grupos de usuários com interesses parecidos [15].

Os SRs baseados em filtragem colaborativa podem ser divididos em duas classes, baseados em memória e em modelo [1, 4]. Também conhecidas como baseadas em vizinhança [15], nas abordagens baseadas em memória as avaliações dos usuários obtidas anteriormente para cada item são armazenadas e utilizadas diretamente na predição das avaliações [15], ou seja, basicamente, uma avaliação não conhecida de um item *i* para um usuário *u* é computada por meio da agregação dos valores obtidos para o item *i* a partir de usuários similares a *u* [1]. Um exemplo de agregação pode ser a média dos valores obtidos. Nas abordagens baseadas em modelo as avaliações obtidas anteriormente são utilizadas para gerar um modelo de predição [1]. Basicamente, as interações entre usuários e itens são modeladas utilizando fatores latentes, que representam suas características [15]. Uma técnica conhecida de abordagens baseadas em modelo é a Fatoração de Matriz (FM) [30]. Visando melhorar a acurácia das recomendações e superar limitações encontradas nas abordagens padrões de FC, as duas classes também podem ser combinadas [41].

Os SRs baseados em filtragem colaborativa sofrem com problemas como:

- **Esparsidade**, que ocorre quando o número de avaliações obtidas é muito menor que o número de avaliações que devem ser preditas [1];
- **Novo Item**, que consiste no surgimento de um item na base que ainda não recebeu avaliações de usuários suficientes para que recomendações confiáveis sejam geradas [1].

Por fim, na abordagem híbrida as recomendações são geradas por meio de uma combinação entre os dois tipos de sistemas de recomendação anteriores (BC e FC) [13]. Alguns sistemas de recomendação propostos utilizam a abordagem híbrida para evitar certas limitações que as outras abordagens apresentam quando aplicadas isoladamente [1].

2.2 Sistemas de Recomendação Aplicados ao Domínio de TV Digital

Com o desenvolvimento da tecnologia digital, o conteúdo disponível para os usuários de TV está cada vez mais volumoso [31]. O número de canais, programas e a variedade de categorias que podem ser acessados demandam aos usuários um tempo grande para identificar

conteúdo relevante [10]. Por isso, os sistemas de recomendação vêm se mostrando bastante promissores nesse domínio de aplicação [28], pois possibilitam a identificação e caracterização de programas e usuários de forma adequada, facilitando a busca por conteúdos.

Tendo em vista a importância da utilização de sistemas de recomendação aplicados ao domínio de TV Digital, trabalhos que propõem pesquisas nessa área estão cada vez mais frequentes [2, 10, 32, 40]. Além disso, grandes empresas fabricantes de TV e provedoras de conteúdo estão adotando essas ferramentas [10]. Principalmente após as *Smart* TVs, que além do conteúdo televisivo oferecem aos usuários uma série de funcionalidades, como acesso à Internet [32], instalação de aplicativos [16], entre outras.

Em TVD, em geral os perfis dos usuários são gerados por meio da análise de seus históricos. Em contrapartida as informações utilizadas para a criação dos perfis dos itens são geralmente coletadas do EPG [49], que apresentam informações textuais sobre os programas ofertados pelos provedores de televisão, como título, horário de início e de fim, categorias, descrição, dentre outras. Além disso, vídeo, som e padrões de comportamento dos usuários também são opções de fontes de informação.

2.3 Formas de Obtenção de *Feedback*

A obtenção de *feedback* do usuário caracteriza-se uma tarefa importante no domínio de sistemas de recomendação, pois é necessário identificar as preferências dos usuários para que recomendações acuradas sejam geradas [45], ou seja, encontrar relações entre usuários e itens [26]. Para obter essas informações dos usuários duas técnicas de aquisição de *feedback* são bastante utilizadas: *feedback* explícito e implícito [38, 54].

O *feedback* explícito caracteriza-se na ação do usuário atribuir avaliações, críticas, comentários, diretamente para os itens [26, 34]. Plataformas multimídias como *Netflix* e *TiVo*[2] fazem uso de técnicas de obtenção de *feedback* explícito [26]. Existem várias abordagens de *feedback* explícito, como: a avaliação de um item pelo usuário como relevante ou irrelevante [38], a utilização de escalas numéricas ou simbólicas (1-5, número de estrelas, etc) [38], o uso de comentários textuais em itens [38], etc. Porém, apesar de oferecer acurácia significativa por representar uma opinião direta do usuário [29], o uso de *feedback*

[2]http://www.tivo.com/

explícito apresenta algumas limitações, pois nem todos os usuários estão dispostos a oferecer *feedback* explícito, uma vez que essa tarefa demanda tempo [54]. Além disso, usuários muitas vezes atribuem avaliações aleatórias [34], alguns ambientes apresentam limitações que impossibilitam a atribuição explícita de avaliações [26], etc. Portanto, como alternativa à atribuição explícita de avaliações, é possível inferir as preferências dos usuários por meio do *feedback* implícito [26], que reflete suas opiniões de forma indireta por meio da observação do comportamento dos usuários [26].

Enquanto o *feedback* explícito expressa a preferência dos usuários, o *feedback* implícito representa confiança [26], e uma vez que o usuário permite a coleta de dados, não são necessárias ações adicionais da sua parte [26]. O *feedback* implícito pode ser obtido de diversas formas, por exemplo, análise do histórico de compras, do histórico de navegação, dos padrões de busca, etc [26, 34]. Tal abordagem de obtenção de informação se mostra mais valiosa quando o domínio de aplicação dificulta ou impossibilita a interação direta entre usuários e itens, como na TV Digital, em que a interação é realizada pelo controle remoto que apresenta limitações [3]. Além disso, como o usuário não é interrompido na coleta de dados [54], sua experiência ao utilizar a aplicação ou sistema é aumentada. Na TV, por exemplo, pode-se avaliar padrões de comportamento dos usuários para conseguir informações sobre suas preferências, como tempo que um usuário assistiu a um programa, quantidade de vezes que assistiu certo canal em uma semana, etc. Como o *feedback* implícito é obtido de forma indireta, deve-se ter cuidado ao analisar as informações coletadas, para que ruídos sejam evitados (comuns nessa abordagem [26]). Outra limitação da coleta implícita é a dificuldade na identificação de itens que o usuário não gostou [26].

Dependendo das características apresentadas no domínio de aplicação as duas abordagens podem ser combinadas para que uma representação mais acurada das preferências dos usuários seja obtida [29].

2.4 Mineração de Texto

A mineração de texto é uma área da computação que visa contornar o problema de sobrecarga de informação gerado, principalmente, com o advento da Web 2.0 [7], provocando a proliferação de documentos disponíveis na rede [20]. Estima-se que 85% das informações

de negócios no mundo estão em formato de texto [24], isso torna a mineração de texto uma área de pesquisa com grande valor científico e comercial [27].

A mineração de texto utiliza diversas técnicas de mineração de dados, aprendizagem de máquina, recuperação de informação, para realizar tarefas, como categorização de texto, extração de informação, extração de termos, entre outras [20]. Representa uma tarefa complexa, uma vez que lida com informações textuais desestruturadas e difusas [27].

Da mesma forma que a mineração de dados em geral, a mineração de texto visa identificar e explorar informações relevantes em meio a uma base de dados grande, todavia na mineração de texto a base de dados é formada por coleções de documentos [20]. Uma coleção de documentos consiste em um grupo de dados baseado em informação textual [20], por exemplo, um grupo de páginas web.

A mineração de texto pode ter diferentes conceitos, dependendo da perspectiva almejada. Caso o objetivo seja a extração de informação, a mineração de texto consiste em extrair fatos dos textos. Porém, se o objetivo desejado for a mineração de informação textual, a mineração de texto consiste na utilização de técnicas de aprendizagem de máquina para encontrar padrões úteis. Por fim, se a finalidade almejada for a descoberta de conhecimento, a mineração de texto consiste na extração de informações ainda não conhecidas [24].

Algoritmos de mineração de texto representam os documentos por meio de conjuntos de características (modelos de representação). Esses modelos geralmente apresentam dimensionalidade muito grande [20]. Logo, para melhor representar os documentos, algumas formas de descrição de características se destacam, são elas:

- **Representação por caracteres**: consiste na representação por letras, números, etc. Pode ser constituída por todos os caracteres dos documentos ou subconjuntos filtrados [20];

- **Representação por palavras**: consiste na representação por palavras específicas dos documentos. Podem ser excluídas palavras menos relevantes (*stop words*), caracteres especiais, etc [20];

- **Representação por termos**: consiste na representação por termos extraídos dos documentos, que podem ser individuais ou coletivos, é essencialmente composta por um subconjunto dos termos presentes no documento [20];

- **Representação por conceitos**: consiste na representação por características geradas por meio de metodologias de categorização, que podem ser provenientes de palavras do documento ou não [20].

Dentre os tipos de descrição de características, as representações por termos e conceitos apresentam maiores vantagens ao representar documentos textuais [20].

O conjunto de diferentes características extraído do conteúdo textual de todos os documentos é chamado de dicionário [24], por exemplo, dicionário de termos. Para reduzir a dimensionalidade do dicionário, algumas técnicas são utilizadas, como filtragem, lematização e *stemming*. Filtragem consiste no processo de retirar palavras menos significativas (*stop words*), como artigos, conjunções, etc [24]. Em contrapartida a lematização consiste na técnica de mapear verbos para o infinitivo e substantivos para o singular [24]. Por fim, o processo de *stemming* consiste em remover afixos da palavra, permanecendo apenas sua raiz. [24].

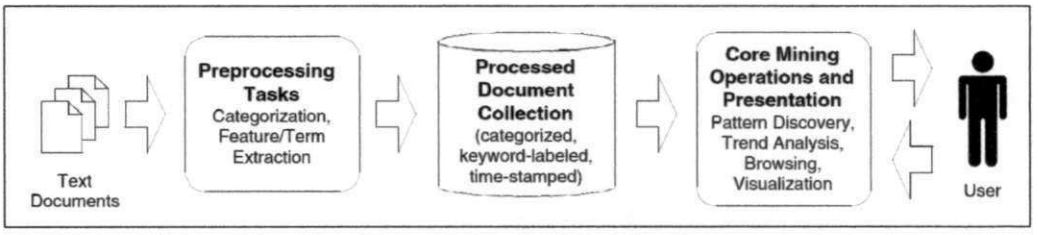


Figura 2.2: Arquitetura em alto nível da mineração de texto.

Na Figura 2.2 [20] pode ser vista uma arquitetura em alto nível da mineração de texto. As fases mais custosas de um sistema de mineração de texto são o pré-processamento (*Preprocessing Tasks*), em que ocorre a preparação dos dados textuais para que as características sejam extraídas, e operações de mineração de núcleo (*Core Mining Operations*), que incluem descoberta de padrões, algoritmos de descoberta de conhecimento, etc [20].

2.5 Categorização de Texto

Categorização de texto (classificação de texto) caracteriza-se no problema de atribuir automaticamente categorias a documentos textuais [46, 53], ou seja, dado um conjunto de cate-

gorias previamente definidas e uma coleção de documentos, identificar quais categorias são mais adequadas para cada documento [20].

A categorização de texto apresenta várias finalidades, como filtragem de *spams*, classificação de páginas web, detecção de gênero textual, etc [20]. Entre as aplicações mais comuns, destacam-se: *indexação de texto*, que consiste na ação de atribuir palavras-chave de um vocabulário controlado em documentos textuais [20]; *classificação de documentos ou filtragem de texto*, que consiste em classificar os documentos em categorias, por exemplo, classificar notícias em "esportiva", "policial", "política", etc, caso só existam duas categorias disponíveis para a classificação, a tarefa é chamada de filtragem de texto, por exemplo, filtrar e-mails indesejados (*spams*) [20]; *categorização de páginas web hierarquicamente*, que consiste em classificar páginas web de forma automática em catálogos hierárquicos postados por sites como *Yahoo*, auxiliando na busca por páginas de um determinado tema [20].

A categorização de texto pode ser realizada empregando-se duas estratégias diferentes: indutiva e transdutiva. A primeira induz um modelo de atribuição de categorias. Uma vez que a estratégia transdutiva propaga a informação dos documentos classificados para os não classificados [46].

Antes de realizar a classificação de texto, os documentos textuais são convertidos para outro tipo de representação, por exemplo, vetores de características, que tornam as informações mais adequadas para o tratamento [20], ou redes heterogêneas bipartidas, em que objetos heterogêneos representam os documentos e os termos presentes neles [46]. Uma representação comum utilizada é o conjunto de palavras ou termos contidos nos documentos, em que cada palavra possui um peso. Várias formas de atribuição de pesos para as palavras podem ser adotadas, por exemplo, binária, em que o peso de uma palavra é 0 (zero) se não pertence ao documento e 1 (um), caso contrário [20]. Outras formas de atribuição mais complexas podem ser utilizadas, como usar a frequência da palavra no documento [20], o TF-IDF (Frequência do Termo - Frequência Inversa no Documento) [20], que reflete a importância da palavra para um documento, entre outras.

Dentre as abordagens para categorização de texto destacam-se as que utilizam técnicas de aprendizagem de máquina, pois apresentam grande eficiência [48], em que o classificador é construído baseando-se em um conjunto de documentos de treinamento pré-classificados [20, 48], representando um processo de aprendizado supervisionado. Vários

classificadores podem ser adotados para a categorização de texto, como classificadores probabilísticos, regressões logísticas, classificadores baseados em árvores de decisão, classificadores baseados em regras de decisão, redes neurais, máquinas de vetor de suporte, entre outros [20].

Com o objetivo de analisar seu desempenho, os classificadores podem ser avaliados por meio de experimentos [48]. Para conduzir os experimentos é necessária uma coleção com documentos previamente classificados, a qual é dividida em conjunto de treino e teste. O conjunto de treino é utilizado para construir (treinar) o classificador e geralmente é maior que o conjunto de teste [48]. Já o conjunto de teste é usado para avaliar a eficiência do classificador e não deve ser utilizado na construção do mesmo [48]. Também pode ser utilizada validação cruzada (*n-fold cross-validation*), em que a coleção de documentos é dividida em n partes iguais, e os experimentos são realizados diversas vezes, e em cada iteração uma parte diferente é selecionada como conjunto de teste e as outras restantes como conjunto de treinamento [20].

2.6 Redes Bipartidas

Redes bipartidas são redes que apresentam dois tipos de vértices, um representando um objeto e o outro representando um grupo a que ele pertence, e cada aresta da rede corresponde a uma relação entre objetos de tipos diferentes [43]. Na Figura 2.3 [43] é apresentado um exemplo simples de uma rede bipartida.

Dada a sua capacidade de representação, as redes bipartidas são utilizadas em diversos contextos. Algumas aplicações conhecidas são:

- No problema de redes de transporte, com dois tipos de vértices, um representando as localizações e outro representando as rotas dos transportes [43];
- Para representar redes sociais, por exemplo, na relação de pessoas e eventos (dois tipos de vértices), em que as arestas da rede conectam pessoas aos eventos que frequentaram [43];
- No problema de redes de filiação, por exemplo, na relação entre atores e grupos, sendo os vértices da rede os atores e os grupos (dois vértices), em que arestas ligam atores a

grupos que eles pertencem, não havendo ligação entre vértices do mesmo tipo [43];

- Em redes de recomendação, com dois tipos de vértices representando usuários e itens, e as arestas ligando usuários a itens que eles gostaram ou compraram [43].

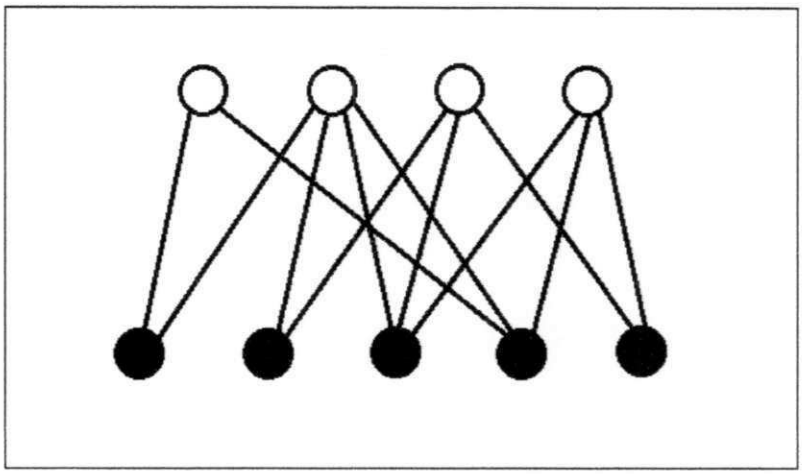


Figura 2.3: Exemplo simples de uma rede bipartida.

Além disso, redes bipartidas são uma alternativa de representação de documentos textuais no problema de categorização de texto, em que os vértices da rede correspondem a documentos e termos extraídos, e uma aresta liga um documento a um termo se esse termo estiver presente no descritivo textual do documento [46]. Essa representação proporciona vantagens em relação à representação baseada em matriz de documentos e termos [46].

Capítulo 3

Trabalhos Relacionados

Neste capítulo são apresentadas pesquisas que têm relação com a arquitetura proposta nesse trabalho, contemplando sistemas de recomendação aplicados ao domínio de TVD, assim como categorização de texto.

3.1 Sistemas de Recomendação Aplicados ao Domínio de TV Digital

Com o grande crescimento do número de programas e canais de TV, os ambientes de TV Digital sofrem com a sobrecarga de informação [5, 31]. Portanto, vários trabalhos em sistemas de recomendação estão focados nesse domínio de aplicação [2, 3, 9, 10, 12, 25, 28, 31–33, 40, 49, 55].

Bambini et al. [3] descreveram a integração de um sistema de recomendação com um provedor (Fastweb[1]) de Televisão sobre Protocolo de Internet (IPTV). O sistema de recomendação implementou as técnicas colaborativa e baseada em conteúdo para recomendar itens (programas ou vídeos sob demanda), adequando-se aos requisitos específicos do domínio de IPTV.

Na Figura 3.1 [3] é apresentada a arquitetura do sistema de recomendação proposto. Os componentes que fazem parte da arquitetura são os seguintes:

- **Coletor de Dados (*Data Collector*)**: pré-processa as informações provenientes de di-

[1] http://www.fastweb.it

ferentes fontes de dados (*Data source*) para gerar a entrada dos algoritmos de recomendação (*Recommender algorithms*);

- **Repositório (*Repository*)**: armazena as informações dos itens e dos usuários em duas matrizes diferentes, matriz de conteúdo dos itens (ICM) e matriz de avaliações dos usuários (URM).

- **Algoritmos de Recomendação (*Recommender algorithms*)**: implementa algoritmos de sistemas de recomendação, um baseado em conteúdo e dois colaborativos. Os algoritmos processam as informações provenientes das matrizes de conteúdo dos itens e de avaliações dos usuários seguindo uma abordagem baseada em modelo. Após a geração desse modelo as recomendações são computadas e as chamadas provenientes da interface de serviços web (*Recommender web services*) são respondidas.

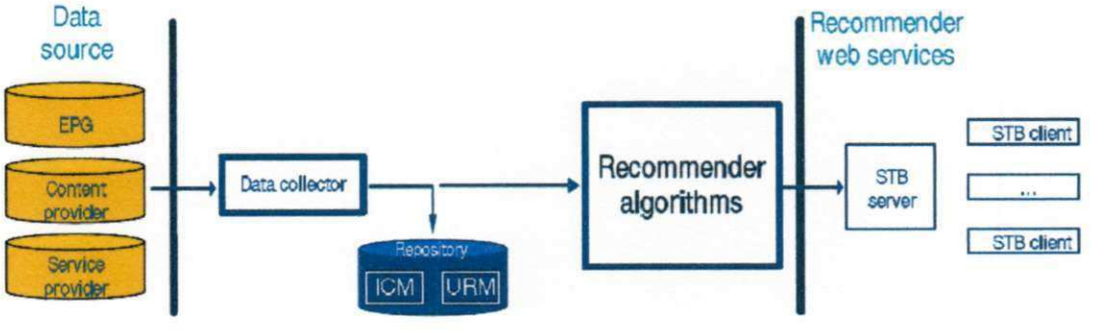


Figura 3.1: Arquitetura do sistema de recomendação para IPTV.

Chang et al. [10] propuseram um *framework* de recomendação de programas de TV para *Smart* TV, abordando questões como acurácia, diversidade, novidade, etc. O *framework* é constituído de três componentes, são eles:

- **Módulo de análise de conteúdo do programa de TV**: coleta informações acerca dos programas, como informações básicas (título, canal, etc), informações de audiência e informações sobre aceitação dos programas pelos usuários;

- **Módulo de análise do perfil do usuário**: coleta e extrai informações acerca dos usuários, como informações demográficas, informações sobre histórico de visualizações, informações sobre preferência e informações sobre redes sociais;

- **Módulo de aprendizado das preferências do usuário**: realiza tarefas de aprendizado relacionadas aos interesses dos usuários, baseadas nos históricos de visualização (experiência passada), similaridade implícita (redes implícitas), relações em redes sociais (redes explícitas).

Martínez et al. [40] apresentaram um sistema de recomendação de programas de TV personalizado. Visando contornar diversas limitações comuns das abordagens de recomendação quando utilizadas separadamente (superespecialização, *cold start*, etc), os autores propuseram a adoção de uma abordagem híbrida com a junção de técnicas baseadas em conteúdo e baseadas em filtragem colaborativa. Além disso, os autores exploraram a tecnologia SVD (do inglês *Singular Value Decomposition*) para reduzir a dimensionalidade dos dados e evitar o problema da esparsidade.

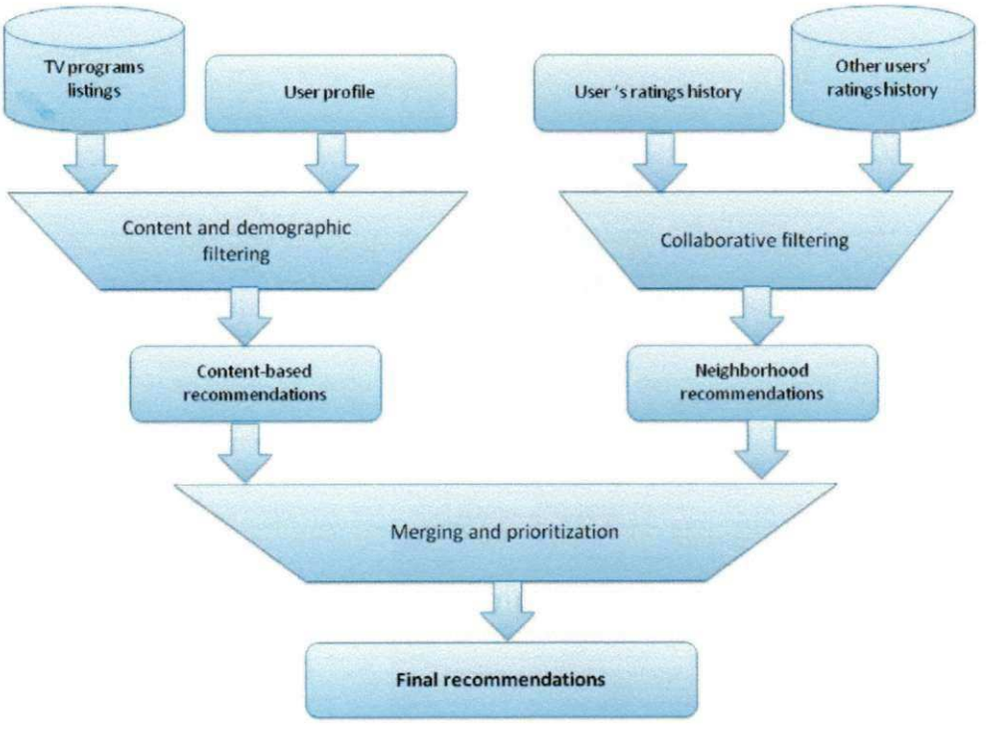


Figura 3.2: Visão geral do *framework* de recomendação.

Uma visão geral do *framework* de recomendação pode ser visto na Figura 3.2 [40], mostrando os dois métodos de recomendação utilizados, baseado em conteúdo (*Content-based recomendations*) e baseado em filtragem colaborativa (*Neighborhood recommendations*), e

uma junção dos resultados de ambos os métodos (*Merging and priorization*) para gerar a recomendação final dos programas (*Final recommendations*).

Ardissono et al. [2] apresentaram um guia de programação pessoal, que utiliza informações sobre programas de TV e as preferências dos usuários para gerar recomendações.

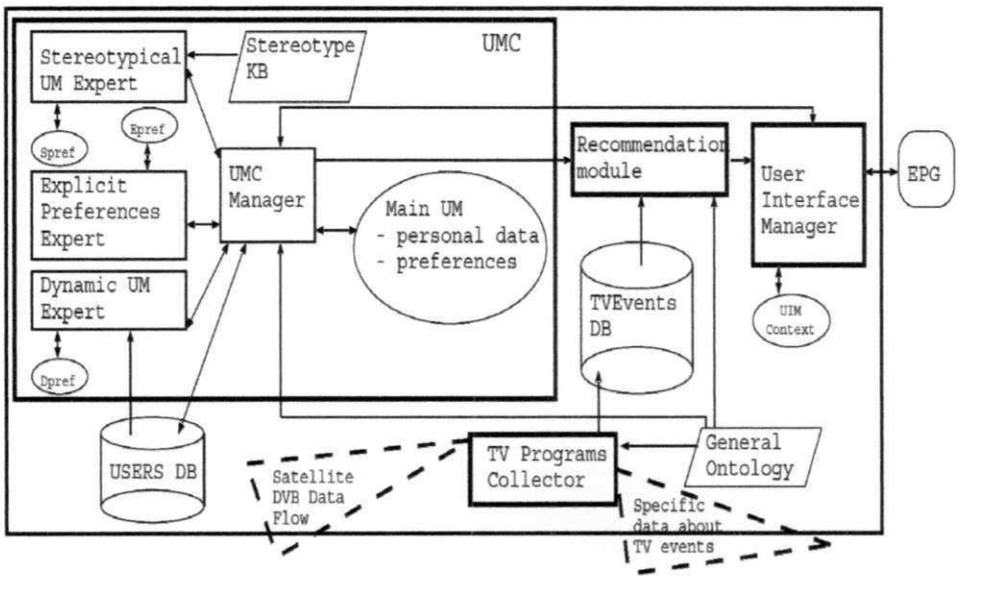


Figura 3.3: Arquitetura do guia de programação pessoal.

Na Figura 3.3 [2] é exposta uma visão geral da arquitetura do guia de programação pessoal proposto no trabalho. O componente modelador do usuário (UMC) gerencia informações provenientes de três módulos, são eles:

- **Especialista em preferências explícitas (*Explicit Preferences Expert*)**: armazena informações explicitamente atribuídas pelos usuários;
- **Especialista no estereótipo do modelo do usuário (*Stereotypical UM Expert*)**: armazena predições sobre as preferências dos usuários;
- **Especialista dinâmico do modelo do usuário (*Dynamic UM Expert*)**: armazena estimativas sobre as preferências dos usuários.

As preferências dos usuários são representadas pela união das predições dos três especialistas, formando os modelos dos usuários (*Main UM*), baseando-se nesses modelos os programas são sugeridos pelo módulo de recomendação (*Recommendation Module*).

Krauss et al. [32] propuseram um sistema (*TV Predictor*) que integra mecanismos de recomendação e *Smart* TVs, visando gerar guias de programação personalizados, que consistem em canais pessoais para cada usuário do sistema. Uma das características do sistema proposto é a troca de canais de forma automática, que permite que o usuário assista a programação recomendada sem a necessidade de ações adicionais.

As recomendações dos programas são geradas com base no comportamento de visualização dos usuários e em avaliações explicitamente atribuídas por eles. O sistema de recomendação proposto combina algoritmos de recomendação em uma abordagem híbrida.

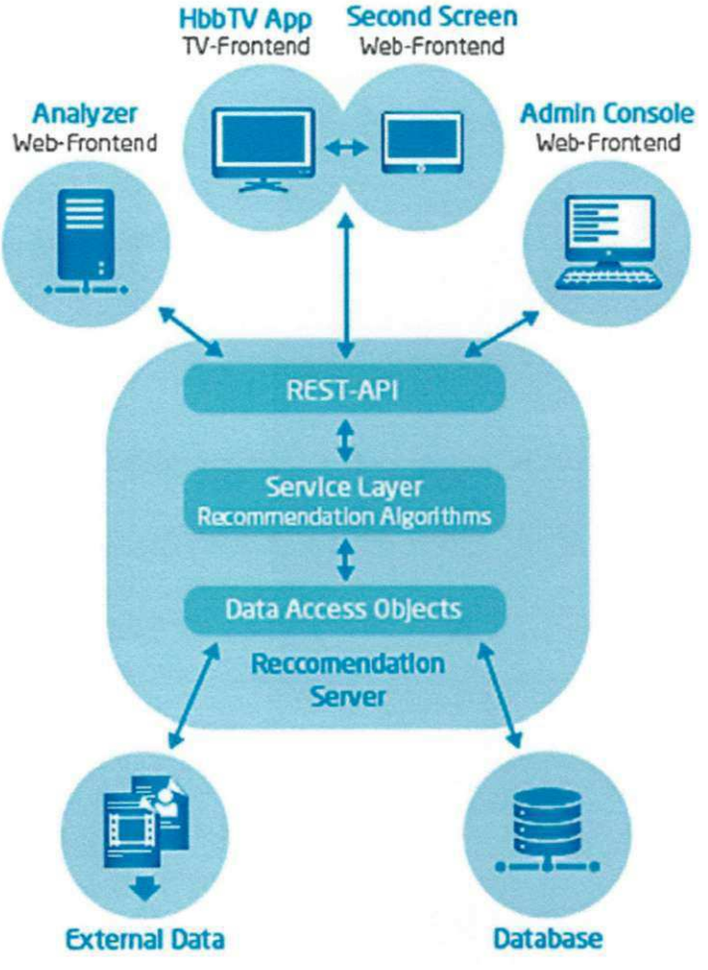


Figura 3.4: Arquitetura do sistema *TV Predictor*.

A arquitetura do sistema é baseada em uma estrutura cliente-servidor (Figura 3.4 [32]). O lado cliente consiste em quatro módulos *front-end* para coletar informações dos usuários a serem utilizadas no lado servidor pelos mecanismos de recomendação, que calculam as

recomendações dos programas baseando-se nas requisições dos usuários.

Engelbert et al. [18] estenderam um registrador de vídeo pessoal com um sistema de recomendação genérico baseado em um classificador Bayesiano e o adaptaram para o domínio de TV. O sistema gera recomendação de programas por meio da análise das preferências dos usuários.

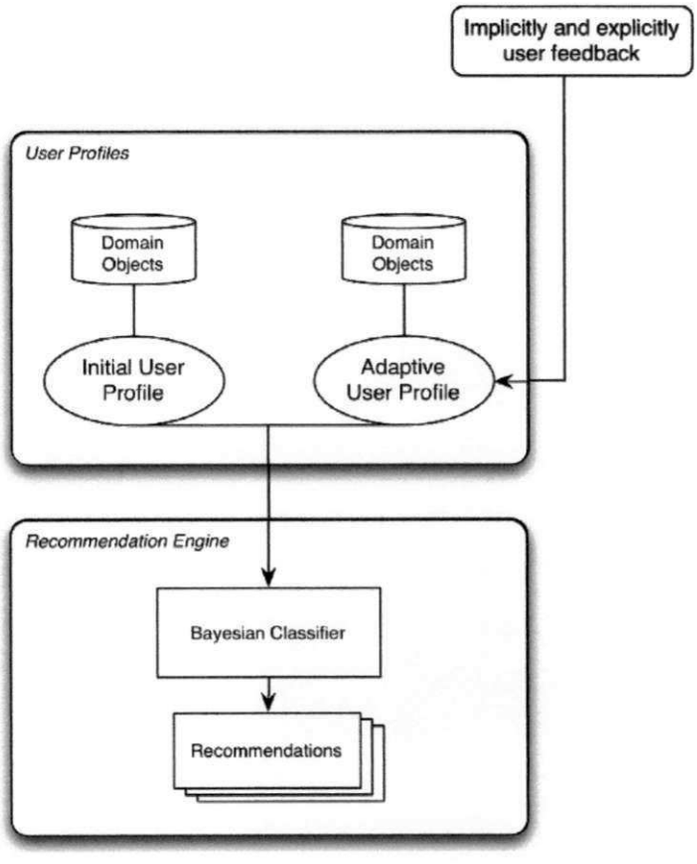


Figura 3.5: Visão geral da arquitetura da adaptação do registrador de vídeo pessoal para aplicações da área de TV.

O processo de recomendação é baseado nos perfis dos usuários, que são obtidos de duas formas, definidos pelos usuários de forma explícita no início do sistema (*initial user profile* (Figura 3.5 [18])) e de forma implícita por meio da análise dos comportamentos dos usuários (*adaptive user profile* (Figura 3.5 [18])). Os itens são representados por atributos do seu conteúdo presentes no EPG (Domain Objects (Figura 3.5 [18])) e podem ser classificados como itens que os usuários gostam ou não gostam. Assim, durante o processo de recomendação um novo item é avaliado comparando seus atributos com os atributos dos itens avaliados anteri-

ormente e uma probabilidade calculada por um classificador Bayesiano (*Bayesian Classifier* (Figura 3.5 [18])) para identificar se o item pertence ao grupo de itens que o usuário gosta ou não gosta.

Diferente dos trabalhos descritos anteriormente, que visam apenas recomendar itens específicos (programas de TV), neste trabalho o objetivo é gerar uma arquitetura para a recomendação de termos extraídos dos descritivos textuais dos programas e classificados baseando-se em suas categorias especificadas no EPG, que possa ser usada por diferentes aplicações de *Smart* TV. Assim, além da recomendação de programas (componente da arquitetura proposta) que é processada a partir da análise dos históricos dos usuários, também é gerada a recomendação de termos. Portanto, a arquitetura proposta pode ser reutilizada para facilitar a geração de recomendações para aplicações de contextos diferentes, sendo esta a principal característica que diferencia a arquitetura proposta com relação as demais apresentadas.

3.2 Categorização de Texto

Como grande parte dos conteúdos disponíveis estão apresentados em forma textual [46], alguns trabalhos direcionam o seu foco para categorização de texto [46, 48, 50].

Rossi et al. [46] propuseram um algoritmo para categorização de documentos textuais inspirado na estrutura de uma rede bipartida heterogênea (Figura 3.6 [46]) para induzir um modelo de categorização. A abordagem baseada em redes bipartidas heterogêneas proposta pelos autores visa contornar problemas comumente encontrados na representação de coleções de documentos textuais, como esparsidade e grande dimensionalidade. A rede é composta por dois objetos de tipos diferentes, documentos e termos extraídos de seus descritivos textuais, em que no conjunto de treinamento alguns documentos apresentavam-se previamente classificados e a indução consistiu em atribuir pesos para termos com relação às classes conhecidas.

Neste trabalho foi feita uma adaptação da abordagem proposta pelos autores para realizar a fase de classificação dos termos, considerando cada programa como sendo um documento e as categorias dos mesmos representando as classes da rede. Como cada programa presente no EPG tem suas categorias predefinidas, a adaptação objetivou apenas identificar qual a

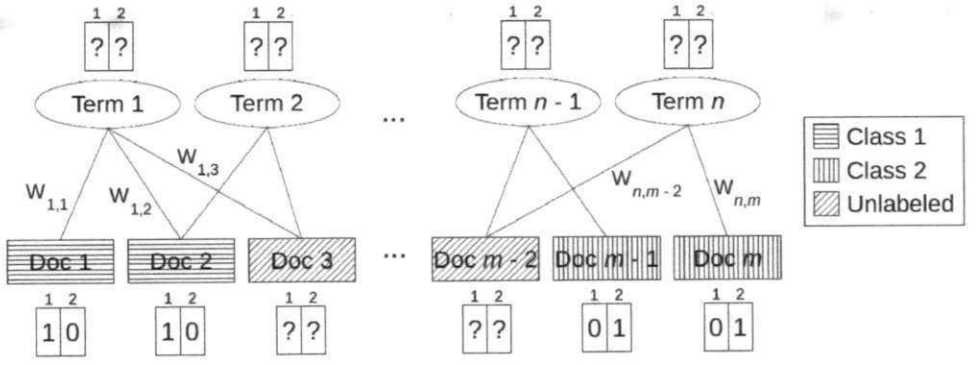


Figura 3.6: Rede bipartida heterogênea usada na categorização de documentos textuais.

relação entre cada termo extraído das descrições dos programas com as categorias presentes no EPG.

Capítulo 4

Solução Proposta

Neste capítulo a arquitetura proposta para extração, classificação e recomendação de termos é apresentada. As etapas principais da solução proposta são: Extração, Classificação e Recomendação de Termos.

4.1 Visão Geral da Arquitetura

Grande parte das arquiteturas de recomendação propostas na literatura trabalha com a recomendação de itens [2, 3, 10, 32]. Porém, no contexto de TV Digital/*Smart* TV, com a agregação de aplicativos com conteúdos diversos, uma abordagem que garanta interoperabilidade das recomendações se mostra necessária. Assim, a arquitetura proposta neste trabalho visa gerar recomendações de termos observando as especificações do domínio de TV Digital para maximizar a experiência do usuário ao assistir TV. A arquitetura de recomendação proposta baseia-se nas seguintes especificações:

- **Diversas aplicações:** vários tipos de aplicações de *Smart* TV podem ser adicionadas à TV Digital, apresentando itens diversos, como livros, filmes, notícias, etc. Assim, a recomendação gerada deve ser intermediária, ou seja, deve ser possível gerar uma recomendação final a partir dela, garantindo interoperabilidade;
- **Interação feita pelo controle remoto:** o usuário interage com a TV por meio do controle remoto. Logo, para maximizar a experiência do usuário ao assistir TV a obtenção de informações deve ser feita por meio de *feedback* implícito;

- ***Feedback* implícito:** o domínio de TV Digital apresenta uma especificação relevante em relação aos demais domínios que utilizam sistemas de recomendação (*e-commerce*, etc), pois existe um período pelo qual o usuário pode consumir um item. Por exemplo, se um programa é transmitido uma vez por semana, um determinado usuário só pode assistir a ele no exato dia e momento da sua transmissão. Assim, observando essa especificação do domínio a forma de obtenção de *feedback* implícito utilizada identifica quantas vezes um programa é transmitido por semana e quantas vezes cada usuário assistiu a ele, criando assim os perfis dos usuários utilizados para gerar as recomendações;

- **EPG com categorias predefinidas:** uma característica importante do domínio de TV Digital é que as informações acerca dos programas disponíveis são fornecidas pelo EPG. Além de título, descrição, entre outras informações, geralmente, cada programa recebe duas categorias predefinidas, por exemplo, "série" e "ação". Essa propriedade não é encontrada em outros domínios, em que um item pode não ter classificação ou as categorias podem não ser bem definidas. Portanto, com as descrições dos programas e suas respectivas categorias é possível identificar a relação entre termos extraídos dos descritivos textuais e categorias.

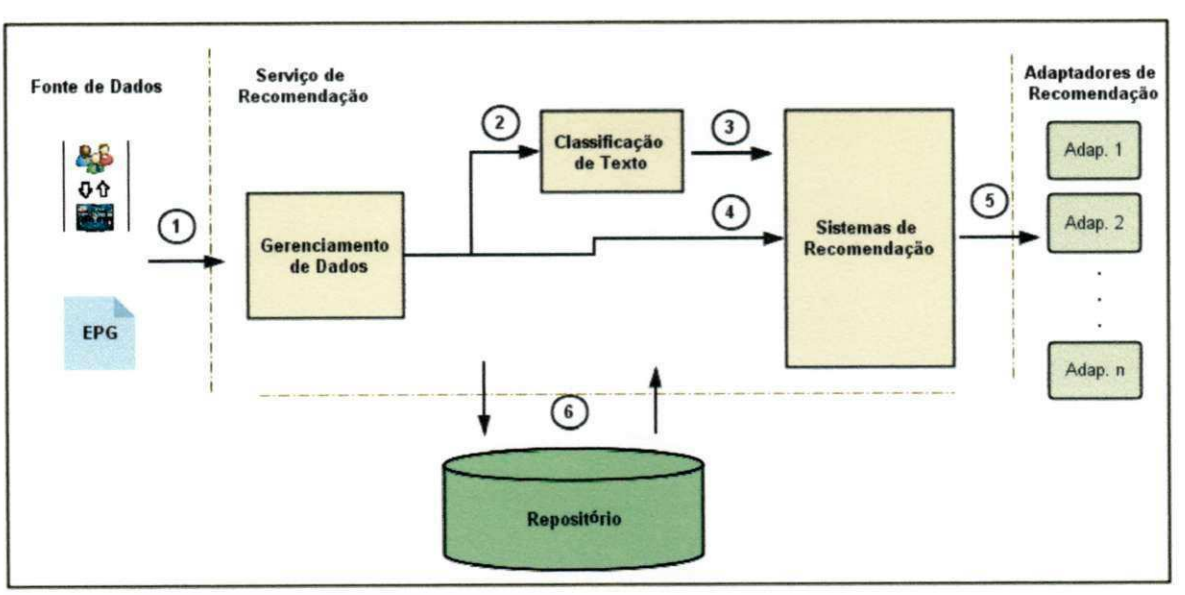


Figura 4.1: Visão geral da arquitetura proposta.

Na Figura 4.1 pode ser vista uma visão geral da arquitetura proposta neste trabalho, exibindo os seguintes componentes:

- **Gerenciamento de Dados**: coleta informações das fontes de dados (1) provenientes do EPG (informações dos programas) e da interação entre usuários e TV (informações dos usuários), tendo como objetivo criar os perfis dos usuários e itens;
- **Classificação de Texto**: realiza a mineração dos descritivos textuais dos programas de TV coletados do EPG (2) e a classificação dos termos extraídos com base nas categorias dos programas;
- **Sistemas de Recomendação**: analisam os perfis dos usuários e itens (4) para gerar recomendações personalizadas de termos que foram previamente classificados (3);
- **Adaptador de Recomendação**: utiliza os termos recomendados (5) para gerar a recomendação final. Para cada aplicação que utilizar a recomendação de termos deve-se criar um adaptador correspondente;
- **Repositório**: armazena e recupera (6) informações dos perfis dos usuários e programas e das fases de extração, classificação e recomendação de termos.

4.2 Arquitetura Proposta

O objetivo principal da arquitetura (Figura 4.2) é garantir a usabilidade das recomendações por diferentes aplicações de *Smart* TVs. Assim, a partir de informações coletadas implicitamente os perfis dos usuários e itens são gerados e suas informações passadas para as fases de classificação de texto e geração das recomendações.

Na classificação de texto os termos extraídos dos descritivos textuais são classificados baseando-se nas categorias dos programas, possibilitando a identificação de termos de "ação", "comédia ", "suspense", entre outras categorias. A geração da recomendação de termos é processada a partir da recomendação de programas e da classificação prévia dos termos.

O esboço da arquitetura proposta neste trabalho pode ser visto na Figura 4.2. A seguir os componentes da arquitetura são detalhados.

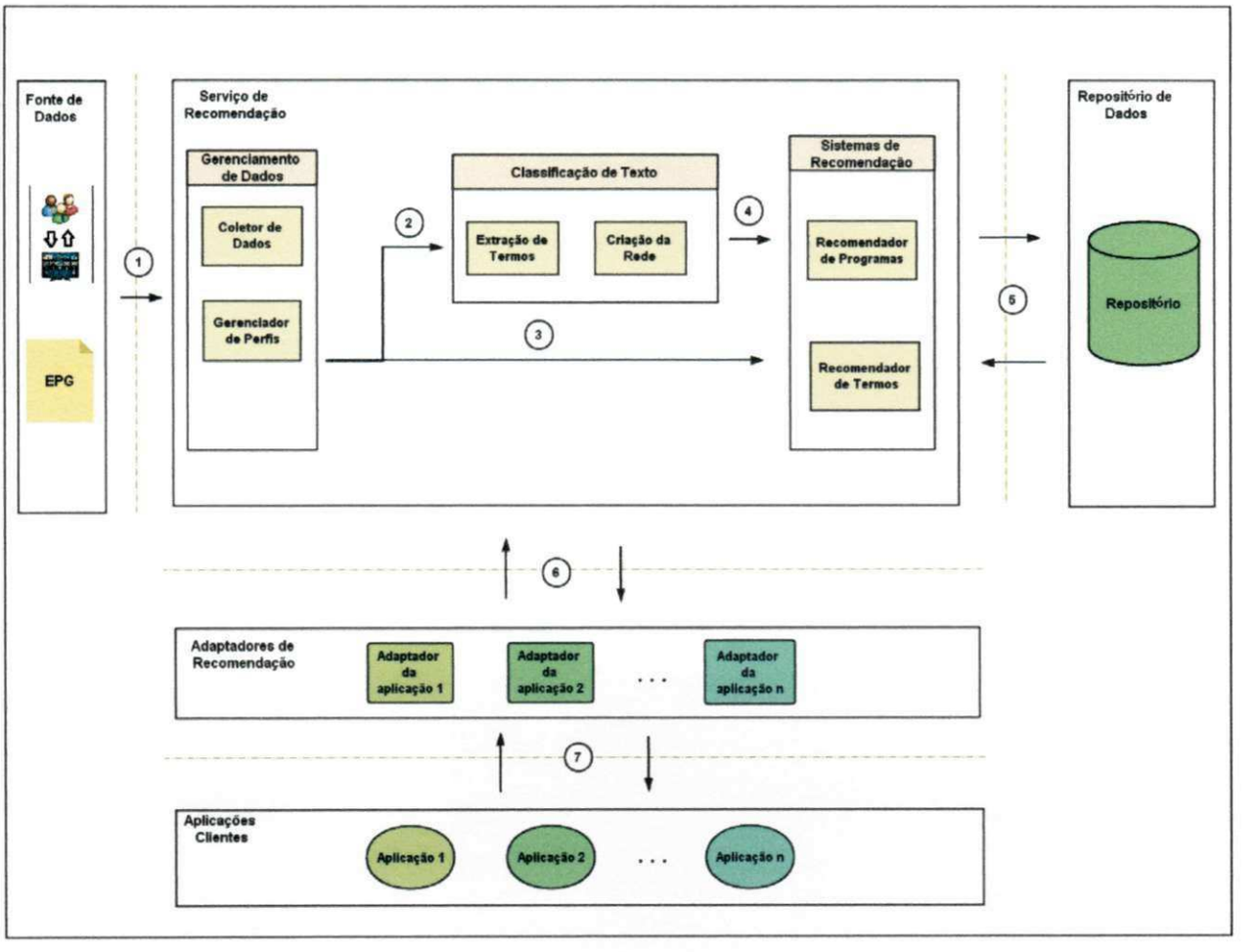


Figura 4.2: Visão detalhada da arquitetura proposta.

4.2.1 Coleta e Gerenciamento de Dados

A coleta das informações dos programas de TV e dos usuários do sistema é realizada pelo componente Coletor de Dados (Figura 4.2), que é um elemento geralmente presente em arquiteturas de recomendação [3]. Na arquitetura proposta as informações dos programas são extraídas do EPG, que fornece dados como título dos programas, descrição, categorias, horário de início e fim, classificação etária, etc. Já as informações dos usuários são coletadas implicitamente quando estes interagem com a TV.

O Gerenciador de Perfis (Figura 4.2) é responsável pela gestão das informações coletadas das fontes de dados (1) e tem como principal objetivo criar os perfis dos usuários e itens que são armazenados no repositório de dados (5) e posteriormente utilizados nas fases de

classificação e recomendação.

Os perfis dos programas são formados por suas características textuais extraídas do EPG. Ao passo que os perfis dos usuários são criados a partir da análise de seus históricos de visualização coletados. Para cada programa presente no histórico de um usuário é obtida implicitamente uma avaliação. Para calcular o *feedback* implícito foi analisada a quantidade de vezes que um usuário u assistiu a um programa p e com que frequência esse programa é ofertado semanalmente, aplicando-se a Equação 4.1.

$$\overline{r}_{up} = \frac{v_{up}}{f_p} \times 5, \tag{4.1}$$

onde v_{up} representa a quantidade de vezes que o usuário u assistiu ao programa p e f_p representa o número de vezes que o programa p é transmitido semanalmente. A avaliação obtida é um valor arredondado entre 0 e 5. A Tabela 4.1 descreve um exemplo da obtenção do *feedback* implícito.

Tabela 4.1: Exemplo da obtenção de *feedback* implícito utilizada.

ID do usu.	**ID do prog.**	**Freq. sem. prog.**	**Freq. visua. usu.**	***Feedback* imp.**
1	1	3	3	5
1	2	1	1	5
2	1	3	1	2
⋮	...	...	...	...
2	3	7	5	4

As informações dos perfis dos usuários (3) são utilizadas pelos Sistemas de Recomendação (Figura 4.2) na geração das recomendações e as informações dos perfis dos itens (2) são utilizadas na Classificação de Texto (Figura 4.2).

4.2.2 Extração e Classificação de Termos

A extração de termos provenientes dos descritivos textuais de programas de TV ocorre no componente Extração de Termos (Figura 4.2), já a classificação dos termos com relação as categorias desses programas ocorre no componente Criação da Rede (Figura 4.2).

A fase de extração de termos se dá pela mineração dos descritivos textuais dos programas de TV para extrair termos representativos. Para realizar essa tarefa, primeiramente são descartadas palavras menos significativas, como preposições, artigos, etc, denominadas *stop words* [19]. Logo em seguida, é realizado o processo de *stemming*, que consiste no procedimento de redução das palavras aos seus radicais [19]. Ao final desta fase, cada programa apresenta um vetor de termos, em que cada posição do vetor corresponde à frequência de um termo no descritivo do programa.

No componente Criação da Rede ocorre a fase de categorização dos termos extraídos anteriormente com relação às categorias do EPG. Como cada programa do EPG apresenta geralmente duas categorias predefinidas, é possível identificar a relação entre termos e categorias por meio de suas coocorrências. Portanto, ao final dessa fase cada termo extraído apresenta um peso relacionado com às categorias do EPG.

Para realizar a classificação dos termos, uma rede heterogênea bipartida é construída, que consiste em uma rede de objetos de tipos diferentes $G = \{V, E, W\}$, em que V é um conjunto de objetos, E é o conjunto de conexões entre os objetos - não há ligações entre objetos do mesmo tipo - e W é o conjunto de pesos das conexões [46].

Na rede bipartida utilizada neste trabalho é feita uma adaptação da abordagem proposta por Rossi.et.al [46], apresentando dois tipos de objetos: termos e programas. O conjunto de pesos dos termos é dado por $W = \{w_1^T \dots w_\alpha^T\}^T$, em que α é o número de termos extraídos dos descritivos textuais dos programas e w_{ij} é o peso do termo i para a classe j. As classes (categorias dos programas de TV) são representadas pelo vetor $c = \{c_1, \dots, c_{|C|}\}$. Os termos extraídos dos descritivos textuais dos programas são representados pelo vetor $f = \{f_1, \dots, f_\alpha\}$. Cada programa possui um vetor de pesos com as classes, que é representado pela matriz $Y = \{y_1^T, \dots, y_\theta^T\}^T$, em que y_{kj} recebe o valor 1 caso o programa k tenha determinada categoria j, ou 0, caso contrário. O peso da relação entre termos e programas é dado por $D = \{d_1^T, \dots, d_\theta^T\}^T$, em que θ representa o número de programas disponibilizados, e cada posição d_{ki} corresponde à frequência do termo f_i no descritivo do programa d_k.

O objetivo da fase de classificação é construir a matriz W. Para realizar essa tarefa o algoritmo IMBHN (do inglês *Inductive Model Based on Bipartite Heterogeneous Network*) [46] é usado, possibilitando induzir as influências de cada termo para as categorias dos programas. O algoritmo IMBHN realiza o processo de indução por meio da minimização da função

de custo apresentada na Equação 4.2 [46]:

$$\begin{aligned} Q(W) &= \frac{1}{2}\left(\sum_{j=1}^{w}\sum_{k=1}^{\theta}(class(\sum_{i=1}^{\alpha} d_{ki}w_{ij}) - y_{kj})^2\right) \\ &= \frac{1}{2}\left(\sum_{j=1}^{w}\sum_{k=1}^{\theta} error_{kj}^2\right), \end{aligned} \tag{4.2}$$

onde,

$$class(\sum_{i=1}^{\alpha} d_{ki}w_{ij}) = \begin{cases} 1 & c_j = \underset{c_{j^*} \in c}{\text{argmax}}(\sum_{i=1}^{\alpha} d_{ki}w_{ij^*}) \\ 0 & \text{caso contrário} \end{cases} \tag{4.3}$$

O algoritmo visa minimizar o erro quadrado entre os valores preditos e reais das classes dos programas. Por meio do gradiente descendente (*Least-Mean-Square* [46]) os valores da matriz W são ajustados até que um erro mínimo seja alcançado ou um número máximo de iterações seja atingido (condições de parada do algoritmo). Ao final do processo, a matriz W é preenchida com os pesos entre termos e classes.

4.2.3 Recomendação de Termos

Antes de gerar a recomendação de termos, os perfis dos usuários e itens são analisados e recomendações de programas processadas com a finalidade de identificar itens adequados aos interesses dos usuários

A recomendação de programas consiste em avaliar e sugerir programas do interesse dos usuários, essa tarefa é realizada pelo Recomendador de Programas (Figura 4.2). A Tabela 4.2 apresenta uma exemplificação do problema de recomendação de programas, por exemplo, para o *Usuário 5* as avaliações dos programas 1 e 2 não são conhecidas, então uma abordagem de recomendação pode ser usada para encontrar avaliações para estes dois itens e recomendá-los.

Tabela 4.2: Caracterização da recomendação de programas.

	Programa 1	**Programa 2**	**Programa 3**	**Programa 4**
Usuário 1	1	?	3	5
Usuário 2	?	1	?	5
Usuário 3	3	?	3	4
Usuário 4	4	3	5	?
Usuário 5	?	?	3	2

Neste trabalho as listas de programas recomendados para os usuários são geradas utilizando duas abordagens de recomendação, filtragem colaborativa e baseada em conteúdo. A abordagem colaborativa utilizada é a Fatoração de Matriz, em que a aprendizagem é realizada por gradiente descendente estocástico [30]. A FM representa o estado da arte na recomendação de itens [30], pois combina boa escalabilidade e acurácia das predições. Além da recomendação colaborativa, os históricos dos usuários são analisados e os programas com as melhores avaliações incluídos nas listas recomendadas pela FM.

Após a geração das recomendações de programas, é processada a recomendação de termos pelo Recomendador de Termos (Figura 4.2). Como cada programa recomendado possui uma lista de termos extraídos (Extração de Termos (Figura 4.2)) e classificados (Criação da Rede (Figura 4.2)), a recomendação de termos é dada pela Equação 4.4.

$$\overline{r}_{tu} = \frac{1}{|P_u|} \sum_{p=1}^{|P_u|} f_{tp} \times r_{pu} \tag{4.4}$$

onde $|P_u|$ representa o número de programas recomendados ao usuário u que contem o termo t, f_{tp} a frequência de ocorrências do termo t na descrição de um programa p e r_{pu} a avaliação recomendada pelo sistema de recomendação para o programa p para o usuário u. Assim, termos que ocorrem com maior frequência em programas que foram mais bem recomendados para um determinado usuário tendem a receber um peso maior.

4.2.4 Adaptadores de Recomendação

Os Adaptadores de Recomendação (Figura 4.2) são responsáveis pela geração da recomendação final (7) que é utilizada pelas Aplicações Clientes (Figura 4.2). Assim, para cada aplicação deve existir um adaptador correspondente, que utiliza o Serviço de Recomendação (Figura 4.2) de termos.

O adaptador deve conhecer as categorias relacionadas com a aplicação cliente e sua lista de usuários (7), ao informar esses dados ao serviço de recomendação (6) listas de termos recomendados para os usuários são retornadas (6), por exemplo, lista de termos recomendados de ação, de comédia, etc.

Uma técnica possível para gerar a recomendação final é calcular a similaridade entre as listas de termos recomendados e as lista de termos extraídos dos descritivos textuais dos itens

recomendáveis (filmes, livros, notícias, etc). Dentre as abordagens conhecidas de cálculo de similaridade destaca-se a similaridade do cosseno (Equação 4.5) [14, 47], em que dois itens x e y são considerados vetores de dimensão m (termos em comum) e a similaridade entre eles é medida pelo cálculo do cosseno do ângulo entre esses dois vetores.

$$sim(x, y) = cos(x, y) = \frac{x \cdot y}{\| x \| \| y \|} \tag{4.5}$$

4.3 Considerações Finais do Capítulo

Este capítulo apresentou uma arquitetura de recomendação para TV Digital baseada na extração e classificação de termos dos descritivos de programas. A arquitetura baseia-se em especificações do domínio de TV Digital/*Smart* TV. Como a recomendação gerada pela solução proposta é intermediária, uma recomendação final deve ser processada a partir dos termos recomendados para os usuários. Portanto, com as etapas de extração, classificação e recomendação de termos processadas, é possível identificar termos recomendados que têm maior relação com determinadas categorias do EPG, tornando possível o uso dessa abordagem por aplicações com conteúdos diferentes, pois basta identificar quais categorias apresentam relação com a aplicação que pretende utilizar a recomendação de termos. Logo, a solução simplifica a geração da recomendação final, uma vez que não é necessário criar todos os componentes do sistema de recomendação, mas apenas um adaptador para cada aplicação que usa o serviço de recomendação de termos.

Capítulo 5

Validação do Trabalho

Neste capítulo são apresentadas as tarefas realizadas na validação da solução proposta, especificando como a base de dados utilizada na validação foi gerada, quais ferramentas foram utilizadas no trabalho, assim como especificações acerta do protótipo desenvolvido e dos resultados e conclusões obtidos acerca das recomendações processadas.

5.1 Geração da Base de Dados

Para realizar a validação da solução proposta foi necessária a obtenção de uma base de dados com informações acerca de usuários de TV e seus históricos de visualização. Para tanto, um *survey* foi conduzido em forma de formulário (Apêndice A), em que os participantes informaram os seguintes dados:

- Identificação do usuário;
- Quantidade de vezes que assistiram semanalmente aos programas previamente informados no questionário, sendo a quantidade informada um valor inteiro de 0 a 7;
- Outros programas que não foram citados no formulário, mas que foram assistidos pelo participante, colocando título do programa e quantidade de vezes que assistiram a eles semanalmente.

Os participantes foram, majoritariamente, alunos do curso de Ciência da Computação da Universidade Federal de Campina Grande (UFCG) e ao final da coleta de dados a base apresentou as seguintes características:

- 63 participantes de ambos os sexos e com idades entre 20 e 30 anos;
- 29 categorias de programas (séries, variedades, comédia, jornalismo, esportivo, novela, infantil, programa, *reality show*, diversos, informativo, filme, policial, entrevista, documentário, drama, futebol, entretenimento, musical, espetáculo, show, animação, ação, desenho, cinema, suspense, aventura, culinária e educativo);
- 112 programas, cada um com duas categorias associadas;

5.2 Ferramentas

Dentre as ferramentas utilizadas durante o desenvolvimento da solução proposta e do processo de validação, destacam-se:

MyMediaLite[1]. Biblioteca multipropósito de algoritmos para sistemas de recomendação, que contempla abordagens de predição de notas e predição de itens. Possui código aberto e é distribuída segundo os termos da GNU *General Public License* (GPL) [21]. A biblioteca foi utilizada para gerar a recomendação colaborativa de programas.

Microsoft SQL Server[2]. Sistema gerenciador de banco de dados relacional utilizado para armazenar e acessar as informações dos itens e dos usuários.

C#[3]. Linguagem de programação orientada a objetos, que combina rapidez e potência. Tem suas raízes nas linguagens de programação C e por isso assemelha-se com as linguagens C e C++. A linguagem C# foi utilizada tanto na implementação da solução proposta quanto na implementação do protótipo.

PTStemmer[4]. Ferramenta que realiza o processo de redução de palavras da língua portuguesa em suas raízes morfológicas. Foi utilizada no processo de extração de termos

Microsoft Visual Studio [5]. Conjunto de ferramentas e serviços para o desenvolvimento de aplicações, que fornece um ambiente de desenvolvimento para linguagens como .NET, C++, etc. Foi utilizado na implementação da solução proposta e da validação.

[1] http://www.mymedialite.net/

[2] https://www.microsoft.com/sqlserver/pt/br/default.aspx

[3] http://msdn.microsoft.com/en-us/library/ms228593.aspx

[4] https://code.google.com/p/ptstemmer/

[5] http://www.visualstudio.com/

5.3 Protótipo

O objetivo principal do protótipo desenvolvido é a validação da interoperabilidade das recomendações, ou seja, recomendar itens diferentes a partir da recomendação de termos. Logo, o protótipo desenvolvido consiste na utilização da recomendação de termos para gerar a recomendação final. Dois tipos de itens foram recomendados, filmes e livros.

A arquitetura de recomendação proposta deve ser integrada em uma arquitetura de TVD. Porém, para questões de validação o processo se deu em um ambiente *desktop*.

A criação do adaptador de recomendação para a sugestão de filmes a partir da recomendação de termos ocorreu da seguinte forma:

1. Categorias relacionadas com a aplicação de filmes foram especificadas (ação, aventura, comédia, drama, policial e suspense);

2. A lista dos usuários da aplicação de filmes foi determinada;

3. Com as categorias e usuários definidos foram obtidas recomendações de termos relacionadas com cada categoria, por exemplo, lista recomendada de termos de ação, de comédia, etc;

4. Após obter os termos recomendados, a recomendação final foi gerada. Para recomendar filmes para um dado usuário o processo se deu da seguinte forma: primeiro, a sinopse dos filmes foi minerada e uma lista de termos extraída para cada filme da base, representando seu vetor de características; segundo, calculou-se a similaridade (similaridade do cosseno) entre as listas de termos recomendadas para cada usuário e a lista de termos extraída das sinopses, recomendando os filmes que obtiveram maior similaridade.

A recomendação de livros se deu da mesma forma que a recomendação de filmes. Logo, outras aplicações podem utilizar a recomendação de termos desde que seus itens possuam um descritivo textual e tenham suas categorias especificadas. Em uma aplicação para recomendar notícias, por exemplo, as categorias relacionadas podem ser "jornalismo" e "informativo", realizando o mesmo processo indicado para a recomendação de filmes.

5.4 Avaliação da Solução

5.4.1 Descrição dos Dados

Os dados usados na avaliação da solução representam valores das precisões [23] das sugestões de filmes e livros para os 63 usuários da base, que foram recomendados a partir de termos obtidos com a abordagem de extração, classificação e recomendação de termos (proposta no trabalho) e a partir de termos extraídos dos históricos dos usuários (sem personalização). Para calcular as precisões os seguintes passos foram seguidos:

1. Foram obtidos dois tipos de conjuntos de termos para cada usuário, um usando a abordagem de extração, classificação e recomendação (personalizado por categoria, por exemplo, termos de ação, de aventura, de comédia, etc) e outro extraído dos históricos dos usuários (um único conjunto de termos para cada usuário sem classificação por categoria);

2. A partir dos conjuntos de termos anteriores foram retornados itens finais (filmes e livros) e ordenados baseando-se na similaridade entre o conjunto de termos extraídos de suas descrições e os conjuntos citados no passo 1, ou seja, quanto maior a similaridade entre os conjuntos de termos, melhor a ordem de um item para um usuário;

3. Para filmes, a precisão foi calculada baseando-se nos 5 primeiros itens retornados ($P@5$), ao passo que para livros baseou-se nos 4 primeiros ($P@4$). Os valores indicados representam a média de itens por categoria;

4. Com a abordagem de classificação de termos os usuários possuem conjuntos de termos diferentes para cada categoria. Assim, a precisão foi calculada avaliando-se quantos itens que possuem uma determinada categoria foram retornados dado o conjunto de termos recomendados dessa mesma categoria. Por exemplo, dos 5 filmes retornados para um usuário u, quantos são de ação dado o conjunto de termos recomendados de ação desse usuário, ou seja, caso os 5 sejam de ação a precisão é 1 (100% de precisão);

5. Para o conjunto de termos extraído do histórico de cada usuário a precisão foi calculada da mesma forma do passo 4. Porém, como os conjuntos não são personalizados

para cada categoria, sempre são retornados os mesmos itens finais independente da categoria desejada.

Os dados usados nesta avaliação podem ser vistos no Apêndice B, representando as precisões das recomendações para os 63 usuários da base, para cada uma das seis categorias avaliadas e para dois tipos de itens finais (filmes e livros). Para cada usuário as precisões foram calculadas de duas formas, a partir de termos personalizados e a partir de termos extraídos dos seus históricos. Portanto, os dados são pareados.

5.4.2 Objetivos e Hipóteses

O objetivo da validação é comparar as precisões das recomendações finais obtidas a partir da abordagem de extração, classificação e recomendação de termos e utilizando apenas termos extraídos dos históricos dos usuários (sem personalização). Essa comparação visa identificar qual das abordagens apresenta melhores resultados, assim compararam-se as precisões obtidas para cada categoria e para dois tipos de itens, filmes e livros.

Portanto, a principal questão da avaliação da solução é:

P1. A utilização da abordagem de extração, classificação e recomendação de termos melhora a precisão das recomendações finais comparando-se com a utilização de termos extraídos dos históricos dos usuários?

Logo, algumas hipóteses nulas foram consideradas. Para a sugestão de filmes:

$H_0 - 1$: não há diferença entre as precisões da sugestão de filmes para a categoria ação.

$H_0 - 2$: não há diferença entre as precisões da sugestão de filmes para a categoria aventura.

$H_0 - 3$: não há diferença entre as precisões da sugestão de filmes para a categoria comédia.

$H_0 - 4$: não há diferença entre as precisões da sugestão de filmes para a categoria drama.

$H_0 - 5$: não há diferença entre as precisões da sugestão de filmes para a categoria policial.

$H_0 - 6$: não há diferença entre as precisões da sugestão de filmes para a categoria suspense.

Para a sugestão de livros:

$H_0 - 7$: não há diferença entre as precisões da sugestão de livros para a categoria ação.

H_0-8: não há diferença entre as precisões da sugestão de livros para a categoria aventura.

H_0-9: não há diferença entre as precisões da sugestão de livros para a categoria comédia.

H_0-10: não há diferença entre as precisões da sugestão de livros para a categoria drama.

H_0-11: não há diferença entre as precisões da sugestão de livros para a categoria policial.

H_0-12: não há diferença entre as precisões da sugestão de livros para a categoria suspense.

Caso essas hipóteses nulas sejam refutadas tem-se um indício de que existem diferenças significativas entre as precisões obtidas com os dois tipos de conjuntos de termos para a sugestão dos itens finais (filmes e livros), sendo possível identificar se a utilização da abordagem de extração, classificação e recomendação de termos obtém melhores resultados.

5.4.3 Passo a Passo da Análise

Como na análise têm-se comparações dois a dois, alguns testes estatísticos são conhecidos para essa finalidade, como o teste t (paramétrico) e o teste de *Wilcoxon* (não-paramétrico) [8]. Para que o teste t seja utilizado, alguns requisitos devem ser atendidos, como: normalidade e homoscedasticidade. Portanto, o primeiro passo da análise é conhecer o perfil dos dados coletados para que o teste estatístico seja definido. Logo em seguida, o teste escolhido é aplicado com a finalidade de comparar as amostras e refutar ou aceitar as hipóteses levantadas na Subseção 5.4.2.

5.4.4 Conhecendo o Perfil dos Dados Coletados

O primeiro passo da análise é avaliar a normalidade dos dados. Assim, para cada grupo de dados obtido foram realizados testes de normalidade de *Shapiro-Wilk* [44].

Na Tabela 5.1 são apresentados os resultados dos testes de *Shapiro-Wilk* para os dados obtidos na recomendação de livros, em que observando os p-valores encontrados conclui-se com 95% de confiança que nenhuma das amostras avaliadas provém de uma população com distribuição normal, pois os p-valores obtidos nos testes são inferiores a 0,05 (nível de significância).

Ao passo que na Tabela 5.2 são apresentados os resultados dos testes de *Shapiro-Wilk*

Tabela 5.1: Resultados dos testes de normalidade de *Shapiro-Wilk* para os conjuntos de dados obtidos na recomendação de livros, em que cada valor da tabela representa o p-valor do teste.

	Livros	
	Com clas./rec. de termos	**Sem clas./rec. de termos**
Acão	4.063e-10	2.835e-05
Aventura	6.522e-12	1.007e-07
Comédia	2.304e-13	2.811e-09
Drama	2.315e-12	8.313e-07
Policial	7.761e-12	1.515e-08
Suspense	2.824e-09	1.353e-06

para os dados obtidos na recomendação de filmes, em que observando-se os p-valores encontrados também é possível concluir com 95% de confiança que nenhuma das amostras analisadas provém de uma população com distribuição normal.

Tabela 5.2: Resultados dos testes de normalidade de *Shapiro-Wilk* para os conjuntos de dados obtidos na recomendação de filmes, em que cada valor da tabela representa o p-valor do teste.

	Filmes	
	Com clas./rec. de termos	**Sem clas./rec. de termos**
Acão	1.533e-08	1.364e-05
Aventura	3.293e-06	8.086e-06
Comédia	5.657e-11	1.524e-06
Drama	2.178e-08	1.626e-07
Policial	5.467e-12	1.63e-07
Suspense	9.366e-16	6.576e-06

Logo, percebeu-se que nenhum dos grupos de dados segue uma distribuição normal. Assim, o teste t não deve ser usado, tendo como alternativa não-paramétrica o teste de *Wilcoxon*, que foi utilizado na análise dos dados.

5.4.5 Análise dos Dados

Nesta parte do estudo foi utilizado o teste de *Wilcoxon signed-rank*, que consiste em uma versão não paramétrica do teste t para amostras emparelhadas. Esse teste é usado quando a população estudada não apresenta uma distribuição normal [8], característica encontrada nos dados analisados neste trabalho.

Na Tabela 5.3 podem ser vistos os resultados dos testes de *Wilcoxon signed-rank* para a comparação das duas abordagens, com nível de significância de 5%, ou seja, α igual a 0,05. Portanto, para cada teste gerado a hipótese nula é rejeitada caso o p-valor identificado seja menor que 0,05, e aceita, caso contrário. Logo, como as hipóteses nulas $H_0 - 1$, $H_0 - 5$, $H_0 - 6$, $H_0 - 8$, $H_0 - 9$, $H_0 - 10$, $H_0 - 11$ e $H_0 - 12$ foram rejeitas, pode-se afirmar com 95% de confiança nesses casos que as duas abordagens comparadas apresentaram resultados diferentes, ou seja, é possível identificar qual delas obteve melhor precisão na recomendação final. Além disso, como o p-valor identificado no teste da hipótese nula $H_0 - 7$ foi de 0.08228 que é próximo de 0,05 (α), este caso também foi investigado.

Tabela 5.3: Resultados dos testes de *Wilcoxon signed-rank* para a comparação das duas abordagens, em que para cada teste a hipótese nula é que os resultados são iguais e a hipótese alternativa indica que os resultados são diferentes.

Hipótese	**Resultado (p-valor)**	**Decisão**
$H_0 - 1$	$3.407e - 06$	Rejeita
$H_0 - 2$	0.8961	Aceita
$H_0 - 3$	0.4553	Aceita
$H_0 - 4$	0.3631	Aceita
$H_0 - 5$	$8.399e - 11$	Rejeita
$H_0 - 6$	$1.558e - 05$	Rejeita
$H_0 - 7$	0.08228	Aceita
$H_0 - 8$	0.002068	Rejeita
$H_0 - 9$	0.02493	Rejeita
$H_0 - 10$	$3.48e - 08$	Rejeita
$H_0 - 11$	$3.036e - 09$	Rejeita
$H_0 - 12$	$1.765e - 10$	Rejeita

Na Tabela 5.4 são apresentados os resultados dos testes de *Wilcoxon signed-rank* para identificar qual das abordagens apresenta melhores resultados. Cada teste foi realizado com nível de significância de 5%, ou seja, α igual a 0,05. Portanto, para as hipóteses nulas rejeitadas da Tabela 5.3 foram levantadas as seguintes hipóteses alternativas:

$H_A - 1$: a abordagem proposta no trabalho obtém melhor precisão para a sugestão de filmes da categoria ação.

$H_A - 5$: a abordagem proposta no trabalho obtém melhor precisão para a sugestão de filmes da categoria policial.

$H_A - 6$: a abordagem proposta no trabalho obtém melhor precisão para a sugestão de filmes da categoria suspense.

$H_A - 7$[6]: a abordagem proposta no trabalho obtém melhor precisão para a sugestão de livros da categoria ação.

$H_A - 8$: a abordagem proposta no trabalho obtém melhor precisão para a sugestão de livros da categoria aventura.

$H_A - 9$: a abordagem proposta no trabalho obtém melhor precisão para a sugestão de livros da categoria comédia.

$H_{A_1} - 10$: a abordagem proposta no trabalho obtém melhor precisão para a sugestão de livros da categoria drama.

$H_{A_2} - 10$: a utilização dos termos extraídos dos históricos dos usuários obtém melhor precisão para a sugestão de livros da categoria drama.

$H_A - 11$: a abordagem proposta no trabalho obtém melhor precisão para a sugestão de livros da categoria policial.

$H_A - 12$: a abordagem proposta no trabalho obtém melhor precisão para a sugestão de livros da categoria suspense.

Analisando os resultados dos teste da Tabela 5.4 percebe-se que as hipóteses $H_A - 1$, $H_A - 5$, $H_A - 6$, $H_A - 7$, $H_A - 8$, $H_A - 9$, $H_A - 11$, $H_A - 12$ foram aceitas com 95% de confiança, pois os p-valores dos testes são menores que 0,05 (nível de significância), mostrando que nesses casos a abordagem proposta no trabalho apresentou melhores resultados. Já pela análise dos testes das hipóteses $H_{A_1} - 10$, $H_{A_2} - 10$ conclui-se com 95% de confiança que

[6]Como o p-valor identificado no teste da hipótese nula $H_0 - 7$ foi de 0.08228 que é próximo de 0,05 (α), este caso também foi investigado por meio da hipótese alternativa $H_A - 7$.

Tabela 5.4: Resultados dos testes de *Wilcoxon signed-rank* para a comparação das duas abordagens, em que as hipóteses alternativas indicam que a abordagem proposta no trabalho apresenta resultados melhores (exceto a hipótese $H_{A_2} - 10$).

Hipótese	**Resultado (p-valor)**	**Decisão**
$H_A - 1$	$1.704e - 06$	Aceita $H_A - 1$
$H_A - 5$	$4.2e - 11$	Aceita $H_A - 5$
$H_A - 6$	$7.79e - 06$	Aceita $H_A - 6$
$H_A - 7$	0.04114	Aceita $H_A - 7$
$H_A - 8$	0.001034	Aceita $H_A - 8$
$H_A - 9$	0.01246	Aceita $H_A - 9$
$H_{A_1} - 10$	1	Rejeita
$H_{A_2} - 10$	$1.74e - 08$	Aceita $H_{A_2} - 10$
$H_A - 11$	$1.518e - 09$	Aceita $H_A - 11$
$H_A - 12$	$8.823e - 11$	Aceita $H_A - 12$

a utilização de termos extraídos dos históricos dos usuários apresenta melhores resultados para esta ocorrência.

5.4.6 Conclusões da Análise

Após a análise dos resultados dos testes estatísticos observou-se que para recomendação de filmes a utilização da abordagem proposta no trabalho apresentou melhores resultados para 3 categorias estudadas (ação, policial e suspense) e resultados semelhantes para as outras 3 restantes (aventura, comédia e drama). Assim, conclui-se que o uso da abordagem de extração, classificação e recomendação de termos melhora a precisão da recomendação final de filmes. Além disso, para a recomendação de livros observou-se que a abordagem proposta obteve melhores resultados para 5 categorias estudadas (ação, aventura, comédia, policial, suspense) e resultado pior em apenas uma das ocorrências (drama), mostrando que para a recomendação de livros a abordagem proposta também melhora a recomendação final. Portanto, conclui-se que a utilização da abordagem proposta neste trabalho supera em questões de precisão a utilização de termos extraídos dos históricos dos usuários na geração da recomendação final de itens.

Capítulo 6

Considerações Finais

Neste capítulo são apresentadas as conclusões e contribuições do trabalho, como também sugestões de trabalhos futuros.

6.1 Conclusões e Contribuições do Trabalho

Neste trabalho, foi apresentada uma arquitetura para extração, classificação e recomendação de termos aplicada ao domínio de TV Digital e baseada na mineração dos descritivos textuais de itens (programas de TV). Como a pesquisa tem foco na área de sistemas de recomendação aplicados em TV Digital, e por esse ser um domínio de pesquisa que carece de estudos aprofundados, o trabalho proposto apresenta contribuição relevante. Além disso, a abordagem proposta para extração dos termos é importante para contornar os problemas provenientes das restrições de interação entre TV e usuários. Outra contribuição significativa é a abordagem de classificação e recomendação de termos, que possibilita que aplicações com diferentes conteúdos utilizem as recomendações geradas, o que não é possível em abordagens que recomendam itens. Por fim, tendo em vista a dificuldade encontrada neste trabalho para obter-se dados referentes as informações do consumo de TV por usuários, a base de dados gerada representa uma contribuição na área, pois possibilita que outras pesquisas sejam realizadas a partir dos dados coletados.

O protótipo desenvolvido mostrou a viabilidade da utilização da abordagem de recomendação proposta, revelando que é possível recomendar itens a partir dos termos recomendados para os usuários. Assim, para que uma determinada aplicação utilize as recomendações, é

necessário apenas conhecer as categorias provenientes do EPG e identificar quais delas tem maior relação com a aplicação, considerando apenas os termos recomendados com maior relação com as categorias indicadas.

6.2 Trabalhos Futuros

Como trabalho futuro pode ser realizado um estudo sobre melhores formas de extração de termos. Alguns trabalhos têm foco na extração de atributos dos produtos [22, 52], as abordagens propostas nesses estudos podem ser avaliadas e sua integração na proposta atual considerada com o objetivo de obter uma melhor representação dos programas de TV e consequentemente, recomendações mais significativas.

Como a solução proposta neste trabalho é baseada na extração de conteúdos do EPG, que em alguns casos pode conter informações reduzidas [42], um trabalho futuro possível é a investigação de formas de obtenção de dados em bases diferentes para enriquecer a informação do EPG e melhorar a representação textual dos programas de TV, por exemplo, utilizar informações da Wikipédia[1] [42].

[1]https://pt.wikipedia.org

Bibliografia

[1] Gediminas Adomavicius and Alexander Tuzhilin. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Trans. on Knowl. and Data Eng.*, 17(6):734–749, June 2005.

[2] L. Ardissono, C. Gena, P. Torasso, F. Bellifemine, A. Chiarotto, A. Difino, and B. Negro. Personalized recommendation of tv programs. In *In LNAI n. 2829. AI*IA 2003: Advances in Artificial Intelligence*, pages 474–486. Springer Verlag, 2003.

[3] Riccardo Bambini, Paolo Cremonesi, and Roberto Turrin. A recommender system for an iptv service provider: a real large-scale production environment. In Francesco Ricci, Lior Rokach, Bracha Shapira, and Paul B. Kantor, editors, *Recommender Systems Handbook*, pages 299–331. Springer, 2011.

[4] Ana Belén Barragáns-Martínez, Enrique Costa-Montenegro, Juan C. Burguillo, Marta Rey-López, Fernando A. Mikic-Fonte, and Ana Peleteiro. A hybrid content-based and item-based collaborative filtering approach to recommend tv programs enhanced with singular value decomposition. *Inf. Sci.*, 180(22):4290–4311, November 2010.

[5] Patrick Baudisch and Lars Brueckner. Tv scout: Lowering the entry barrier to personalized tv program recommendation. In Matthias Hemmje, Claudia Niederée, and Thomas Risse, editors, *From Integrated Publication and Information Systems to Information and Knowledge Environments*, volume 3379 of *Lecture Notes in Computer Science*, pages 299–309. Springer Berlin Heidelberg, 2005.

[6] J. Bennett and S. Lanning. The netflix prize. In *Proceedings of the KDD Cup Workshop 2007*, pages 3–6, New York, August 2007. ACM.

[7] Michael Berry. *Survey of Text Mining : Clustering, Classification, and Retrieval.* Springer, September 2003.

[8] Sarah Boslaugh and Paul A. Watters. Nonparametric statistics. In *Statistics in a Nutshell: A Desktop Quick Reference (In a Nutshell (O'Reilly))*, pages 207–223. O'Reilly Media, 2008.

[9] Jui-Hung Chang, Chin-Feng Lai, Ming-Shi Wang, and Tin-Yu Wu. A cloud-based intelligent tv program recommendation system. *Computers e Electrical Engineering*, 39(7):2379 – 2399, 2013.

[10] Na Chang, Mhd Irvan, and Takao Terano. A tv program recommender framework. *Procedia Computer Science*, 22(0):561 – 570, 2013. 17th International Conference in Knowledge Based and Intelligent Information and Engineering Systems - KES2013.

[11] Yoon Ho Cho, Jae Kyeong Kim, and Soung Hie Kim. A personalized recommender system based on web usage mining and decision tree induction. *Expert Systems with Applications*, 23(3):329 – 342, 2002.

[12] Paulo Muniz de Ávila and Sérgio Donizetti Zorzo. A personalized tv guide system compliant with ginga. In *Proceedings of the XV Brazilian Symposium on Multimedia and the Web*, WebMedia '09, pages 5:1–5:8, New York, NY, USA, 2009. ACM.

[13] Maunendra Sankar Desarkar, Sudeshna Sarkar, and Pabitra Mitra. Aggregating preference graphs for collaborative rating prediction. In *Proceedings of the fourth ACM conference on Recommender systems*, RecSys '10, pages 21–28, New York, NY, USA, 2010. ACM.

[14] Mukund Deshpande and George Karypis. Item-based top-n recommendation algorithms. *ACM Trans. Inf. Syst.*, 22(1):143–177, January 2004.

[15] Christian Desrosiers and George Karypis. A comprehensive survey of neighborhood-based recommendation methods. In Francesco Ricci, Lior Rokach, Bracha Shapira, and Paul B. Kantor, editors, *Recommender Systems Handbook*, pages 107–144. Springer US, 2011.

[16] Olhar Digital. Olhar digital tem aplicativos para tvs inteligentes. Disponível em: http://olhardigital.uol.com.br/video/olhar-digital-tem-aplicativos-para-tvs-inteligentes/34164, acessado em: 14 de janeiro de 2014.

[16] Olhar Digital. Olhar digital tem aplicativos para tvs inteligentes. Disponível em: http://olhardigital.uol.com.br/video/olhar-digital-tem-aplicativos-para-tvs-inteligentes/34164, acessado em: 14 de janeiro de 2014.

[17] Revista Eletrônica. Metadados de programação de tv. Disponível em: http://www.revistaeletronica.com.br/wordpress/, acessado em: 14 de janeiro de 2014.

[18] B. Engelbert, M.B. Blanken, R. Kruthoff-Bruwer, and K. Morisse. A user supporting personal video recorder by implementing a generic bayesian classifier based recommendation system. In *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2011 IEEE International Conference on*, pages 567–571, March 2011.

[19] Antonio Fariña, Nieves R. Brisaboa, Gonzalo Navarro, Francisco Claude, Ángeles S. Places, and Eduardo Rodríguez. Word-based self-indexes for natural language text. *ACM Trans. Inf. Syst.*, 30(1):1:1–1:34, March 2012.

[20] Ronen Feldman and James Sanger. *The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data.* Cambridge University Press, Cambridge, MA, USA, December 2006.

[21] Zeno Gantner, Steffen Rendle, Christoph Freudenthaler, and Lars Schmidt-Thieme. Mymedialite: A free recommender system library. In *Proceedings of the Fifth ACM Conference on Recommender Systems*, RecSys '11, pages 305–308, New York, NY, USA, 2011. ACM.

[22] Rayid Ghani, Katharina Probst, Yan Liu, Marko Krema, and Andrew Fano. Text mining for product attribute extraction. *SIGKDD Explor. Newsl.*, 8(1):41–48, June 2006.

[23] Jonathan L. Herlocker, Joseph A. Konstan, Loren G. Terveen, and John T. Riedl. Evaluating collaborative filtering recommender systems. *ACM Trans. Inf. Syst.*, 22(1):5–53, January 2004.

[24] Andreas Hotho, Andreas Nurnberger, and Gerhard Paab. A brief survey of text mining. *LDV Forum - GLDV Journal for Computational Linguistics and Language Technology*, 2005.

[25] ShangH. Hsu, Ming-Hui Wen, Hsin-Chieh Lin, Chun-Chia Lee, and Chia-Hoang Lee. Aimed- a personalized tv recommendation system. In Pablo Cesar, Konstantinos Chorianopoulos, and JensF. Jensen, editors, *Interactive TV: a Shared Experience*, volume 4471 of *Lecture Notes in Computer Science*, pages 166–174. Springer Berlin Heidelberg, 2007.

[26] Yifan Hu, Yehuda Koren, and Chris Volinsky. Collaborative filtering for implicit feedback datasets. In *Proceedings of the 2008 Eighth IEEE International Conference on Data Mining*, ICDM '08, pages 263–272, Washington, DC, USA, 2008. IEEE Computer Society.

[27] Ah hwee Tan. Text mining: The state of the art and the challenges. In *In Proceedings of the PAKDD 1999 Workshop on Knowledge Disocovery from Advanced Databases*, pages 65–70, 1999.

[28] Tamas Jambor. Intelligent media indexing and television recommender systems. In *Proceedings of the Third BCS-IRSG Conference on Future Directions in Information Access*, FDIA'09, pages 50–55, Swinton, UK, UK, 2009. British Computer Society.

[29] Diane Kelly and Jaime Teevan. Implicit feedback for inferring user preference: A bibliography. *SIGIR Forum*, 37(2):18–28, September 2003.

[30] Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, August 2009.

[31] Harald Kosch and G. Holbling. Application of recommendation methods for tv programs. In *Multimedia and Expo (ICME), 2011 IEEE International Conference on*, pages 1–4, July 2011.

[32] Christopher Krauss, Lars George, and Stefan Arbanowski. Tv predictor: Personalized program recommendations to be displayed on smarttvs. In *Proceedings of the 2Nd*

International Workshop on Big Data, Streams and Heterogeneous Source Mining: Algorithms, Systems, Programming Models and Applications, BigMine '13, pages 63–70, New York, NY, USA, 2013. ACM.

[33] Hyeong-Joon Kwon and Kwang-Seok Hong. Personalized smart tv program recommender based on collaborative filtering and a novel similarity method. *Consumer Electronics, IEEE Transactions on*, 57(3):1416–1423, August 2011.

[34] Tong Queue Lee, Young Park, and Yong-Tae Park. A time-based approach to effective recommender systems using implicit feedback. *Expert Syst. Appl.*, 34(4):3055–3062, May 2008.

[35] Wei-Po Lee, Che Kaoli, and Jhih-Yuan Huang. A smart tv system with body-gesture control, tag-based rating and context-aware recommendation. *Know.-Based Syst.*, 56:167–178, January 2014.

[36] Greg Linden, Brent Smith, and Jeremy York. Amazon.com recommendations: Item-to-item collaborative filtering. *IEEE Internet Computing*, 7(1):76–80, January 2003.

[37] Jiahui Liu, Peter Dolan, and Elin Rønby Pedersen. Personalized news recommendation based on click behavior. In *Proceedings of the 15th International Conference on Intelligent User Interfaces*, IUI '10, pages 31–40, New York, NY, USA, 2010. ACM.

[38] Pasquale Lops, Marco Gemmis, and Giovanni Semeraro. Content-based recommender systems: State of the art and trends. In Francesco Ricci, Lior Rokach, Bracha Shapira, and Paul B. Kantor, editors, *Recommender Systems Handbook*, pages 73–105. Springer US, 2011.

[39] Cheng-Che Lu and Vincent S. Tseng. A novel method for personalized music recommendation. *Expert Systems with Applications*, 36(6):10035 – 10044, 2009.

[40] A. Martinez, J.J. Pazos Arias, A.F. Vilas, J.G. Duque, and M.L. Nores. What's on tv tonight? an efficient and effective personalized recommender system of tv programs. *Consumer Electronics, IEEE Transactions on*, 55(1):286–294, 2009.

[41] Dorin Militaru and Costin Zaharia. A survey of collaborative filtering-based systems for online recommendation. In *Proceedings of the 12th International Conference on Electronic Commerce: Roadmap for the Future of Electronic Business*, ICEC '10, pages 43–47, New York, NY, USA, 2010. ACM.

[42] Cataldo Musto, Fedelucio Narducci, Pasquale Lops, Giovanni Semeraro, Marco Gemmis, Mauro Barbieri, Jan Korst, Verus Pronk, and Ramon Clout. Enhanced semantic tv-show representation for personalized electronic program guides. In Judith Masthoff, Bamshad Mobasher, MichelC. Desmarais, and Roger Nkambou, editors, *User Modeling, Adaptation, and Personalization*, volume 7379 of *Lecture Notes in Computer Science*, pages 188–199. Springer Berlin Heidelberg, 2012.

[43] Mark Newman. *Networks: An Introduction*. Oxford University Press, Inc., New York, NY, USA, 2010.

[44] Nornadiah Mohd Razali and Yap Bee Wah. Power comparisons of shapiro-wilk, kolmogorov-smirnov, lilliefors and anderson-darling tests. *Journal of Statistical Modeling and Analytics*, 2(1):21–33, 2011.

[45] Francesco Ricci, Lior Rokach, and Bracha Shapira. Introduction to recommender systems handbook. In Francesco Ricci, Lior Rokach, Bracha Shapira, and Paul B. Kantor, editors, *Recommender Systems Handbook*, pages 1–35. Springer, 2011.

[46] R.G. Rossi, T. de Paulo Faleiros, A. de Andrade Lopes, and S.O. Rezende. Inductive model generation for text categorization using a bipartite heterogeneous network. In *Data Mining (ICDM), 2012 IEEE 12th International Conference on*, pages 1086–1091, 2012.

[47] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th International Conference on World Wide Web*, WWW '01, pages 285–295, New York, NY, USA, 2001. ACM.

[48] Fabrizio Sebastiani. Machine learning in automated text categorization. *ACM Comput. Surv.*, 34(1):1–47, March 2002.

[49] Xiaowei Shi and Jin Hua. An adaptive preference learning method for future personalized tv. In *Integration of Knowledge Intensive Multi-Agent Systems, 2005. International Conference on*, pages 260 – 264, 18-21, 2005.

[50] Yang Song, Lu Zhang, and C. Lee Giles. Automatic tag recommendation algorithms for social recommender systems. *ACM Trans. Web*, 5(1):4:1–4:31, February 2011.

[51] Xiaoyuan Su and Taghi M. Khoshgoftaar. A survey of collaborative filtering techniques. *Adv. in Artif. Intell.*, 2009:4:2–4:2, January 2009.

[52] Tak-Lam Wong, Wai Lam, and Tik-Shun Wong. An unsupervised framework for extracting and normalizing product attributes from multiple web sites. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '08, pages 35–42, New York, NY, USA, 2008. ACM.

[53] Yiming Yang. An evaluation of statistical approaches to text categorization. *Inf. Retr.*, 1(1-2):69–90, May 1999.

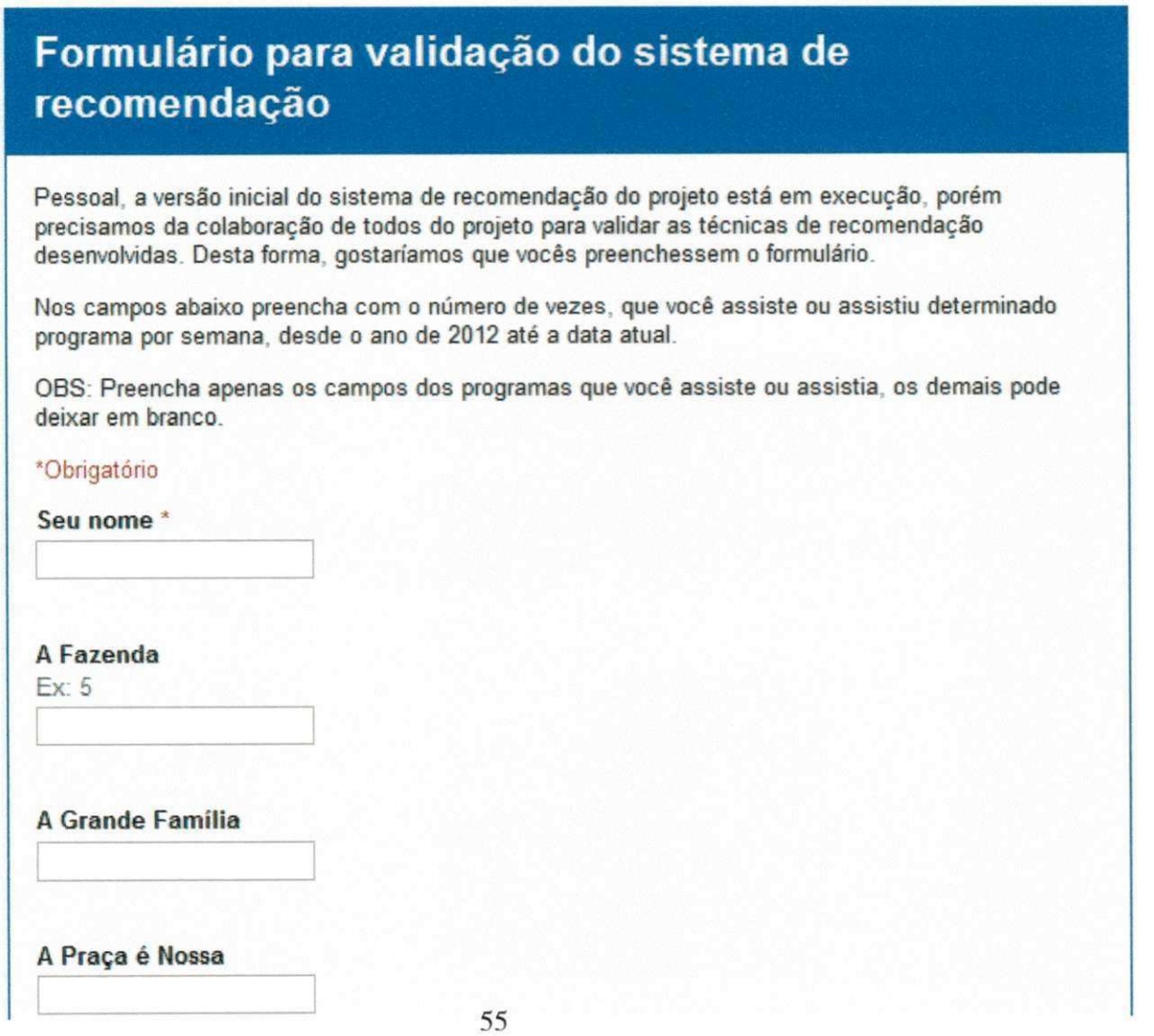
[54] Bangzuo Zhang, Yu Guan, Haichao Sun, Qingchao Liu, and Jun Kong. Survey of user behaviors as implicit feedback. In *Computer, Mechatronics, Control and Electronic Engineering (CMCE), 2010 International Conference on*, volume 6, pages 345–348, 2010.

[55] Ya Zhang, Weiyuan Chen, and Zibin Yin. Collaborative filtering with social regularization for tv program recommendation. *Knowledge-Based Systems*, 54(0):310 – 317, 2013.

Apêndice A

Formulário de Validação

O formulário usado para a coleta dos dados utilizados na validação do trabalho pode ser visto na Figura A.1.



Formulário para validação do sistema de recomendação

Pessoal, a versão inicial do sistema de recomendação do projeto está em execução, porém precisamos da colaboração de todos do projeto para validar as técnicas de recomendação desenvolvidas. Desta forma, gostaríamos que vocês preenchessem o formulário.

Nos campos abaixo preencha com o número de vezes, que você assiste ou assistiu determinado programa por semana, desde o ano de 2012 até a data atual.

OBS: Preencha apenas os campos dos programas que você assiste ou assistia, os demais pode deixar em branco.

*Obrigatório

Seu nome *

A Fazenda
Ex: 5

A Grande Família

A Praça é Nossa

Auto esporte

Avenida Brasil

Balacobaco

Band Kids

Big Brother

Bom Dia & Cia

Caldeirão do Huck

Câmera Record

Chaves

Cheias de Charme

Cine belas artes

Cine espetacular

CQC

CSI

De frente com Gabi

Domingo Espetacular

Esporte Fantástico

Esquenta

Estrelas

Eu, A Patroa e as Crianças

Fantastico

Faustão

Fina Estampa

Globo Esporte

Globo Repórter

House

Jogo Aberto

Jogos de futebol

Jornal da Band

Jornal da Record

Jornal do SBT

Jornal Nacional

JPB

Legendarios

Mais você

Malhação

MTV Hits

MTV Shows

O Encantador de Cães

Programa do Jô

O melhor do Brasil

Olhar Digital

Os Simpsons

Pânico na Band

Prison Break

Programa Sílvio Santos

Rebelde

Record Kids

Sábado Animado

Salve Jorge

Sessão da tarde

Tela quente

The bing bang theory

The Walking Dead

Todo mundo odeia o Chris

Top 10 MTV

Tudo é possível

Tv Xuxa

Video Show

Viola, Minha Viola

Zorra Total

Outros

Ex: Friends(5), A tarde é sua(3), Jornal Hoje(4)

Apêndice B

Resultados Obtidos

Na Tabela B.1 e Tabela B.2 estão expostos respectivamente os resultados gerais referentes a média das precisões obtidas na recomendação de filmes e livros para todos os usuários da base usando as abordagens de classificação e recomendação de termos propostas neste trabalho e sem essas abordagens (usando apenas os termos presentes nos históricos dos usuários), em que listas de termos recomendados foram geradas e a partir desses termos foram recomendados filmes e livros. Dentre todos os filmes da base foram recomendados os 5 mais bem avaliados, e a precisão foi calculada observando quantos desses 5 filmes possuíam a mesma categoria da lista de termos recomendada, por exemplo, a partir da lista de termos de ação foi avaliada a quantidade de filmes de ação que foi retornada nas 5 primeiras posições da recomendação final e calculada a precisão. Na recomendação de livros a precisão foi calculada baseando-se nos 4 itens mais bem avaliados.

Tabela B.1: Resultados obtidos da recomendação de filmes

	Categorias					
	Ação	**Aventura**	**Comédia**	**Drama**	**Policial**	**Suspense**
Precisão da Recomendação de Filmes com Classificação de Termos	66%	44%	34%	23%	44%	38%
Precisão da Recomendação de Filmes sem Classificação de Termos	50%	45%	34%	26%	20%	30%

Tabela B.2: Resultados obtidos da recomendação de livros

	Categorias					
	Ação	**Aventura**	**Comédia**	**Drama**	**Policial**	**Suspense**
Precisão da Recomendação de Livros com Classificação de Termos	47%	33%	20%	7%	44%	74%
Precisão da Recomendação de Livros sem Classificação de Termos	42%	25%	14%	29%	23%	38%

Além dos resultados gerais em que calculou-se a média geral das precisões obtidas (TabelaB.1 e Tabela B.2), os resultados para cada usuário e categoria da base podem ser vistos na Tabela B.3 (Precisão das recomendações de filmes para cada categoria com classificação de termos), Tabela B.4 (Precisão das recomendações de filmes para cada categoria sem classificação de termos), Tabela B.5 (Precisão das recomendações de livros para cada categoria com classificação de termos) e Tabela B.6 (Precisão das recomendações de livros para cada categoria sem classificação de termos)

Tabela B.3: Precisão das recomendações de filmes para cada categoria com classificação de termos.

Usuários	**Ação**	**Aventura**	**Comédia**	**Drama**	**Policial**	**Suspense**
1	0.8	0.4	0.4	0	0.4	0.4
2	0.6	0.4	0.4	0.2	0.4	0.4
3	0.8	0.4	0.4	0.2	0.6	0.4
4	0.6	0.4	0.4	0.6	0.4	0.4
5	0.4	0.2	0.2	0.2	0.4	0.2
6	0.4	0.4	0.4	0.4	0.4	0.4
7	0.8	0.6	0.4	0.2	0.4	0.4
8	0.8	0.8	0.4	0.2	0.4	0.4
9	0.6	0.4	0.4	0.2	0.6	0.4

Continua na próxima página

Tabela B.3 – continuação da página anterior

Usuários	Ação	Aventura	Comédia	Drama	Policial	Suspense
10	0.8	0.6	0.4	0.2	0.4	0.4
11	0.8	0.8	0.4	0.2	0.4	0.4
12	0.6	0.2	0.4	0.2	0.6	0.4
13	0.8	0.4	0.2	0	0.4	0.4
14	0.8	0.6	0.4	0.2	0.4	0.4
15	0.8	0.6	0.4	0.2	0.6	0.4
16	0.4	0.4	0.2	0.2	0.4	0.4
17	0.8	0.4	0.4	0.2	0.4	0.4
18	0.8	0.2	0.4	0.2	0.4	0.4
19	0.8	0.6	0.2	0.2	0.4	0.4
20	0.6	0.4	0.4	0.2	0.4	0.4
21	0.6	0.4	0.4	0.2	0.6	0.4
22	0.6	0.4	0.4	0.2	0.4	0.4
23	0.8	0.4	0.2	0	0.4	0.4
24	0.8	0.8	0.4	0.4	0.4	0.4
25	0.8	0.6	0.2	0.2	0.4	0.4
26	0.8	0.4	0.4	0.4	0.6	0.4
27	0.8	0.6	0.4	0.2	0.4	0.4
28	0.4	0.4	0.2	0.2	0.4	0.2
29	0.8	0.4	0.4	0.2	0.4	0.4
30	0.6	0.4	0.4	0.4	0.4	0.4
31	0.6	0.4	0.2	0.2	0.6	0.4
32	0.6	0.6	0.4	0	0.6	0.4
33	0.6	0.4	0.4	0.4	0.4	0.4
34	0.6	0.4	0.4	0.2	0.4	0.4
35	0.8	0.4	0.4	0.2	0.4	0.4
36	0.6	0.2	0.4	0.2	0.4	0.4
37	0.4	0.4	0.2	0.2	0.4	0.2

Continua na próxima página

Tabela B.3 – continuação da página anterior

Usuários	Ação	Aventura	Comédia	Drama	Policial	Suspense
38	0.8	0.4	0.4	0.2	0.4	0.4
39	0.6	0.4	0.2	0.6	0.6	0.4
40	0.6	0.4	0.2	0.4	0.4	0.4
41	0.6	0.4	0.4	0.2	0.4	0.4
42	0.8	0.8	0.4	0.4	0.4	0.4
43	0.6	0.2	0.2	0	0.6	0.4
44	0.8	0.6	0.2	0.4	0.6	0.4
45	0.8	0.4	0.4	0.4	0.4	0.4
46	0.8	0.4	0.4	0.2	0.4	0.4
47	0.8	0.8	0.2	0.2	0.6	0.4
48	0.8	0.2	0.6	0.2	0.4	0.4
49	0.6	0.8	0.4	0.2	0.4	0.4
50	0.8	0	0.4	0.4	0	0.4
51	0.2	0.2	0.2	0.2	0.4	0.4
52	0.6	0.4	0.4	0	0.4	0.4
53	0.8	0.4	0.4	0.2	0.4	0.4
54	0.6	0.2	0.2	0.2	0.4	0.4
55	0.4	0.2	0.4	0.4	0.4	0.4
56	0.6	0.4	0.4	0.2	0.4	0.4
57	0.6	0.4	0.4	0.4	0.4	0.4
58	0.2	0.4	0.4	0.6	0.6	0.2
59	0.6	0.6	0.2	0.2	0.4	0.4
60	0.8	0.2	0.2	0.2	0.4	0.2
61	0.8	0.6	0.4	0	0.4	0.4
62	0.6	0.4	0.2	0.2	0.6	0.4
63	0.6	0.8	0.2	0.2	0.6	0.4

Tabela B.4: Precisão das recomendações de filmes para cada categoria sem classificação de termos.

Usuários	Ação	Aventura	Comédia	Drama	Policial	Suspense
1	0.8	0.8	0.2	0.2	0	0.4
2	0.4	0.2	0.4	0.4	0.4	0.4
3	0.6	0.8	0.4	0.2	0	0.2
4	0.4	0.2	0.6	0	0.2	0.2
5	0.6	0.6	0.2	0.2	0.2	0.6
6	0.2	0.4	0.2	0.6	0.4	0.6
7	0.4	0.2	0.4	0.2	0.4	0.4
8	0.8	0.6	0.2	0.2	0.2	0.4
9	0.2	0.2	0.6	0.2	0.4	0.2
10	0.6	0.4	0.4	0.2	0.2	0.2
11	0.4	0.6	0.2	0.4	0.2	0
12	0.8	0.6	0.2	0.2	0	0.4
13	0.4	0.4	0.4	0.2	0.2	0.2
14	0.6	0.4	0.2	0.4	0.2	0.4
15	0.4	0.4	0.4	0.2	0.2	0.2
16	0.4	0.4	0.2	0.4	0.4	0.4
17	0.4	0.2	0.4	0.4	0.2	0.4
18	0.6	0.4	0.4	0.2	0	0.2
19	0.4	0.4	0.4	0.2	0.2	0.2
20	0.6	0.4	0.4	0	0.2	0.4
21	0.8	0.6	0	0.4	0.2	0.6
22	0.8	0.4	0.2	0	0	0.6
23	0.4	0.2	0.4	0.2	0.4	0.4
24	0.6	0.4	0.4	0.2	0.2	0.4
25	0.4	0.2	0.6	0	0.2	0.2
26	0.6	0.4	0.4	0	0.2	0.2

Continua na próxima página

Tabela B.4 – continuação da página anterior

Usuários	Ação	Aventura	Comédia	Drama	Policial	Suspense
27	0.4	0.4	0.4	0.2	0.2	0.4
28	0.4	0.8	0.2	0.4	0.2	0
29	0.6	0.6	0.4	0.2	0	0.4
30	0.6	0.4	0.4	0.2	0	0.2
31	0.6	0.4	0	0.6	0.4	0.6
32	0.4	0.2	0.6	0	0.2	0.2
33	0.6	0.6	0.4	0.2	0.2	0.2
34	0.6	0.4	0.4	0.2	0.2	0.2
35	0.6	0.4	0.4	0.2	0.2	0.4
36	0.8	0.6	0.2	0.2	0.2	0.4
37	0.4	0.4	0	0.6	0.4	0.4
38	0.4	0.2	0.4	0.2	0.4	0.4
39	0.6	0.6	0.4	0	0	0.4
40	0.6	0.6	0	0.6	0.2	0.4
41	0.8	0.6	0.2	0.2	0	0.4
42	0.2	0.4	0.6	0.2	0.2	0
43	0.2	0.4	0.4	0.4	0.4	0.2
44	0.2	0.6	0.2	0.6	0.2	0
45	0.8	0.6	0.2	0.2	0	0.4
46	0.6	0.4	0.4	0.2	0	0.4
47	0.6	0.6	0	0.6	0.2	0.4
48	0.6	0.6	0.4	0.2	0.2	0.2
49	0.2	0.2	0.6	0.2	0.4	0.2
50	0.4	0.8	0.2	0.4	0	0
51	0.2	0.6	0.4	0.4	0.2	0
52	0.6	0.4	0.4	0.2	0.2	0.2
53	0.6	0.6	0.4	0.2	0.2	0.2
54	0.6	0.4	0.2	0.2	0.2	0.6

Continua na próxima página

Tabela B.4 – continuação da página anterior

Usuários	Ação	Aventura	Comédia	Drama	Policial	Suspense
55	0.4	0.4	0.4	0.2	0.2	0.2
56	0.6	0.4	0.2	0.4	0.2	0.2
57	0.6	0.8	0.4	0.2	0	0
58	0.2	0.4	0.4	0.4	0.2	0.4
59	0.6	0.6	0.4	0.2	0.2	0.2
60	0.2	0.2	0.4	0.4	0.6	0.2
61	0.2	0.4	0.6	0.2	0.2	0.2
62	0.4	0.2	0.6	0.2	0.2	0.4
63	0.4	0.2	0.4	0.2	0.4	0.4

Tabela B.5: Precisão das recomendações de livros para cada categoria com classificação de termos.

Usuários	Ação	Aventura	Comédia	Drama	Policial	Suspense
1	0.5	0.5	0.25	0	0.5	0.75
2	0.25	0.25	0.25	0	0.5	0.75
3	0.5	0.5	0.25	0	0.5	0.75
4	0.5	0.25	0	0.25	0.5	0.75
5	0.5	0.5	0	0.25	0.5	1
6	0.25	0.25	0.25	0	0.5	0.75
7	0.5	0.25	0.25	0	0.5	0.75
8	0.5	0.25	0.25	0	0.25	0.75
9	0.5	0.5	0.25	0	0.5	1
10	0.5	0.25	0.25	0	0.5	0.75
11	0.5	0.25	0.25	0	0.25	0.75
12	0.5	0.25	0.25	0	0.25	0.75

Continua na próxima página

Tabela B.5 – continuação da página anterior

Usuários	Ação	Aventura	Comédia	Drama	Policial	Suspense
13	0.5	0.25	0.25	0	0.5	0.75
14	0.5	0.25	0.25	0	0.5	0.75
15	0.25	0.25	0.25	0	0.25	0.75
16	0.5	0.5	0	0.25	0.5	0.75
17	0.5	0.25	0.25	0	0.25	0.75
18	0.75	0.25	0.25	0	0.75	0.5
19	0.5	0.25	0.25	0	0.5	0.75
20	0.5	0.25	0.25	0	0.5	1
21	0.5	0.5	0.25	0	0.5	0.75
22	0.25	0.25	0.25	0.25	0.5	0.75
23	0.5	0.5	0.25	0	0.5	0.75
24	0.5	0.25	0.25	0	0.25	0.75
25	0.5	0.5	0.25	0	0.5	0.75
26	0.5	0.25	0.25	0	0.5	0.75
27	0.5	0.25	0.25	0	0.5	0.75
28	0.5	0.5	0	0.25	0.5	1
29	0.5	0.25	0.25	0	0.5	0.75
30	0.5	0.5	0	0.25	0.5	0.5
31	0.25	0.25	0.25	0	0.5	1
32	0.5	0.25	0.25	0	0.5	0.75
33	0.75	0.5	0.25	0.25	0.5	0.75
34	0.5	0.25	0.25	0	0.5	1
35	0.5	0.25	0.25	0	0.5	0.75
36	0.25	0.25	0.25	0	0.5	0.75
37	0.75	0.5	0	0.25	0.5	0.75
38	0.5	0.5	0.25	0	0.5	0.75
39	0.5	0.25	0.25	0.25	0.5	0.75
40	0.25	0.25	0	0.25	0.5	0.75

Continua na próxima página

Tabela B.5 – continuação da página anterior

Usuários	Ação	Aventura	Comédia	Drama	Policial	Suspense
41	0.5	0.25	0.25	0	0.25	0.75
42	0.5	0.25	0.25	0	0.25	0.5
43	0.25	0.25	0.25	0	0.5	0.75
44	0.5	0.25	0	0	0.25	0.75
45	0.5	0.25	0.25	0	0.5	0.75
46	0.5	0.25	0.25	0	0.5	0.75
47	0.5	0.25	0.25	0.25	0.25	0.75
48	0.5	0.25	0.25	0	0.25	0.75
49	0.5	0.25	0.25	0	0.25	0.5
50	0.5	0.5	0	0	0.25	0.5
51	0.5	0.5	0.25	0.25	0.5	0.75
52	0.5	0.25	0	0.25	0.5	0.5
53	0.5	0.5	0.25	0	0.5	0.75
54	0.25	0.25	0.25	0	0.25	0.5
55	0.5	0.5	0	0.25	0.5	0.5
56	0.5	0.5	0.25	0	0.5	0.75
57	0.25	0.25	0.25	0.25	0.5	0.75
58	0.75	0.5	0	0.25	0.5	0.5
59	0.25	0.25	0.25	0	0.5	1
60	0.75	0.5	0.25	0.25	0.5	1
61	0.5	0.5	0.25	0	0.5	0.5
62	0.25	0.25	0	0.25	0.5	0.5
63	0.25	0.25	0.25	0	0.25	1

Tabela B.6: Precisão das recomendações de livros para cada categoria sem classificação de termos.

Usuários	Ação	Aventura	Comédia	Drama	Policial	Suspense
1	0.25	0.25	0	0.25	0.25	0.5
2	0.25	0	0	0.25	0.5	0.5
3	0.25	0.5	0	0.5	0.25	0.25
4	0.5	0.25	0.25	0.25	0.25	0.5
5	0.5	0.5	0	0.25	0	0.5
6	0	0.25	0.5	0.5	0.25	0.25
7	0.25	0.25	0.25	0.25	0	0.25
8	0	0	0.25	0.5	0.25	0.25
9	0.5	0.5	0.25	0	0.25	0.5
10	0	0	0.25	0.25	0.25	0.5
11	0.25	0.25	0	0.5	0	0.5
12	0.25	0.25	0	0.75	0.25	0.25
13	0.5	0	0.25	0.25	0.25	0.5
14	0.75	0.5	0	0.25	0.25	0.25
15	0.5	0	0.25	0.25	0.5	0.25
16	0.5	0.25	0	0	0.5	0.5
17	0.75	0.25	0	0.25	0.25	0.25
18	0.5	0.25	0.25	0.25	0.25	0.25
19	0.5	0.25	0.25	0	0.25	0.75
20	0.75	0.25	0	0.25	0.25	0.25
21	0.25	0.5	0.25	0.75	0	0
22	0.25	0.5	0	0.5	0	0.25
23	0.5	0.25	0.25	0.25	0.25	0.25
24	0.75	0.25	0	0.25	0.5	0.5
25	0.25	0.25	0.25	0.25	0	0.25
26	0.5	0.25	0	0.25	0	0.5

Continua na próxima página

Tabela B.6 – continuação da página anterior

Usuários	Ação	Aventura	Comédia	Drama	Policial	Suspense
27	0.5	0.25	0	0.25	0.25	0.25
28	0.25	0.25	0	0.25	0.25	0.75
29	0.5	0.25	0.25	0.25	0.25	0.25
30	0.25	0.25	0	0.5	0	0.5
31	0.5	0.5	0	0.5	0.25	0.25
32	0.5	0.25	0	0.25	0.25	0.25
33	0.5	0.25	0.25	0.5	0.25	0
34	0.5	0.25	0	0.25	0.25	0.25
35	0.5	0.25	0	0.25	0.5	0.5
36	0.5	0	0.25	0.25	0.25	0.25
37	0.75	0.5	0	0	0.25	0.75
38	0.5	0.25	0.25	0	0.25	0.25
39	0.25	0.25	0.5	0.5	0	0.25
40	0.25	0	0	0.5	0.25	0.25
41	0.75	0.75	0	0.25	0	0.5
42	0.25	0.25	0.25	0.25	0	0.5
43	0.75	0.25	0	0	0.5	0.75
44	0	0	0.25	0.5	0	0.25
45	0.5	0.25	0.25	0.25	0.25	0
46	0.25	0.25	0.25	0.5	0	0.25
47	0.5	0.25	0	0.25	0.25	0.5
48	0.5	0.25	0.25	0.25	0.25	0.25
49	0.5	0.25	0	0.25	0.25	0.25
50	0	0	0.5	0.5	0	0.25
51	0.25	0	0.25	0.25	0.25	0.5
52	0.25	0	0.5	0.5	0.25	0
53	0.25	0	0	0.5	0.5	0.5
54	1	0.5	0	0	0.5	0.5

Continua na próxima página

Tabela B.6 – continuação da página anterior

Usuários	Ação	Aventura	Comédia	Drama	Policial	Suspense
55	0.25	0.25	0.25	0	0.25	0.75
56	1	0.5	0	0	0.25	0.5
57	0.25	0.5	0.25	0.25	0	0.25
58	0.5	0.25	0	0	0.5	0.75
59	0.5	0.25	0.25	0.5	0.25	0.25
60	0.5	0.25	0.25	0	0.25	0.75
61	0.5	0	0	0.25	0.25	0.5
62	0.25	0.25	0.25	0.25	0.25	0.5
63	0.25	0.25	0.25	0.25	0.25	0.25