UNIVERSIDADE FEDERAL DE OURO PRETO
INSTITUTO DE CIÊNCIAS EXATAS E BIOLÓGICAS
DEPARTAMENTO DE COMPUTAÇÃO

VICTOR CARLOS GIVISIEZ DE FREITAS
Orientador: Prof. Dr. Reinaldo Silva Fortes

UMA ABORDAGEM DE RECOMENDAÇÃO BASEADA EM CONTEÚDO ATRAVÉS DE PERFIS DICOTÔMICOS DOS USUÁRIOS

Ouro Preto, MG
2024

UNIVERSIDADE FEDERAL DE OURO PRETO
INSTITUTO DE CIÊNCIAS EXATAS E BIOLÓGICAS
DEPARTAMENTO DE COMPUTAÇÃO

VICTOR CARLOS GIVISIEZ DE FREITAS

UMA ABORDAGEM DE RECOMENDAÇÃO BASEADA EM CONTEÚDO ATRAVÉS DE PERFIS DICOTÔMICOS DOS USUÁRIOS

Monografia apresentada ao Curso de Ciência da Computação da Universidade Federal de Ouro Preto como parte dos requisitos necessários para a obtenção do grau de Bacharel em Ciência da Computação.

Orientador: Prof. Dr. Reinaldo Silva Fortes

Ouro Preto, MG
2024

MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DE OURO PRETO
REITORIA
INSTITUTO DE CIENCIAS EXATAS E BIOLOGICAS
DEPARTAMENTO DE COMPUTACAO

FOLHA DE APROVAÇÃO

Victor Carlos Givisiez de Freitas

Uma abordagem de recomendação baseada em conteúdo através de perfis dicotômicos dos usuários

Monografia apresentada ao Curso de Ciência da Computação da Universidade Federal
de Ouro Preto como requisito parcial para obtenção do título de Bacharel em Ciência da Computação

Aprovada em 16 de Outubro de 2024.

Membros da banca

Reinaldo Silva Fortes (Orientador) - Doutor - Universidade Federal de Ouro Preto
Anderson Almeida Ferreira (Examinador) - Doutor - Universidade Federal de Ouro Preto
Pedro Henrique Lopes Silva (Examinador) - Doutor - Universidade Federal de Ouro Preto

Reinaldo Silva Fortes, Orientador do trabalho, aprovou a versão final e autorizou seu depósito na Biblioteca Digital de Trabalhos de Conclusão de Curso da UFOP em 16/10/2024.

Documento assinado eletronicamente por **Reinaldo Silva Fortes**, **PROFESSOR DE MAGISTERIO SUPERIOR**, em 16/10/2024, às 16:16, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do Decreto nº 8.539, de 8 de outubro de 2015.

A autenticidade deste documento pode ser conferida no site http://sei.ufop.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0 , informando o código verificador **0789790** e o código CRC **C5444FDB**.

R. Diogo de Vasconcelos, 122, - Bairro Pilar Ouro Preto/MG, CEP 35402-163
Telefone: 3135591692 - www.ufop.br

Dedico este trabalho aos meus pais, Carlos e Vitória, irmãos Müller e Karlla, e a todos que estiveram comigo durante essa etapa.

Agradecimentos

Primeiramente, agradeço aos meus pais por confiarem em mim e sempre me incentivarem em qualquer decisão. Aos meus irmãos por todo carinho e apoio.

Agradeço ao meu orientador, Reinaldo Silva Fortes por toda paciência e apoio nas indecisões. Agradeço aos docentes e servidores da UFOP por todo conhecimento que me foi transmitido. Agradeço a todos os amigos que pelo companheirismo fizeram os dias em Ouro Preto muito mais fáceis.

"*All we have to decide is what to do with the time that is given us.*"(TOLKIEN, 1954)

Resumo

Sistemas de recomendação surgiram da necessidade dos usuários em encontrar de maneira prática e rápida um conteúdo que lhes interessa na Internet, mas que teriam dificuldade de encontrar devido ao grande volume de opções. Sendo assim surgiram vários métodos de filtragem para auxiliar nesse processo de recomendação. Dentre os métodos existentes, este trabalho busca explorar e desenvolver técnicas que abordam a utilização do método de filtragem baseada em conteúdo. Foram apresentadas diversas técnicas utilizadas na filtragem baseada em conteúdo, sendo elas o uso da representação de itens e usuários, entidades fundamentais no processo de recomendação, a extração de informação textual, utilizando técnicas de pré-processamento de dados como o *tf-idf* e como podem ser feitos os cálculos de similaridades. A partir dos conceitos e técnicas aprendidos foi proposto um método que utiliza a criação de dois perfis do usuário, sendo um deles baseados nos itens que o usuário gosta e o outro baseado nos itens que o usuário não gosta, denominados **perfis dicotômicos**. Utilizando os perfis criados, o método considera a recomendação de itens que sejam similares ao perfil do usuário com itens que ele gosta e, ao mesmo tempo, recomendar itens que sejam dissimilares ao perfil do usuário com itens que ele não gosta. Posteriormente validou-se o método criado em um domínio de filmes. Ao utilizar-se de variadas métricas de ranqueamento, capazes de avaliar diferentes critérios de qualidade das recomendações – NDCG (precisão), EPC (novidade) e EILD (diversidade) –, pôde se observar que a utilização dos dois perfis em igual proporção consegue recomendar melhor do que usando apenas um dos perfis ou uma combinação com pesos diferentes para os dois perfis.

Palavras-chave: Sistemas de recomendação; Filtragem baseada em conteúdo; Representação de itens e usuários; Perfis dicotômicos;

Abstract

Recommender Systems arose from the need for users to find content that interests them on the Internet in a practical and fast way but would have difficulty finding it due to the large volume of options. Consequently, several filtering methods have emerged to assist in this recommendation process. Among the existing methods, this paper seeks to explore and develop techniques that approach the use of the content-based filtering method. Several techniques used in content-based filtering were presented, such as the use of the representation of items and users, fundamental entities in the recommendation process, the extraction of textual information, using data pre-processing techniques such as *tf-idf* and as similarity calculations can be made. Based on the concepts and techniques learned, a method was proposed to create two user profiles, one based on items the user like and the other based on items that the user does not like, called **dichotomous profiles**. Using the profiles created, the method considers recommending items that are similar to the user's profile with items he likes and also recommending items that are dissimilar to the user's profile with items he dislikes. Subsequently, we validated the created method on a movie domain. By using various ranking metrics capable of evaluating different quality criteria for recommendations – NDCG (precision), EPC (novelty), and EILD (diversity) –, it was observed that utilizing the two profiles in equal proportions can recommend better than using only one of the profiles or a combination with different weights for the two profiles.

Keywords: Recommender systems; Content-based filtering method; Representation of items and users; Dichotomous profiles;

Lista de Ilustrações

Lista de Tabelas

Lista de Abreviaturas e Siglas

DCG	Discounted Cumulative Gain
DECOM	Departamento de Computação
EILD	Expected Intra-List Distance
EPC	Expected Popularity Complement
IFS	Information Fatigue Syndrome
IoT	Internet of Things
MAE	Mean Absolute Error
NLP	Natural Language Processing
NDCG	Normalized Discounted Cumulative Gain
RMSE	Root Mean Squared Error
RS	Recommender Systems
TF-IDF	Term Frequency–Inverse Document Frequency
UFOP	Universidade Federal de Ouro Preto

Sumário

1 Introdução

Atualmente, tem-se vivido em um mundo digital cada vez mais conectado, onde a utilização da Internet, com o surgimento de diversos mecanismos de compartilhamento de conteúdo, apresenta um grave problema ao usuário na tarefa de recuperar realmente o conteúdo que o interessa. Tal problema, denominado *Sobrecarga de Informação* ou *Information Fatigue Syndrome* (IFS) (LEWIS, 1999), ganhou muita importância na década de 90, uma vez que essa sobrecarga é um dos principais obstáculos a enfrentar para capturar o conhecimento. Nosso cérebro tem dificuldade em manter-se ativo com esse excesso de informação e a quantidade de energia que precisa ser gasta não ajuda a consumir todas as informações. Sendo assim, tem-se como resultado um esgotamento físico e psicológico na busca por informação, o que pode acarretar uma geração incorreta de decisões.

Dado o problema, tem-se a necessidade de encontrar formas de lidar com a quantidade crescente de informação. Várias estratégias têm sido propostas para lidar com o problema, que vão desde a simples criação de menos dados e informações, a recorrer ao desenvolvimento de reconhecimento de padrões que auxiliem o usuário a concentrar de forma mais eficiente seus esforços na busca pelo conhecimento em desenvolvimento (MAYER-SCHöNBERGER; CUKIER, 2013).

Surgem então, os *Recommender Systems* (RS), a partir da necessidade de filtrar a quantidade de opções disponíveis para o usuário, automatizando a geração de recomendações baseadas na análise dos dados (MELVILLE; SINDHWANI, 2017). Tem como principal objetivo filtrar o conteúdo a ser entregue ao usuário, inferir a "preferência" que o usuário daria a um conteúdo para então realizar uma recomendação baseada nessa preferência. Dessa forma, a experiência de consumo na plataforma se torna mais otimizada, produtiva e interessante.

Um sistema de recomendação pode ser utilizado para recomendar conteúdo em diferentes domínios, como livros, músicas, filmes, varejo e notícias. Embora existam particularidades específicas em cada domínio, dois personagens são fundamentais, os usuários do sistema e os conteúdos oferecidos pela plataforma.

Mesmo sendo uma definição simples, existem diversas formas de tratar esse problema, podendo ainda dividir os métodos em diferentes categorias de abordagens, tais como Filtragem Baseada em Conteúdo (*Content-Based*), Filtragem Colaborativa (*Collaborative Filtering*), Filtragem Demográfica (*Demographic*), Filtragem Baseada em Conhecimento (*Knowledge-Based*) e Sistemas Híbridos (*Hybrid*) (D ZANKER M; G, 2010).

Apesar de a Filtragem Colaborativa ser mais explorada na literatura, o presente trabalho concentra seus esforços em estudar os mecanismos de recomendação utilizados na Filtragem Baseada em Conteúdo. Tal motivação explica-se pelo fato de que cada abordagem possui suas

vantagens e desvantagens, podendo ser úteis em diferentes cenários. Sendo assim, acredita-se que melhorias e aprimoramentos na técnica podem oferecer bons resultados conforme o método de avaliação e/ou prover *insights* para aprimoramentos em outras técnicas, tornando os Sistemas de Recomendação mais eficazes e podendo resultar em uma experiência mais personalizada, relevante e satisfatória para os usuários.

Nesse sentido, monstra-se neste trabalho um estudo das técnicas de recomendação mais utilizadas atualmente e trabalhos que justifiquem a utilização da técnica de filtragem baseada em conteúdo, também foram apresentados um conjunto de técnicas de extração de informação, técnicas de processamento textual e cálculo de métricas de avaliação para posteriormente criar uma metodologia que consiste em definir dois perfis de usuários, sendo os dois perfis construídos sobre o conteúdo textual dos itens. Formalizou-se a criação dos dois perfis para o usuário, denominados **perfis dicotômicos**, sendo um deles baseado nos itens que o usuário "gosta" e o outro baseado nos itens que o usuário "não-gosta". A recomendação passa a ser feita de forma a privilegiar itens que sejam mais similares aos itens que o usuário gosta e, ao mesmo tempo, mais dissimilares aos itens que ele não gosta.

Os experimentos mostraram bons resultados de recomendação em três diferentes critérios de qualidade, envolvendo *precisão*, *novidade* e *diversidade* das recomendações, ao explorar recomendações baseadas nos perfis dicotômicos, diferentemente das técnicas clássicas que focam em recomendar apenas itens similares ao que usuário "gosta".

O restante deste capítulo é organizado da seguinte forma: Na Seção 1.1 são apresentadas justificativas para a realização deste trabalho. A Seção 1.2 descreve os objetivos gerais e específicos que busca-se alcançar com a realização deste trabalho; Na Seção 1.3 discorre sobre a organização do trabalho e o conteúdo que será apresentado nos próximos capítulos.

1.1 Justificativa

O desenvolvimento deste trabalho se faz de muita importância para o contexto atual, a Internet apresenta um enorme aumento no número de dados e fenômenos recentes como o surgimento da *Internet of Things* (IoT) têm aumentado muito mais esse número. Esse enorme número de dados pode dificultar a tarefa do usuário em escolher um produto ou serviço dentre a grande variedade que lhe é apresentada, sendo assim identificar e escolher conteúdos relevantes para o usuário é de suma importância.

Há ainda o fato que o maior número de pesquisas no que se refere a sistemas de recomendação se conduzem para a utilização da técnica de filtragem colaborativa. Uma vez que a filtragem baseada em conteúdo não venha sendo tão utilizada, ela ainda pode ser eficaz em determinados contextos e sugerir novas características para as demais técnicas.

A ideia de se explorar itens mais dissimilares ao perfil do usuário também pode gerar

novos *insights* e tornar-se uma boa estratégia de mapeamento de perfil de usuário.

Portanto, este trabalho merece destaque e torna-se justificável por buscar e explorar técnicas que visem obter melhorias no processo de recomendação baseada na filtragem baseada em conteúdo.

1.2 Objetivos

Este trabalho tem por objetivo geral explorar adaptações e novidades nos algoritmos utilizados em sistemas de recomendação que usam abordagens da técnica de filtragem baseada em conteúdo. Para isso inicialmente cria-se uma representação de um modelo clássico de filtragem baseada em conteúdo e a partir disso pode-se fazer variações dos métodos a fim de obter melhores resultados. O modelo proposto é aplicado no domínio de recomendação de filmes, entretanto, outros domínios podem ser facilmente considerados a partir de uma pequena adaptação.

1.2.1 Objetivos Específicos

Os objetivos específicos detalham processos necessários para alcançar o objetivo geral deste trabalho. Sendo assim os definiremos da seguinte maneira:

- Explorar as técnicas mais utilizadas atualmente na filtragem baseada em conteúdo.
- Definir um método baseado em conteúdo, que considere a criação de perfis **dicotômicos** de usuário.
- Avaliar e validar o método proposto utilizando variadas medidas de avaliação de sistemas de recomendação.

1.3 Organização do Trabalho

O restante deste trabalho possui a seguinte organização:

Capítulo 2: É apresentada uma breve história dos Sistemas de Recomendação, uma fundamentação teórica sobre o tema e trabalhos relacionados;

Capítulo 3: Apresenta a arquitetura funcional de um modelo proposto utilizando técnicas da recomendação baseada em conteúdo.

Capítulo 4: Apresenta resultados experimentais realizados sobre a metodologia implementada.

Capítulo 5: É apresentada as considerações finais do trabalho e as propostas de trabalhos futuros.

2 Revisão Bibliográfica

Este capítulo apresenta de forma introdutória os conceitos utilizados em um sistema de recomendação para que, posteriormente, possam ser compreendidas as metodologias utilizadas no trabalho. A Seção 2.1 faz uma breve introdução do conceito de sistemas de recomendação, também apresenta as diferentes técnicas mais comumente utilizadas, assim como suas características e limitações. A Seção 2.2 apresenta os conceitos da representação de itens e usuários, principais entidades em um RS. A Seção 2.3 apresenta algumas técnicas de extração de informação textual, etapas essenciais em sistemas de recomendação que utilizem a técnica de filtragem baseada em conteúdo. A Seção 2.4 apresenta como pode ser feita a avaliação dos resultados obtidos utilizando alguma das técnicas de filtragem de RS. Por fim, a Seção 2.5 apresenta os principais trabalhos realizados por outros autores sobre RS baseada em conteúdo.

2.1 Sistemas de Recomendação

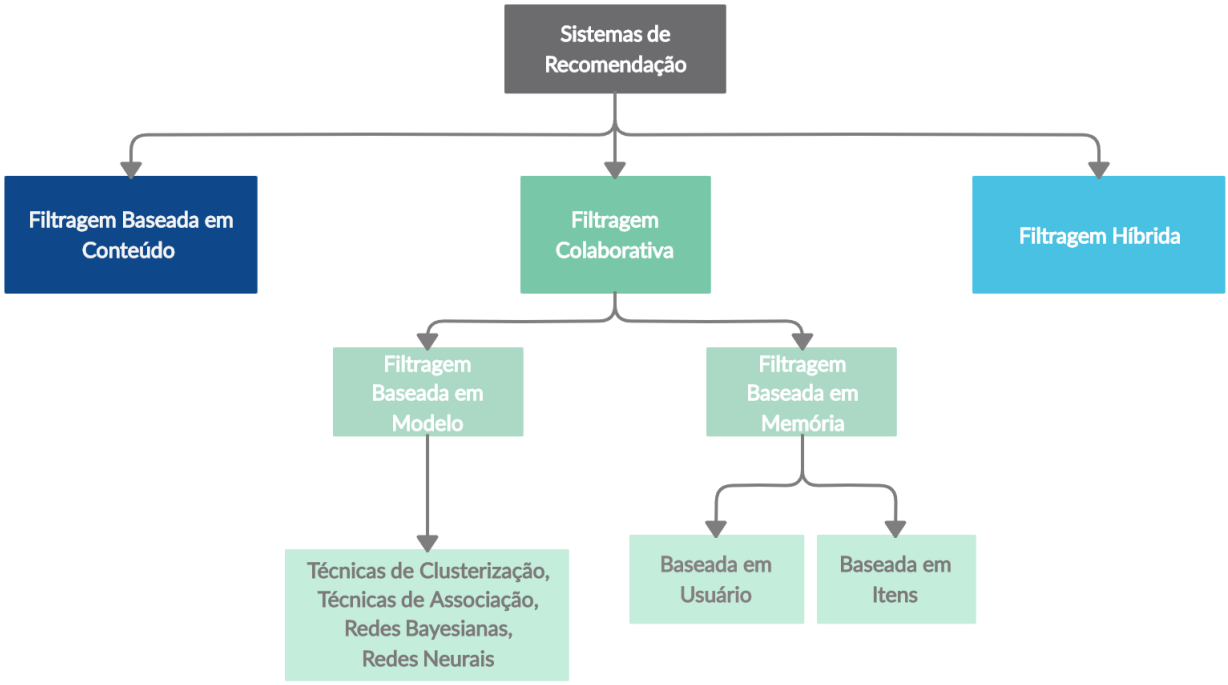
Os Sistemas de Recomendação são um conjunto de técnicas e ferramentas de software que sugerem itens com maior possibilidade de interesse para um usuário específico. As sugestões estão relacionadas com várias características que interferem na tomada de decisão, como quais produtos comprar, que músicas ouvir ou quais notícias ler (RICCI; ROKACH; SHAPIRA, 2011).

Sendo assim, os RS atraem a atenção tanto na academia quanto na indústria por ajudarem a gerenciar a sobrecarga de informação, reunindo informações de forma autônoma e adaptando-as proativamente aos interesses individuais (ADOMAVICIUS; TUZHILIN, 2005), e a consequência disso é que uma vez que aplicando algoritmos eficientes e utilizando corretamente os dados pode-se ter ganhos expressivos, tanto economicamente ao melhorarmos o desempenho do site e aumentando o número de vendas, quanto socialmente ao diminuirmos o tempo das pessoas para encontrar os produtos desejados e melhorando sua satisfação.

Nos problemas de recomendação, as principais entidades são o usuário e o item, tendo como objetivo principal recomendar itens com melhores avaliações aos usuários (TAKAHASHI, 2015). Surgem assim, várias técnicas de recomendação, algumas mais simples, que utilizam apenas avaliações de usuários em itens e outras baseadas em uma ontologia da construção do perfil de usuário, sendo mais dependentes do conhecimento e, por isso, mais complexas.

Assim, como se pode observar na Figura 2.1, dentre as técnicas mais consolidadas há a recomendação baseada em conteúdo, a filtragem colaborativa, a recomendação demográfica, a recomendação baseada em conhecimento e a filtragem híbrida. Essas técnicas serão melhor apresentadas nas subseções a seguir, assim como suas vantagens e limitações.

Figura 2.1 – As principais técnicas de recomendação utilizadas.



Fonte: Elaborado pelo autor.

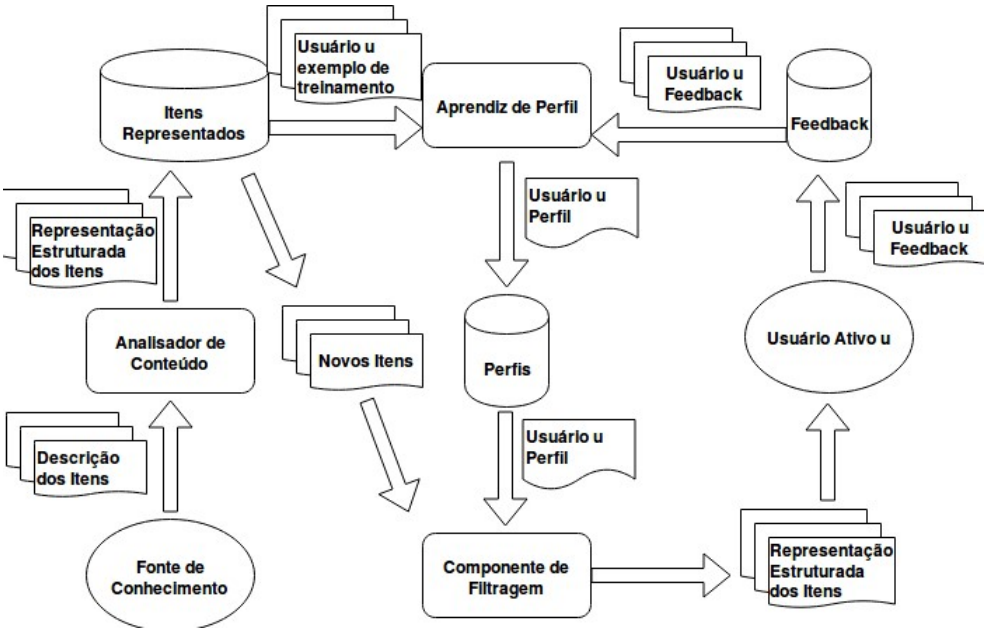
2.1.1 Filtragem Baseada em Conteúdo

A técnica baseada em conteúdo é caracterizada por fazer uma análise de conteúdo nos itens e no perfil do usuário. As informações sobre o perfil do usuário podem ser obtidas pelo próprio usuário, como uma consulta realizada por ele, ou coletadas através do conteúdo dos itens que o usuário consome (HERLOCKER, 2000). O conteúdo nesse tipo de filtragem é geralmente descrito por palavras-chave. A importância de uma palavra em um documento pode ser definida de diferentes formas, sendo a mais comum a TF-IDF, um modelo estatístico definido por SALTON (1989).

A Figura 2.2, utilizada no trabalho de (LOPS; GEMMIS; SEMERARO, 2011), mostra uma arquitetura em alto nível do sistema de recomendação baseado em conteúdo, em que é feita uma demonstração de extração de características dos itens e posteriormente dos usuários. Conforme fluxograma apresentado na figura, é possível observar na etapa inicial a coleta de informações de uma fonte de conhecimento que descreve os itens, posteriormente realiza-se uma análise de conteúdo e a criação de um representação estruturada dos itens. Seguindo, tem-se outro módulo de aprendizagem das preferências do usuário, o módulo utiliza normalmente avaliações já realizadas pelo usuário para inferir suas preferências. Posteriormente tem-se o modulo componente de filtragem, podendo funcionar de maneira contínua na exploração do perfil do usuário e na recomendação de itens relevantes, conforme a similaridade da representação do perfil com a dos itens a serem recomendados.

Dentre as principais vantagens da técnica de filtragem baseada em conteúdo, tem-se o fato de ela não precisar que um item já tenha sido avaliado para gerar a recomendação, além de todos os itens terem chance de serem recomendados, já que dependem apenas do perfil do usuário.

Figura 2.2 – Arquitetura de um sistema de recomendação baseado em conteúdo.



Fonte: (LOPS; GEMMIS; SEMERARO, 2011).

Porém, a técnica baseada em conteúdo apresenta as seguintes limitações (ADOMAVICIUS; TUZHILIN, 2005):

- Análise de conteúdo é limitada: o conteúdo de dados pouco estruturados é difícil de ser analisado. A aplicação da filtragem baseada em conteúdo para extração e análise de conteúdo multimídia, por exemplo (vídeo e som), é muito mais complexa do que a extração e análise de documentos textuais. Outro problema, relativo à análise de conteúdo textual, é que sistemas deste tipo não conseguem distinguir um conteúdo bem escrito de um conteúdo mal escrito se eles utilizam termos muito semelhantes.
- Super especialização: quando o sistema pode recomendar somente itens similares a itens avaliados positivamente, pode ocorrer a super especialização. Desta forma, os itens mais dissimilares ao perfil do usuário não serão apresentados.

2.1.2 Filtragem Colaborativa

A técnica de filtragem colaborativa foi criada para resolver pontos falhos da filtragem baseada em conteúdo (HERLOCKER, 2000). Essa técnica exige classificações dos usuários para

os itens que devem ser recomendados. Eles não exigem descrições e isso é o que os diferencia das técnicas baseados em conteúdo (BRIDGE et al., 2005).

Para utilizar a técnica de filtragem colaborativa, deve-se realizar três etapas (CAZELLA; NUNES; REATEGUI, 2010):

1. Representar os dados de entrada, ou seja, conforme os usuários avaliam alguns itens demonstrando seus interesses, esses dados vão sendo armazenados em um banco de dados;
2. Formar similaridades, conforme o sistema compara o perfil do usuário alvo com o perfil dos demais usuários, cria-se regras de associação válidas para considerá-los vizinhos;
3. Gerar a recomendação, na qual o sistema recomenda itens ao usuário alvo considerando os itens que seus vizinhos mais gostaram.

Como a filtragem colaborativa tem sua essência na troca de experiências entre os usuários que possuem interesses em comum, essa técnica tem se mostrado muito eficiente e por isso vem sendo a mais utilizada (SCHAFER, 2007).

Porém, a técnica baseada em filtragem colaborativa também apresenta suas limitações:

- Problema de *"Cold Start"*: quando um novo usuário entra no sistema, ele não tem interações ou histórico suficientes para que o algoritmo faça recomendações precisas. Sem informações anteriores sobre suas preferências, fica difícil determinar o que recomendar. Podendo ocorrer também quando um novo item é adicionado ao sistema, ele ainda não tem avaliações ou interações de usuários. Consequentemente, o sistema não consegue recomendá-lo até que outros usuários comecem a interagir com ele.
- Problema de Esparsidade: em muitos sistemas, como plataformas de *streaming* ou *e-commerce*, há inúmeros usuários e itens, mas a maioria dos usuários interage com apenas uma pequena fração dos itens. Isso gera uma matriz de interações muito esparsa (ou seja, com poucos dados preenchidos), dificultando a geração de recomendações.

2.1.3 Filtragem Demográfica

Essa técnica recomenda itens baseados nos dados demográficos do perfil do usuário. O perfil do usuário é criado pela classificação dos usuários em estereótipos que representam as características de uma classe de usuários (CAZELLA; REATEGUI, 2005).

Muitos sistemas apresentam soluções de recomendação eficazes baseados em dados demográficos, utilizando, por exemplo, características como a localização, idioma e idade de outros usuários.

Embora essa abordagem tenha sido bastante utilizada em trabalhos de *marketing*, hoje ela é muito pouco utilizada devido às suas limitações e por ter se mostrado menos eficiente em

comparação as outras técnicas, segundo (RICCI; ROKACH; SHAPIRA, 2011) tem havido pouco estudo de aperfeiçoamento na área. Suas principais limitações são:

- Dependência de Dados Demográficos: a filtragem demográfica requer dados explícitos dos usuários e nem sempre é fácil obter essas informações com precisão. Muitos usuários podem não querer fornecer esses dados ou fornecer informações incorretas, comprometendo a eficácia da técnica.
- Inferências Simplificadas: a filtragem demográfica assume que pessoas com atributos demográficos semelhantes compartilham preferências semelhantes, o que muitas vezes não é verdade. Usuários dentro da mesma faixa etária ou grupo de gênero podem ter gostos muito distintos, resultando em recomendações imprecisas.

2.1.4 Filtragem Baseada em Conhecimento

A principal ideia por trás da filtragem baseada em conhecimento é fazer recomendações que correspondam diretamente às necessidades ou preferências explícitas de um usuário, baseando-se em regras ou inferências que consideram as características dos itens e as exigências do usuário.

Segundo (RICCI; ROKACH; SHAPIRA, 2011) a recomendação baseada em conhecimento não estima totalmente a utilidade de um item antes de recomendá-lo a um usuário, uma técnica utilizada é a aplicação de heurísticas para pesar a importância de um item, através do conhecimento adquirido sobre o usuário. O grande problema da técnica baseada em conhecimento seria justamente adquirir o conhecimento do usuário (BUSATTO, 2013).

Os sistemas baseados em conhecimento tendem a funcionar melhor que os outros no início de sua implantação, mas se eles não forem aprimorados com aprendizagem, eles podem ser superados por outros métodos superficiais que podem explorar os registros de interação humano-computador (RICCI; ROKACH; SHAPIRA, 2011). Porém, a técnica baseada em conhecimento pode auxiliar a técnica colaborativa, uma vez que a técnica colaborativa tem dificuldades em iniciar com poucas informações do usuário. A filtragem baseada em conhecimento também apresenta algumas limitações:

- Dependência de Conhecimento de Domínio: o sistema precisa ter conhecimento detalhado dos itens e das preferências dos usuários, o que pode ser desafiador em domínios com muitos itens ou características complexas.
- Dificuldade de Capturar Preferências Implícitas: a técnica depende de preferências explícitas do usuário, podendo não capturar completamente os interesses sutis ou não declarados de um usuário. Em comparação, a filtragem colaborativa consegue captar padrões baseados em comportamento implícito.

- Manutenção de Dados: para manter o sistema atualizado, o conhecimento sobre os itens e as regras que determinam as recomendações precisam ser constantemente revisados e mantidos, o que pode ser caro e trabalhoso.

2.1.5 Filtragem Híbrida

A técnica de filtragem híbrida visa combinar os pontos fortes de diferentes abordagens e até mesmo de algoritmos diferentes em uma mesma abordagem (STERN; HERBRICH; GRAEPEL, 2009).

O intuito de utilização das técnicas híbridas é que uma combinação de algoritmos pode fornecer recomendações mais precisas e efetivas do que um único algoritmo, uma vez que as limitações de uma técnica pode ser superadas por outro algoritmo (SCHAFER; FRANKOWSKI; SEN, 2009).

Várias estratégias de filtragem híbrida são conhecidas e elas se diferenciam conforme seus componentes são combinados, as principais são (BURKE, 2002):

- Ponderada: nesta estratégia a filtragem baseada em conteúdo e a filtragem colaborativa são implementadas separadamente e uma combinação linear é feita com seus resultados. Pode ser necessária uma normalização nos resultados individuais, antes de aplicar a combinação linear, caso as técnicas gerem valores em escalas diferentes.
- Mista: nesta estratégia as combinações são geradas no processo final de recomendação, ou seja, após os resultados gerados pelas técnicas de filtragem baseada em conteúdo e filtragem colaborativa tem-se uma combinação de tal forma que ambas as recomendações sejam apresentadas ao usuário na mesma lista.
- Combinação sequencial: nesta estratégia a filtragem baseada em conteúdo cria os perfis dos usuários e, posteriormente, estes perfis são usados no cálculo da similaridade da filtragem colaborativa.
- Comutação: nesta estratégia o sistema utiliza algum critério, como, por exemplo, a confiança no resultado, para comutar ou chavear entre a filtragem baseada em conteúdo e a filtragem colaborativa.

Assim como as demais técnicas também apresenta as suas limitações:

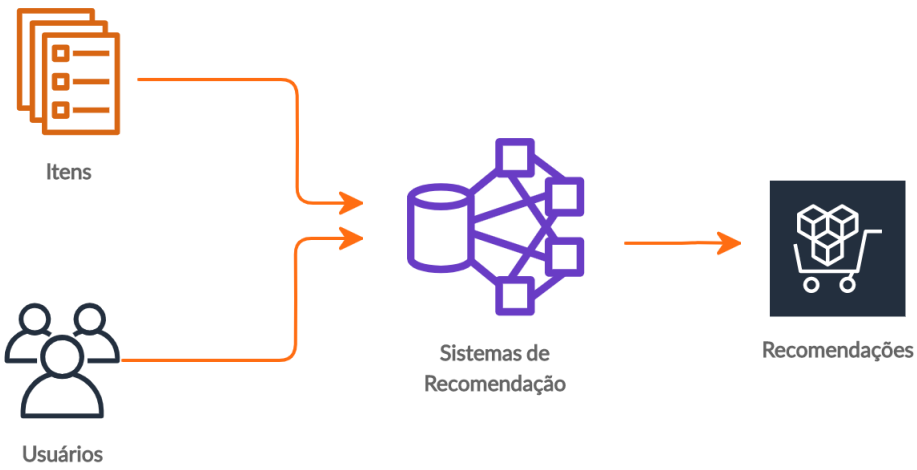
- Complexidade de Implementação: implementar um sistema de filtragem híbrida é tecnicamente complexo, por envolver combinar diferentes métodos de recomendação. Cada técnica possui suas próprias características, e integrá-las de maneira eficiente pode exigir um esforço significativo.

- Combinação de Resultados: a combinação dos resultados das várias abordagens deve ser feita de forma balanceada, para evitar que uma técnica domine a outra ou que o sistema perca sua eficácia. Decidir como pesar as diferentes abordagens pode ser desafiador e depende muito do contexto.
- Escalabilidade: em um sistema com muitos usuários e itens, a filtragem híbrida pode não escalar, pois cada técnica pode exigir diferentes estruturas de dados ou algoritmos de processamento, impactando o desempenho à medida que o sistema cresce.

2.2 Representação de Itens e Usuários

Independentemente da técnica de filtragem utilizada para um sistema de recomendação, ela necessita basicamente de duas entidades de entrada do processo, sendo elas um conjunto de usuários e um conjunto de itens, conforme a Figura 2.3.

Figura 2.3 – Representação itens e usuários.



Fonte: Elaborado pelo autor.

O trabalho da técnica consiste em conhecer, ou aprender, as preferências de cada elemento do conjunto de usuários e, a partir dessas informações, relacionar os elementos mais relevantes do conjunto itens.

Na filtragem baseada em conteúdo mais especificamente, não é suficiente utilizar interações explícitas, como qualificações de natureza binária “gosto” e “não gosto” e interações implícitas, como movimentos do mouse e cliques de acesso. Esses tipos de dados, apesar de serem discretos, não trazem uma boa definição no quesito de auxílio a representatividade dos itens e usuários. Dessa forma, faz-se necessária a utilização de metadados, informação textual e bases de conhecimento.

A Subseção 2.2.1 faz uma melhor contextualização da representação dos itens e a Subseção 2.2.2 apresenta um detalhamento da construção de perfis de usuários.

2.2.1 Representação de itens

Os itens podem ser representados por um conjunto de características ou propriedades (LOPS; GEMMIS; SEMERARO, 2011), por exemplo, filmes podem ser representados pelos atributos de ano, gênero e diretor, sendo assim uma representação bem estruturada e possibilitando a utilização de técnicas de aprendizado de máquina que possam inferir um perfil de usuário (PAZZANI; BILLSUS, 2007).

As técnicas baseadas em conteúdo utilizam na definição de cada item um vetor de palavras-chave, extraídas de textos que descrevem tal item. A construção desses vetores consiste em duas etapas essenciais, a definição de quais palavras são consideradas características e a atribuição de valores, ou pesos, para cada termo.

Na etapa de definição de quais palavras são consideradas características faz-se necessária a utilização de algumas técnicas, como remoção de *stopwords* através de heurísticas, radicalização, lematização e descarte de palavras que possuem sentimentos.

2.2.2 Construção de Perfis de Usuários

Os sistemas de recomendação dependem, na maioria, da qualidade da informação que direta ou indiretamente se consegue obter do usuário. A importância do *feedback* do usuário faz-se necessária para efeito de comparação na etapa de avaliação de novos itens obtidos no resultado de recomendação. Atualmente identifica-se duas maneiras essenciais de se obter o *feedback* proveniente da classificação de itens por parte do usuário (KOREN, 2008):

- Classificações Explícitas: Muito utilizada na maioria dos sistemas, essa classificação consiste na atribuição de valores, podendo ser de natureza binária, como “gosto” e “não gosto”, ou abranger uma faixa de valores discretos, numéricos de 0 a 10. Em sistemas de recomendação baseada em conteúdo tem-se utilizado também o processamento da análise de opiniões em texto livre, as denominadas *reviews*, onde o usuário descreve sua opinião na forma de comentários aos conteúdos.
- Classificações Implícitas: Utilizada normalmente como complemento das classificações explícitas, essa técnica monitora a interação do usuário com o sistema e registra seus padrões de utilização, tem-se como exemplos, movimentos do mouse, padrão de cliques, tempo gasto a ver um item, padrões de navegação.

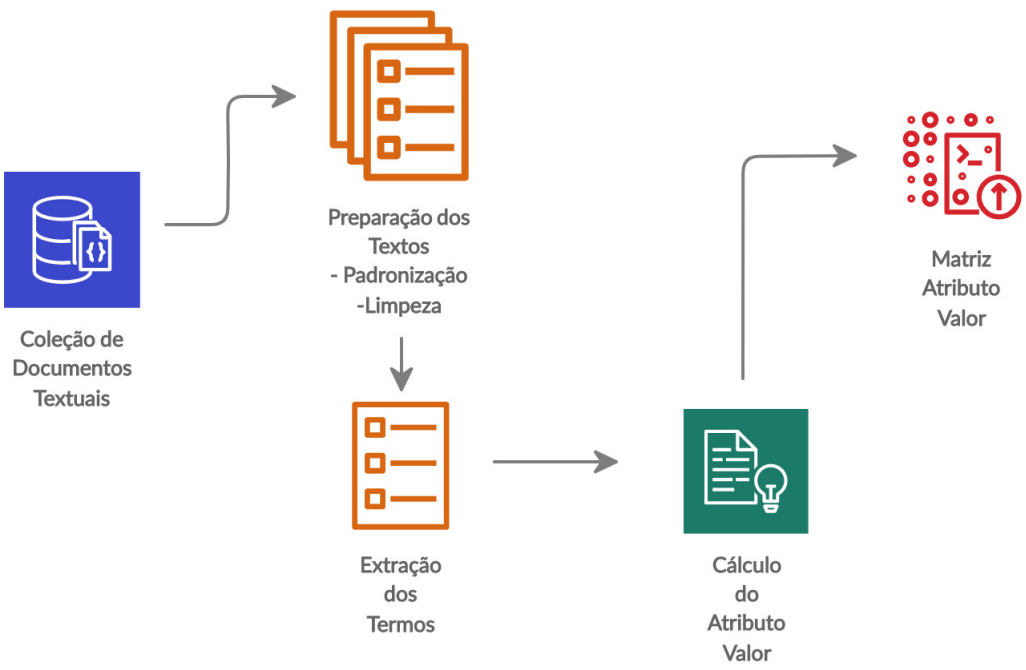
De modo geral, a qualidade das recomendações fornecidas ao usuário depende muito das características do perfil do usuário, sendo assim a precisão dos dados, a quantidade de

informação que ele armazena, e se a informação está atualizada são parâmetros importantíssimos na construção do perfil e melhor será a qualidade da recomendação.

2.3 Extração de Informação de Textos

O objetivo principal dessa etapa é extrair textos escritos em linguagem natural (não estruturados) computacionalmente, e criar uma representação estruturada e manipulável. As subseções seguintes explicam detalhadamente como pode ser feito esse processo. Portanto, a Subseção 2.3.1 monstra algumas técnicas essenciais para redução dos dados, posteriormente as Subseções 2.3.2 e 2.3.3 exploram algumas técnicas para determinação da importância de um termo e, na Subseção 2.3.4, apresentam-se algumas medidas para avaliação do grau de semelhança entre itens. A Figura 2.4 mostra como seria o fluxo dessa etapa de extração de textos.

Figura 2.4 – Pré-processamento de dados textuais.



Fonte: Elaborado pelo autor.

2.3.1 Pré-processamento de Dados Textuais

A etapa de pré-processamento de dados textuais consiste em preparar ou transformar os dados em uma estrutura que o sistema consiga processar. Para isso, além de colocar os dados em uma estrutura padronizada, essa etapa utiliza algumas técnicas para redução de dados para conservar as propriedades principais do conjunto (FELDMAN; SANGER, 2006).

- Amostragem: Consiste em fazer a seleção de um subconjunto de dados, mas que seja relevante e que caracterize o conjunto completo. Existem várias maneiras de se realizar a amostragem, a técnica mais simples consiste na seleção aleatória de amostras, em que os dados possuem a mesma probabilidade de serem selecionados ou técnicas mais refinadas como validação cruzada em que conjuntos de treinamento e teste são construídos várias vezes e sejam avaliados em cada construção.
- Redução de Ruídos: São considerados ruído toda informação que foi de alguma forma representada de maneira errada. Um exemplo de ruído em textos é a escrita errada de palavras. A remoção de ruídos tem o objetivo de minimizar interpretações errôneas do conjunto de dados.
- Redução de Dimensionalidade: No início da análise dos documentos textuais, devemos construir um vocabulário, esse vocabulário inicial contém todos os termos utilizados no conjunto de documentos, sendo assim podem possuir muitos termos com baixa relevância na representação de seus respectivos documentos, acarretando um gasto extra de espaço e processamento. Podemos então utilizar técnicas de redução de dimensionalidade que busquem diminuir a quantidade de termos, sem prejudicar as propriedades do conjunto de dados. Uma maneira muito simples de remover características de baixa relevância é a exclusão de termos comuns na coleção de documentos, esses termos costumam ser artigos e preposições de uma determinada língua, denominados *stopwords* (MANNIG; RAGHAVAN; SHUTZE, 2008).

2.3.2 Tf-Idf

Tf-idf é uma técnica utilizada para se conhecer a importância de um determinado texto em relação a um documento em uma coleção de documentos, utilizando medidas estatísticas temos que o valor *tf-idf* de uma palavra aumenta proporcionalmente à medida que aumenta o número de ocorrências dela em um documento, no entanto, esse valor é equilibrado pela frequência da palavra no conjunto de documentos. Isso auxilia na distinção de palavras mais comuns.

- Frequência de Termo (tf): É a proporção do número de vezes que a palavra ocorre em um documento em comparação com o número total de palavras nesse documento. Aumenta à medida que o número de ocorrências dessa palavra aumenta no texto. O *tf* é definido na Equação 2.1.

$$tf_{i,j} = \frac{n_{i,j}}{\sum_k n_{i,j}} \tag{2.1}$$

- Frequência Inversa de Termos (idf): Usada para calcular o peso de palavras menos comuns em todos os documentos do conjunto. As palavras que ocorrem raramente no conjunto de documentos tem uma relevância maior no *idf*. O *idf* é definido na Equação 2.3.2.

$$idf(w) = log\frac{N}{df_t} \tag{2.2}$$

Combinando ambos os cálculos, temos o *tf-idf* de cada palavra no conjunto de documentos, definido na Equação (2.3):

$$w_{i,j} = tf_{i,j} \times log\frac{N}{df_i} \tag{2.3}$$

tf_{if} = número de ocorrências de *i* em *j*

df_i = conjunto de documentos contendo *i*

N = total de documentos

Dentre as vantagens do *tf-idf* temos sua fácil implementação e melhor performance computacional quando comparado com técnicas mais complexas, como aprendizado profundo, porém uma das suas limitações é o fato de tratar cada palavra de forma independente, não capturando relações semânticas ou contextuais entre elas. Ele não entende sinônimos ou a importância de frases.

2.3.3 Embedding de Palavras

O *embedding* de palavras é uma técnica de representação de palavras em um espaço vetorial de forma que palavras com significados semelhantes tenham vetores próximos entre si. É uma das técnicas fundamentais em *Natural Language Processing* (NLP), por permitir transformar dados textuais em uma forma numérica que algoritmos de aprendizado de máquina conseguem processar.

O *embedding* de palavras surgiu a partir da necessidade de representar palavras de forma que capturassem suas similaridades semânticas e contextuais, superando as limitações dos métodos tradicionais de representação de palavras. Seu funcionamento consiste em transformar cada palavra em um vetor, ou seja, uma lista de números, esses vetores capturam as relações semânticas entre palavras, de modo que palavras com significados semelhantes ou que costumam aparecer em contextos semelhantes fiquem próximas entre si no espaço vetorial.

A técnica tem se mostrado muito importante porque captura informações semânticas, tornando possível que os algoritmos entendam o significado e o contexto das palavras, ao invés de tratá-las como um termo individual. Isso é amplamente utilizado em aplicações de NLP como análise de sentimentos, tradução automática, busca semântica e sistemas de recomendação.

Os trabalhos de Mikolov et al. (2013) desenvolveram o conceito moderno de *embedding* de palavras, sendo revolucionário e marcando o início do uso de redes neurais para a criação de *embedding* de palavras com a introdução do modelo **Word2Vec** em 2013.

2.3.4 Métricas de Similaridade e Dissimilaridade

Para análise das entidades usuário-item presentes nos sistemas de recomendação necessita-se de métricas que avaliem o grau de semelhança entre cada representação, para que sejam produzidos os relacionamentos entre elas. Deste modo, podemos dividir as métricas em dois grupos: de similaridade, na qual os valores altos representam uma maior correlação entre os objetos; e de dissimilaridade, em que valores baixos indicam pouca semelhança entre os objetos.

- *Similaridade Cosseno*: A similaridade de cosseno é muito utilizada na área de recuperação de informação. Essa métrica analisa a correlação entre as instâncias utilizando a angulação entre elas, para isso temos que valores de 1 *(cos(0)* a -1 *cos(180)* sendo quanto maior o valor, menor a angulação e consequentemente, maior é a correlação entre elas. A Similaridade de Cosseno é definida na Equação (2.4).

$$cos(\overrightarrow{i}, \overrightarrow{j}) = \frac{\overrightarrow{i} \cdot \overrightarrow{j}}{\left\|\overrightarrow{i}\right\|_2 \times \left\|\overrightarrow{j}\right\|_2} = \frac{\sum_{n=1}^{k} w_n^i w_n^j}{\sqrt{\sum_{n=1}^{k}(w_n^i)^2}\sqrt{\sum_{n=1}^{k}(w_n^j)^2}} \tag{2.4}$$

- *Correlação de Pearson*: A correlação de Pearson é uma métrica de similaridade muito utilizada na área de sistemas de recomendação (KOREN, 2008). Esta métrica define que o valor 1 corresponde a total correlação, ou seja, as linhas dos vetores se coincidem e estão no mesmo sentido, enquanto o valor -1 corresponde ao caso que as linhas dos vetores se coincidem, porém, estão em sentidos opostos. A Correlação de Pearson é definida na Equação (2.5).

$$p(i,j) = \frac{\sum_{n=1}^{k}(w_n^i - \overline{w_i})(w_n^j - \overline{w_j})}{\sqrt{\sum_{n=1}^{k}(w_n^i - \overline{w_i})^2}\sqrt{\sum_{n=1}^{k}(w_n^j - \overline{w_j})^2}} \tag{2.5}$$

- *Distância Euclidiana*: A distância Euclidiana diferentemente das outras duas métricas abordadas é tida como uma métrica de dissimilaridade, ela realiza uma análise baseada na magnitude da diferença entre os vetores. Sendo assim, quanto maior a distância, maiores são as diferenças entre as instâncias. A Distância Euclidiana é definida na Equação (2.6).

$$d(i,j) = \sqrt{\sum_{n}^{i=1}(w_n^i - w_n^j)^2} \tag{2.6}$$

2.4 Avaliação de Sistemas de Recomendação

A avaliação dos sistemas de recomendação deve ser dada analisando-se um conjunto de propriedades para mensurar o quanto uma técnica se mostra eficiente. A maioria das pesquisas foca especialmente no poder de predição da solução proposta, porém existem outras propriedades que se mostram de suma importância, são elas, a capacidade de apresentar novos itens que não

correspondem aos gostos previamente definidos pelos usuários, mas que possam agradá-los, a diversidade dos itens recomendados, a confiabilidade do sistema ao preservar a privacidade do usuário, a escalabilidade do sistema conforme o aumento de dados, a velocidade em predizer um gosto do usuário (SHANI; GUNAWARDANA, 2011).

Existem ainda vários cenários para a avaliação dos sistemas de recomendação, sendo os mais utilizados a experimentação *offline*, onde são utilizados um conjunto de dados que simulam as interações dos usuários (avaliações, notas e descrições) e o estudo de usuário, na qual se seleciona um conjunto de usuários e é feito um teste de interação com o sistema, analisando posteriormente suas conclusões.

Uma forma de avaliar a qualidade de um sistema de recomendação é comparar as recomendações de um conjunto de teste com as avaliações fornecidas pelos usuários. Tipicamente esses sistemas são avaliados utilizando métricas de precisão, as Subseções 2.4.1, 2.4.2, 2.4.3, 2.4.4, 2.4.5 apresentam algumas das métricas mais utilizadas.

2.4.1 Precisão e Revocação

As métricas de Precisão (*Precision*) e Revocação (*Recall*) estão dentre as técnicas mais utilizadas na literatura quanto a medidas de avaliação de sistemas de recuperação de informação. Elas fazem uma análise quanto a quantidade de dados recuperados e a relevância que esses dados tem para o usuário (HERLOCKER, 2000).

A precisão é utilizada para identificar a proporção de itens relevantes dentre todo o conjunto de itens recomendados e pode avaliar a tarefa de "encontrar bons itens", sua medida é dada pela Equação (2.7):

$$\textit{Precisão} = \frac{I_r}{I} \tag{2.7}$$

Onde I_r representa a quantidade de itens relevantes recomendados e I é o total de itens recomendados.

A revocação, no entanto, é definida como a quantidade de itens relevantes recomendados dentre todo o conjunto de itens relevantes e pode ser utilizada para avaliar a tarefa de "encontrar todos os bons itens", sua medida é dada pela Equação (2.8):

$$\textit{Revocação} = \frac{I_r}{S_r} \tag{2.8}$$

Onde I_r representa a quantidade de itens relevantes recomendados e S_r é o total de itens relevantes para o usuário.

As duas métricas podem ainda ser unidas em uma só através da média harmônica entre elas. Essa nova métrica chamada de F_1 ou *F-Measure* combina precisão e revocação de modo

a trazer um número único que indique a qualidade geral do método, sua medida é dada pela Equação (2.9):

$$F_1 = \frac{2 \times Precisao \times Revocacao}{Precisao + Revocacao} \tag{2.9}$$

2.4.2 MAE e RMSE

O erro absoluto médio (*Mean absolute error* - MAE) e a raiz do erro quadrático médio (*Root mean squared error* - RMSE) são métricas utilizadas para avaliar o desvio médio entre uma classificação prevista com as classificações que realmente foram feitas pelos usuários, mostrando quão próximo estão seus valores.

O MAE é definido como a diferença média absoluta entre avaliações previstas e avaliações reais, e sua medida é dada pela Equação (2.10):

$$MAE = \frac{\sum_{u,i} |p_{u,i} - r_{u,i}|}{N} \tag{2.10}$$

O RMSE no entanto põe mais ênfase em erros absolutos maiores, ou seja, o erro de um ponto aumenta o acumulado de erro em um ponto, um erro em dois pontos acumula quatro pontos na soma, atribuindo assim um peso maior para erros maiores, sua medida é dada pela Equação (2.11):

$$RMSE = \sqrt{\frac{\sum_{u,i} (p_{u,i} - r_{u,i})^2}{N}} \tag{2.11}$$

2.4.3 NDCG

A métrica *Normalized Discounted Cumulative Gain* (NDCG) é definida por medir a eficiência de um sistema de recomendação baseado no grau de relevância dos itens posicionados no início da lista de recomendação, sendo ainda a relevância dos itens definidas como binária ou discreta, dependendo da metodologia utilizada pelo sistema. No caso de utilização de classificação binária, podemos formalizar um item como sendo relevante “1” ou irrelevante “0”.

O sentido de seu surgimento deu-se para situações em que os sistemas de recomendação possuíam medidas não binárias de relevância, tendo-se ainda nessas condições a necessidade que a maioria dos documentos com maior relevância sejam retornados no topo da lista, sua medida é dada pela Equação (2.12):

$$NDCG(n) = \frac{DCG_n}{IDCG_n} \tag{2.12}$$

Onde o *Discounted Cumulative Gain* (DCG) é fundamentado em duas regras, documentos com relevância máxima são mais importantes que documentos de relevância grande e quanto

mais longe da primeira posição da lista um documento estiver, menos relevante o documento será para o usuário. Tendo em vista essas duas regras, a fórmula do DCG é dada pela Equação (2.13):

$$DCG(n) = rel_1 + \sum_{i=2}^{n} \frac{rel_i}{log_2(i)} \tag{2.13}$$

O rel_i é a relevância do documento na posição *i* da lista e *n* é o tamanho da lista de documentos retornados.

O *Ideal DCG* (IDCG) seria a lista ideal de relevância dos documentos retornados, o IDCG é encontrado colocando-se nas primeiras posições os documentos de maior relevância, se ainda houver espaço na lista, preenche-se colocando os documentos de relevância mais próximos de 1 e assim por diante até que se preencha todas as *p* posições possíveis da lista.

As listas de documentos retornados podem variar de comprimento dependendo da consulta. Assim, a comparação do desempenho de um sistema utilizando várias consultas não pode ser consistentemente feita utilizando apenas o DCG. Portanto, é utilizada a medida de avaliação NDCG, onde o DCG da consulta é igualado em um tamanho *n*. Calcula-se também o IDCG até essa mesma posição e realiza-se a razão entre o DCG e o IDCG (JäRVELIN; KEKäLäINEN, 2002).

2.4.4 EPC

A métrica *Expected Popularity Complement* (EPC), conforme descrita por Vargas e Castells (2011), refere-se da novidade ao considerar a popularidade inversa dos itens recomendados. Trata-se de uma métrica de avaliação de sistemas de recomendação que visa mensurar a novidade das recomendações, considerando tanto a relevância dos itens sugeridos quanto a probabilidade de que esses itens não tenham sido previamente conhecidos pelos usuários, sua medida é dada pela seguinte Equação (2.14):

$$EPC = C \sum_{i_k \in R} \text{disc}(k) \cdot p(\text{rel} \mid i_k, u) \cdot (1 - p(\text{seen} \mid i_k)) \tag{2.14}$$

Onde:

- C é uma constante de normalização;
- $i_k \in R$ percorre sobre o conjunto de recomendações R;
- $\text{disc}(k)$ é uma função de desconto, geralmente decrescente em função de k, a posição do item na lista de recomendações;
- $p(\text{rel} \mid i_k, u)$ é a probabilidade de o item i_k ser relevante para o usuário u;

- $p(\text{seen} \mid i_k)$ é a probabilidade de o item i_k já ter sido visto pelo usuário.

Quanto mais próximo de 1 for o resultado da métrica EPC, maior será a novidade das recomendações. Isso indica que o sistema está sugerindo itens mais inovadores ou menos populares entre os usuários. Essa abordagem permite avaliar a capacidade do sistema de fornecer recomendações que se distanciam do comum, introduzindo itens menos conhecidos ao público.

2.4.5 EILD

A métrica *Expected Intra-List Diversity* (EILD) também introduzida por Vargas e Castells (2011), refere-se a forma de avaliar a diversidade dos sistemas de recomendação. A EILD foi desenvolvida para garantir que as listas de recomendações oferecessem uma variedade de itens, incentivando a descoberta de novas opções e evitando que o sistema recomendasse apenas itens muito similares entre si, dada pela seguinte Equação (2.15).

$$EILD = C \sum_{i_k, i_l \in R} \text{disc}(k) \cdot \text{disc}(l \mid k) \cdot p(\text{rel} \mid i_k, u) \cdot p(\text{rel} \mid i_l, u) \cdot d(i_k, i_l) \quad \text{com } k \neq l \tag{2.15}$$

Onde:

- C é uma constante de normalização;
- $\sum_{i_k, i_l \in R}$ percorre sobre os pares de itens pertencentes ao conjunto de recomendações R;
- $\text{disc}(k)$ é uma função de desconto associada à posição k;
- $\text{disc}(l \mid k)$ é a função de desconto para a posição l, condicionada à posição k;
- $p(\text{rel} \mid i_k, u)$ é a probabilidade de o item i_k ser relevante para o usuário u;
- $p(\text{rel} \mid i_l, u)$ é a probabilidade de o item i_l ser relevante para o usuário u;
- $d(i_k, i_l)$ é uma medida de dissimilaridade entre os itens i_k e i_l;
- A condição $k \neq l$ assegura que está sendo considerado apenas pares diferentes de itens.

Quanto maior o valor da EILD, maior a diversidade na lista recomendada. Isso significa que os itens na lista recomendada são, em sua maioria, dissimilares entre si, oferecendo uma variedade maior de opções para o usuário, ajudando a enriquecer a experiência do usuário.

2.4.6 Classificação das métricas

Comparar diferentes métricas pode ser desafiador, especialmente quando as métricas estão em diferentes escalas, unidades ou representam aspectos distintos de desempenho. No entanto, há várias abordagens e técnicas estatísticas que ajudam a realizar comparações significativas. Uma abordagem utilizada seria a classificação das métricas.

Segundo Carterette e Bennett (2008), utilizar medidas de ranqueamento podem mostrar que os resultados possuem uma correlação maior quando comparadas com medidas absolutas, e são mais estáveis quanto a utilização da média.

A classificação facilita a comparação, a análise e a tomada de decisões. Para classificar itens de um conjunto de forma eficaz, é importante seguir um processo sistemático, considerando o tipo de dados, os critérios de classificação e o método de ordenação. A seguir são apresentadas algumas técnicas de classificação mais comumente utilizadas:

- *Standard Competition Ranking (Ranking Padrão de Competição)*: Neste método, quando duas ou mais entidades têm o mesmo valor, elas compartilham a mesma posição. As posições subsequentes são puladas conforme o número de entidades empatadas. Se dois participantes estão empatados em 2º lugar, a próxima posição disponível será a 4ª. Comum em competições e classificações esportivas, onde a ordem é importante, mas a continuidade das posições não é necessária.

- *Modified Competition Ranking (Ranking Modificado)*: Neste método, os saltos nos números do ranqueamento ocorrem antes de conjuntos de itens empatados (ao contrário do *Standard Competition Ranking*, onde os saltos aparecem depois). Essa abordagem garante que um competidor só estará classificado em uma posição se tiver superado um número exato de oponentes — por exemplo, um participante só será classificado em segundo lugar se tiver superado todos, exceto um adversário.

- *Dense Ranking (Ranking Denso)*: As entidades com valores iguais compartilham a mesma posição, e a contagem das posições subsequentes continua sequencialmente, sem pular números. Se dois participantes estão empatados em 2º lugar, a próxima posição será a 3ª, seguindo uma sequência contínua. Frequentemente aplicado em contextos acadêmicos e esportivos onde a clareza e continuidade são essenciais.

- *Ordinal Ranking (Ranking Sequencial)*: Neste método, cada item recebe uma posição única e distinta, mesmo que os valores sejam iguais. As posições são frequentemente atribuídas de forma arbitrária ou por critérios de desempate. Se dois participantes têm o mesmo valor, eles podem ser colocados em posições diferentes (como 2º e 3º), mesmo sem um critério claro de desempate. Utilizado em ranqueamentos onde se deseja uma diferenciação clara entre os itens, independentemente de seu desempenho.

- *Fractional Ranking (Ranking Fracionado)*: Itens empatados recebem a média das posições que ocupariam se não houvesse empates. Isso é feito somando as posições e dividindo pelo número de entidades empatadas. Se dois participantes estão empatados em 2º lugar, ambos receberão a posição 2,5 (a média entre 2 e 3). Útil em análises estatísticas onde se deseja representar o desempenho de forma mais precisa, considerando a distribuição dos valores.

2.5 Trabalhos Relacionados

Nesta seção, é fornecido o embasamento científico necessário para definir a metodologia deste trabalho, com base na descrição de trabalhos correlatos. Uma abundância de estudos são focados em melhorar a técnica de filtragem baseada em conteúdo analisando o conteúdo textual dos itens e/ou utilizando apenas as avaliações positivas do usuário, poucos trabalhos exploram as avaliações negativas do usuário.

Nas subseções seguintes, são apresentados dois trabalhos que se assemelham ao nosso método proposto, pelo fato de considerar a utilização das avaliações negativas dos usuários como uma informação muito importante na descoberta das preferências do usuário. A Subseção 2.5.1 enfatiza como é importante as informações do **feedback** negativo do usuário e como os dados podem ser categorizados e a Subseção 2.5.2 apresenta outros trabalhos que utilizaram as características de dissimilaridade para enriquecer a representação dos itens.

2.5.1 Feedback negativo do usuário

A maioria dos algoritmos de recomendação se concentra em comportamentos positivos, negligenciando o impacto essencial do *feedback* negativo. A avaliação negativa, uma área promissora, ajuda a revelar aspectos negativos dos comportamentos dos usuários, influenciando a otimização dos algoritmos e melhorando a compreensão das preferências dinâmicas dos usuários.

O trabalho de Ma et al. (2018) visa o estudo e a revisão das estratégias de avaliação negativa do usuário e propor direções futuras de pesquisa para aprimorar os sistemas de recomendação. O estudo conseguiu classificar as estratégias de avaliação negativa em cinco categorias:

- *Static Negative Sampling (Estática)*: seleciona as avaliações negativas de forma fixa, com base em critérios como popularidade ou uma distribuição predefinida.
- *Dynamic Negative Sampling (Dinâmica)*: ajusta a seleção de avaliações negativas durante o treinamento com base em critérios como similaridade do usuário ou atributos de conhecimento.
- *Adversarial Negative Generation (Geração Adversarial)*: utiliza técnicas de aprendizado adversarial para gerar avaliações negativas plausíveis, maximizando o desempenho do sistema.

- *Importance Re-weighting (Reponderação de Importância)*: ajusta os pesos das avaliações negativas, priorizando as mais informativas para melhorar a precisão do modelo.
- *Knowledge-enhanced Negative Sampling (Baseada em Conhecimento)*: usa conhecimento externo, como gráficos de conhecimento, para refinar a seleção de amostras negativas.

Segundo Ma et al. (2018), os principais desafios incluem identificar precisamente a avaliação negativa do usuário, balancear a acurácia, eficiência e estabilidade das técnicas e lidar com os diferentes cenários e objetivos, requerendo uma estratégia universal. O estudo ainda indica que as direções futuros devem se concentrar em criar estratégias que melhor se adaptem as mudanças nas preferências do usuário e aos diferentes contextos de recomendação, lembrando também de focar em melhorar a performance e a robustez dos algoritmos.

A avaliação negativa é o elemento crítico e insubstituível na recomendação que pode potencialmente melhorar a modelação das preferências dinâmicas dos usuários com as suas interações esparsas, segundo Ma et al. (2018).

Outro trabalho, publicado recentemente por Wang et al. (2023) aborda a importância de incorporar *feedback* negativo em sistemas de recomendação sequenciais para aprimorar a personalização e a experiência do usuário. Tradicionalmente, as técnicas priorizam a aprendizagem do perfil do usuário a partir de interações positivas, deixando de lado o uso das avaliações negativas, sejam as avaliações realizadas explicitamente, como *dislikes*, ou implicitamente, como *skips*. Essa abordagem pode restringir a capacidade dos sistemas de se ajustarem rapidamente às preferências dos usuários e de evitarem recomendações indesejadas.

Os autores propuseram uma solução baseada em uma nova função de perda denominada **not-to-recommend**, que otimiza a probabilidade de não recomendar itens que receberam *feedback negativo*. Essa abordagem permite que o sistema de recomendação aprenda diretamente com exemplos negativos durante a fase de recuperação de itens, melhorando a capacidade do modelo de alinhar as recomendações às preferências reais dos usuários. Experimentos práticos em um sistema de recomendação de grande escala demonstraram que a incorporação dessa função de perda resultou em uma redução significativa de recomendações indesejadas e aumento da satisfação dos usuários.

2.5.2 Características de dissimilaridade

Uma abordagem apresentada por Zigkolis, Karagiannidis e Vakali (2013) visa dar importância aos valores de dissimilaridade dos itens e seus atributos, nesse sentido ele melhorou a representação dos itens com os recursos fornecidos a fim de aumentar a capacidade de um recomendador em destacar os itens preferidos. O trabalho implementa um framework que segue a seguinte estrutura:

- Comunidades de Usuários: os autores identificam similaridades entre usuários com base em itens avaliados ou em atributos desses itens. A formação de comunidades permite criar perfis mais ricos de usuários, agrupando-os de acordo com suas preferências compartilhadas.
- Extração de Características de Dissimilaridade: para cada item avaliado, características de dissimilaridade são extraídas, considerando as diferenças nas preferências entre um usuário e sua comunidade. Esse processo enriquece a representação dos itens, indo além dos atributos tradicionais.
- Incorporação em Classificadores: essas características são então integradas a classificadores, como árvores de decisão (C4.5) e SVM, para aprimorar o desempenho de sistemas de recomendação.

O trabalho realizou experimentos utilizando dados reais do **Yahoo! Music** e os resultados mostraram que a inclusão de características de dissimilaridade melhora significativamente a precisão dos classificadores de recomendação, especialmente em comparação com métodos tradicionais. A análise indica que esses novos recursos aumentam a relevância das recomendações e oferecem *insights* mais profundos sobre as preferências dos usuários.

Zigkolis, Karagiannidis e Vakali (2013) conclui que as características de dissimilaridade propostas são uma adição valiosa para sistemas de recomendação, permitindo capturar sutilezas nas preferências dos usuários que poderiam passar despercebidas em abordagens tradicionais. O artigo também sugere a possibilidade de expandir esse framework para redes sociais e outras aplicações onde recomendações diversificadas são essenciais.

O trabalho de Wang et al. (2024), publicado recentemente, aborda a utilização das preferências do usuário, em vez de focar apenas em características dos itens, o trabalho considera as preferências persistentes do usuário, que podem mudar ao longo das sessões de uso. A estrutura do método apresentado se baseia em três etapas, sendo a primeira caracterizada pela predição das preferências dos usuários, utilizando um modelo supervisionado e baseando-se em dados de comportamento passado. Na segunda etapa, cada item é associado a uma função de valor, que reflete a probabilidade de satisfação do usuário para diferentes preferências. A etapa final seria o processamento sequencial de um algoritmo onde, a cada posição, um item é escolhido para maximizar a relevância em relação às preferências não utilizadas por itens anteriormente recomendados, ou seja, quando um item de uma preferência específica é recomendado, o sistema reduz a probabilidade de que itens futuros sejam da mesma preferência, incentivando assim a inclusão de itens diversificados. O método foi testado no **Youtube** e os resultados do trabalho evidenciam uma ampliação da diversidade das recomendações e uma melhora na experiência de usuário.

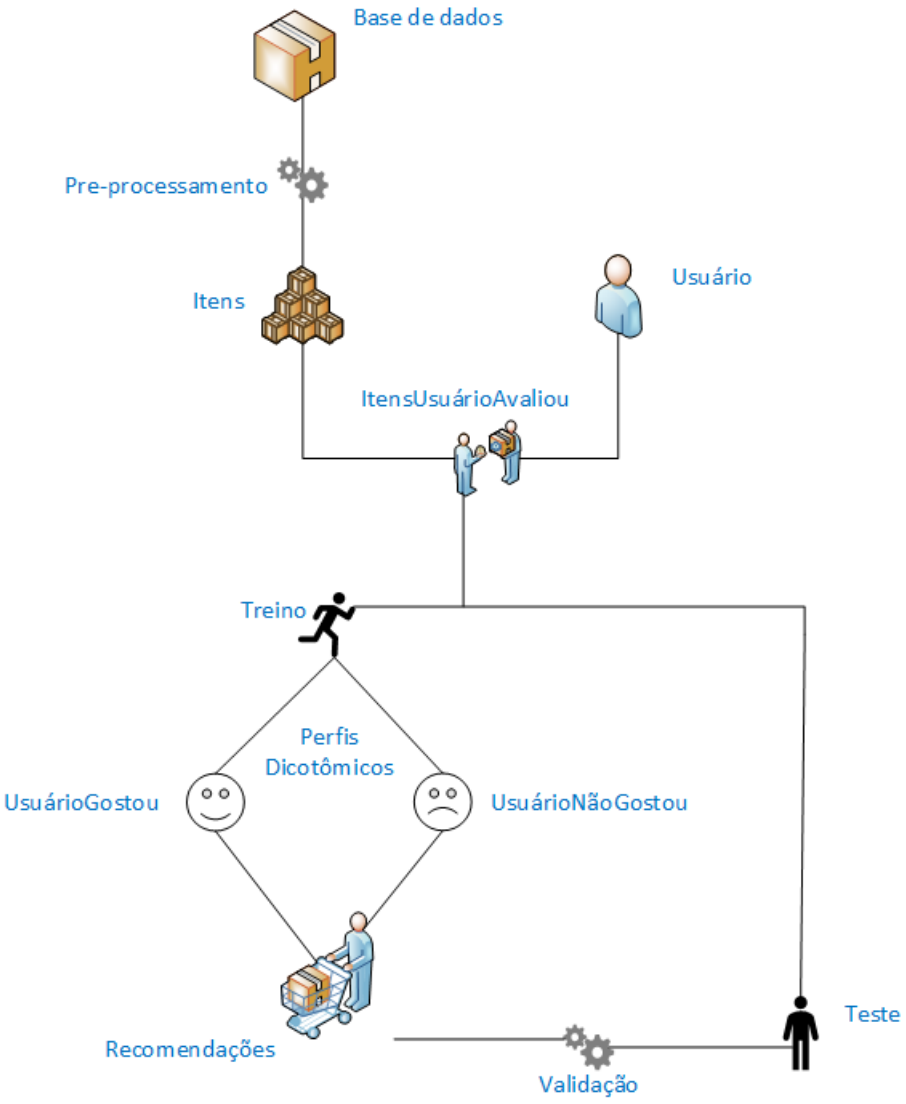
Os trabalhos relacionados apresentados foram de grande importância e auxiliaram no embasamento do método proposto neste trabalho, as características de explorar uma melhor

representação do perfil do usuário com a utilização do *feedback* negativo e a importância da dissimilaridade se assemelham as características deste trabalho.

3 Desenvolvimento

No capítulo anterior, foram apresentados conceitos relacionados diretamente com esse trabalho, os quais fundamentam a construção do método proposto. Neste capítulo será apresentado o processo de construção do método, detalhando cada módulo separadamente e demonstrando como as técnicas de processamento de texto e construção de perfil de usuário se integram à metodologia.

Figura 3.1 – Fluxograma com as etapas de funcionamento do modelo desenvolvido.



Fonte: Elaborado pelo autor.

Como o objetivo deste trabalho é explorar técnicas de sistemas de recomendação baseados na filtragem de conteúdo e partindo do pressuposto que se um usuário gosta de um determinado

item, esse mesmo usuário também gostará de outro item com características semelhantes, tem-se como etapa principal analisar itens que já tiveram avaliações pelos usuários, a Figura 3.1 ilustra o fluxograma completo do método proposto, onde na etapa inicial é selecionada uma base de dados na qual os itens são de representação textual, após o pre-processamento textual podemos separar os itens avaliados por determinado usuário nos subconjuntos de treino e teste, sendo que os itens do subconjunto de treino serão utilizados para construir os perfis **dicotômicos** e os itens do conjunto de teste para realizar as recomendações. Nas subseções seguintes são descritas as etapas de construção do método de forma mais detalhada.

3.1 Domínio da aplicação

Assim como em muitos exemplos de soluções de recuperação de informação, e no que diz respeito aos sistemas de recomendação, é difícil encontrar uma solução única que se encaixe em todas as situações, o ideal é que esses sistemas sejam abordados como um desenvolvimento incremental, construído sobre os dados que estão disponíveis e conforme a área do problema.

A fim de maximizar a eficiência de um método de recomendação, um aspecto crucial é definir o domínio em que se está trabalhando para possibilitar uma melhor compreensão dos dados.

O método proposto neste trabalho visa a utilização de dados na qual os itens tenham um conteúdo textual que o descrevem, como exemplo pode-se ter o resumo de artigos científicos, a descrição de um produto ou o resumo de um filme.

3.2 Preparação dos dados

Considerando que a etapa de obtenção dos dados tenha sido concluída e, portanto, tenham-se os dados disponíveis, é necessário realizar um pré-processamento desses dados a fim de se ter um melhor entendimento das informações. A etapa de pré-processamento tem um alto custo computacional e exige um planejamento cuidadoso para obter um bom desempenho, pois o processo de preparação dos dados consiste numa série de tarefas destinadas a obter um conjunto final de dados na qual será criado e validado o método.

Como o método explora domínios de cunho textual, é comum na etapa de pré-processamento a remoção de *stopwords*. Os termos classificados como preposições, artigos, conjunções normalmente são irrelevantes na definição do item. As *stopwords* são termos muito comuns e ocorrem com muita frequência na língua considerada, portanto sua remoção causa uma redução drástica no conjunto de atributos.

De mesma forma também é removido os termos de pouca relevância, utilizando para isso o (*tf-idf*), a Seção 2.3 explica detalhadamente o funcionamento do processo de remoção utilizando a frequência do termo.

Sendo assim, após toda a etapa de pré-processamento textual dos itens, realizada a limpeza de termos irrelevantes, optou-se por utilizar a técnica de *tf-idf* para o cálculo de importância dos termos no método proposto.

3.3 Divisão de dados em treino e teste

No processo de aprendizagem supervisionada em mineração de dados, é típico que após as etapas de pré-processamento e formatação, os dados sejam fragmentados em dois subconjuntos, denominados "conjunto de treinamento" e "conjunto de testes".

Sobre o "conjunto de treinamento" aplica-se o algoritmo de indução de conhecimento, com isso se obtém um método "treinado", que de certa forma representa o conhecimento extraído. Posteriormente o método obtido é aplicado ao "conjunto de testes", como o "conjunto de testes" também é previamente rotulada, pode-se medir a qualidade do método, comparando-se o resultado obtido com a rotulação disponível no "conjunto de testes" (CABENA et al., 1997).

É importante garantir que os conjuntos de treino e teste contenham a mesma distribuição dos dados, ou seja, de nada adianta treinar o método em um conjunto de dados que não representa o cenário real do problema de estudo.

As principais formas de fragmentação do conjunto de dados são:

- *Hold-out-validation*: Esta é a forma mais simples de separar os dados. Define-se um percentual para cada conjunto de dados (treino e teste) e cria-se as amostras.
- *Cross-validation (Validação Cruzada)*: Esta técnica consiste em dividir a base de dados em **n** partes (*folds*). Destas, **n-1** partes são utilizadas para o treinamento e uma serve como base de testes. O processo é repetido **n** vezes, de forma que cada parte seja usada uma vez como conjunto de testes. Ao final, a correlação total é calculada pela média dos resultados obtidos em cada etapa, obtendo-se assim uma estimativa da qualidade do método gerado e permitindo análises estatísticas.

Portanto, neste trabalho, após a etapa de pré-processamento dos dados textuais, selecionam-se os itens avaliados por cada usuário e realizada uma divisão do conjunto de dados nos subconjuntos de "treino" e "teste", sendo a forma de fragmentação *hold-out-validation* escolhida.

3.4 Perfis dicotômicos de usuário

Uma das estratégias utilizadas em sistemas de recomendação baseada em conteúdo consiste em recomendar itens aos usuários, com base no aprendizado de seu perfil. O perfil do usuário é obtido das preferências, gostos ou das classificações fornecidas pelo próprio usuário.

Portanto, no processo de construção do perfil de usuário no método proposto, definiu-se por utilizar duas formalizações, sendo uma delas focada em construir um perfil do usuário baseado nos itens que o usuário "gosta" e a outra focada em construir um perfil do usuário baseado nos itens que o usuário "não gosta", denominados perfis **dicotômicos**.

A criação dos dois perfis justifica-se, por trabalhar com a hipótese de recomendar novos itens que sejam mais similares ao perfil composto por itens que o usuário "gosta" e, ao mesmo tempo, também recomendar novos itens que sejam mais dissimilares ao perfil composto por itens que o usuário "não gosta".

Assim como no trabalho realizado por Basu, Hirsh e Cohen (1998), pode-se formalizar o problema de recomendação como um problema de aprendizado, onde foi criada uma função que assume como entrada um filme e um usuário e produz como resultado uma indicação se o filme seria "relevante" ou "não relevante". Sendo assim, também foi proposto um limiar para definir se o item seria relevante para o usuário e utiliza-se essa distinção para a construção do perfil.

A construção dos dois perfis pode ser feita paralelamente, porém suas etapas devem ser realizadas sequencialmente, então após selecionado da nossa base de dados um usuário específico e todos os itens que tiveram avaliações atribuídas por esse usuário, pode-se realizar as seguintes etapas de construção:

- Construção do perfil usuário "gosta" e "não gosta":

    1. Para a construção do perfil "gosta", supondo que se esteja trabalhando com as avaliações dos usuários onde os valores podem ser inteiros de 1 a 5, na qual quanto mais alto o valor, melhor é a aceitação do item pelo usuário, filtra-se do subconjunto de treino os itens com valores de avaliações maiores ou iguais a "4" e de forma contrária para a construção do perfil "não gosta", filtra-se do subconjunto de treino os itens com valores de avaliações menores ou iguais a "2".
    2. Seleciona-se **n** primeiros itens melhor avaliados no caso do perfil "gosta" e realiza-se uma união dos itens, o resultado final será um conjunto composto por representar o perfil do usuário com os itens que o usuário demonstrou maior interesse.

       A construção do perfil "não gosta" se dá de forma análoga, porém seleciona-se os **n** primeiros itens pior avaliados pelo usuário para compor o conjunto.

A Tabela 3.1 exemplifica o método que seleciona os 5 melhores itens para representar o perfil "gosta", ou seja, após realizado todo o pré-processamento para melhorar a representação de cada item, pose-se realizar uma união dos 5 itens melhor avaliados pelo usuário a fim de se ter uma representação do perfil do usuário dos itens que ele "gosta".

Tabela 3.1 – Exemplo união dos top **n** = 5 itens que o usuário 'gosta'

	União top n = 5 itens
Perfil Gosta	$item_1$ + $item_2$ + $item_3$ + $item_4$ + $item_5$

Fonte: Elaborado pelo autor.

3.5 Recomendações

O caráter exploratório deste trabalho consiste em recomendar aos usuários itens que são mais similares ao perfil do que o usuário gosta e, ao mesmo tempo, mais dissimilares ao perfil do que o usuário não gosta, denominados perfis **dicotômicos**, para isso cria-se uma função que visa dar peso ou importância ao perfil que está sendo considerado:

$$S_I = \alpha * PG_I + \beta * (1 - PN_I) \tag{3.1}$$

S_I = *score* final do item

α = significância do perfil "gosta"

PG_I = similaridade do perfil "gosta" com o item

β = significância do perfil "não-gosta"

PN_I = similaridade do perfil "não-gosta" com o item

A Equação (3.1) apresenta o cálculo necessário para realizar a recomendação do item, onde se atribui valores a um α e β utilizados para dar significância ao perfil do que o usuário gosta e do que o usuário não gosta, respectivamente, e sendo a função objetivo maximizar o *score* final do item.

Se é atribuído um valor 0 para α, deixa-se de considerar o perfil gosta do usuário e trabalha-se apenas com a similaridade do item com o perfil não gosta, porém, como o interesse é em recuperar os itens mais dissimilares ao perfil não gosta o ideal é fazer a diferença de 1 por essa similaridade, sendo assim quanto mais alto o valor do *score* mais distante esse item está do perfil não gosta.

Após utilizada a equação e explorados os valores de α e β para encontrar os *scores* de todos os itens do conjunto de testes, pode-se recomendar os top **n** itens que obtiveram *scores* mais altos.

Ao utilizar a base de treinamento para construção do perfil do usuário pode-se calcular a similaridade de cada item na base de teste com os perfis **dicotômicos** do usuário, sendo antes, porém necessária uma etapa de extração e contagem de frequência dos termos do conteúdo textual de cada item no conjunto de teste. As Subseções 2.3.1 e 2.3.2 explicam detalhadamente o processo de extração e contagem de frequência dos termos, e a Subseção 2.3.4 o cálculo de

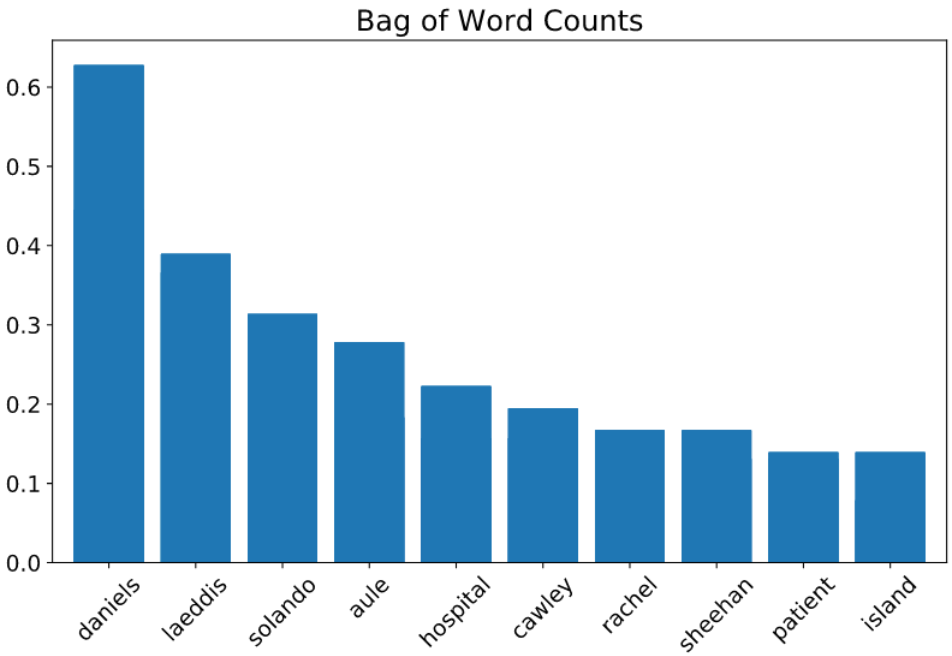
similaridade entre os itens e os dois perfis.

Como neste trabalho a representação do item dá-se de um conteúdo textual que descreve o item, uma estratégia para extração e contagem de termos comumente utilizada é o *tf-idf*, como explicado na Seção 2.3.2, ele é uma medida estatística que reflete a importância de uma palavra para um documento específico em relação a todas as palavras em uma coleção de documentos.

O valor *tf-idf* aumenta proporcionalmente ao número de vezes que a palavra aparece no documento, mas é compensado pela frequência da palavra no conjunto de documentos.

A Figura 3.2 apresenta um gráfico com a importância relativa de algumas palavras no texto de um item em relação ao conjunto de itens.

Figura 3.2 – Importância do termo no conjunto de documentos.



Fonte: Elaborado pelo autor.

O resultado final é uma matriz onde as linhas são representadas pelos itens, as colunas são representadas por cada termo presente na coleção de documentos e cada campo é a importância de cada termo em relação ao conjunto textual de todos os itens.

Agora que se tem um conhecimento da importância de cada termo nos perfis do usuário e em cada item, pode-se calcular a similaridade dos perfis do usuário com todos os itens.

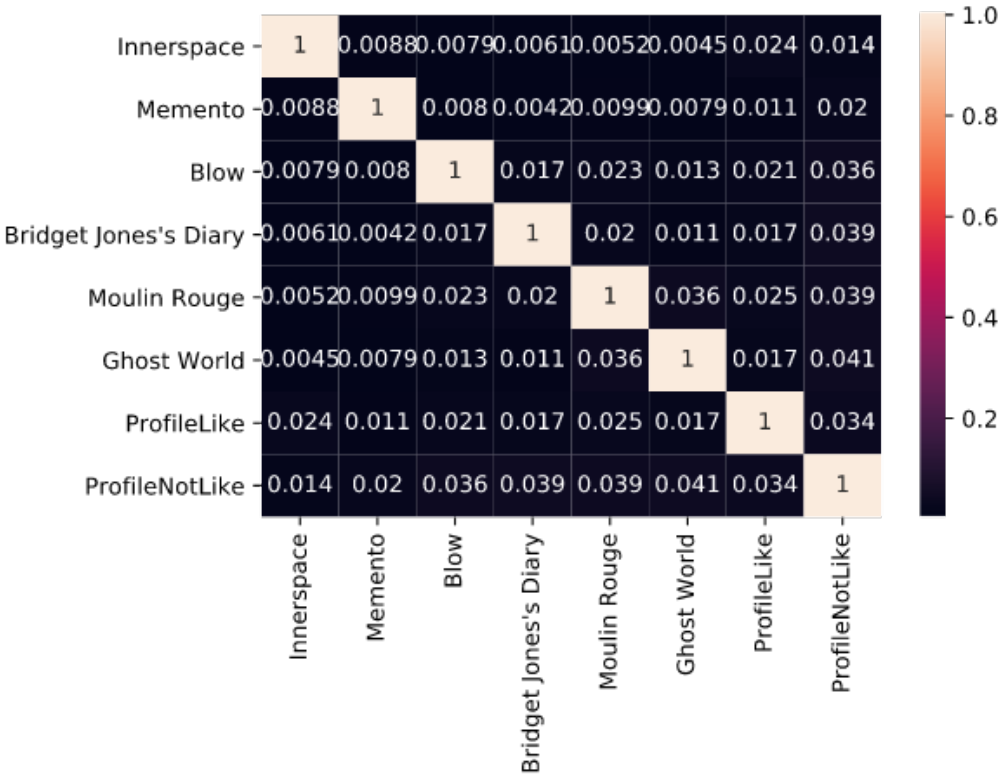
A escolha da métrica de proximidade é essencial para calcular a similaridade entre os itens, com ela pode-se aplicar técnicas como classificação e agrupamento. As medidas de proximidade podem nos dar informações tanto quanto a similaridade ou a dissimilaridade.

A métrica de similaridade mais utilizada no processamento de textos é a similaridade de cosseno, pois ela não considera apenas a magnitude da contagem de termos de cada item, mas a

medida de orientação para a qual o item esta se definindo.

Portanto, optou-se por utilizar essa métrica, sendo calculadas as similaridades dos perfis **dicotômicos** do usuário com todos os itens da base de teste. A Figura 3.3 apresenta um exemplo da similaridade do perfil "gosta" e do perfil "não-gosta" de um determinado usuário com os outros itens também avaliados pelo mesmo usuário.

Figura 3.3 – Cálculo de similaridade do perfil com itens utilizando similaridade de cosseno.



Fonte: Elaborado pelo autor.

Após calculadas as similaridades dos itens com os perfis **dicotômicos**, pode-se agora gerar recomendações, ou seja, sugerir ao usuário alguns itens do conjunto de teste na qual o valor de *score* seja o mais alto.

Podem ser encontradas algumas vantagens e limitações do método proposto quando comparado ao método da Filtragem Colaborativa, que tem sido amplamente utilizado na literatura. Dentre as vantagens identificam-se uma maior possibilidade de recomendar itens que ainda não foram avaliados por nenhum usuário, aumentando assim a diversidade, tem-se também uma transparência dos itens recomendados já que é possível listar explicitamente recursos de conteúdo ou descrições que resultaram naquele item e uma independência do usuário, por ser possível criar um perfil de usuário baseado unicamente em suas preferências. No caso das limitações, observa-se que a análise de conteúdo pode ser limitada, ao ser preciso ter uma descrição rica de conteúdo, não sendo possível distinguir um conteúdo bem descrito de um mal descrito, outro

ponto seria a necessidade de um número considerável de avaliações que devem ser coletadas antes do sistema começar a fornecer recomendações confiáveis aos usuários.

4 Experimentos

Finalizado o desenvolvimento do método proposto, torna-se essencial realizarmos experimentos a fim de verificar o funcionamento e validar as implementações. Este capítulo apresenta os resultados dos experimentos computacionais realizados para demonstrar o uso do método proposto. Na seção 4.1 são apresentadas as bases de dados utilizadas e suas descrições. Em seguida, na Seção 4.2 foram realizadas todas as configurações adequadas dos dados ao modelo proposto e geradas as recomendações. Os resultados são apresentados na Seção 4.3.

4.1 Bases de dados

Assim como apresentado anteriormente, o método proposto aplica-se a domínios na qual o conteúdo do item seja de cunho textual, portanto para se consiga validar o método optou-se por utilizar bases de dados de resumos de filmes, onde é utilizada a 'sinopse' como característica principal de cada item.

Houve-se a necessidade de utilizar duas bases de dados distintas para a criação e validação do método, sendo elas a base de dados *MovieLens* selecionada por fornecer um número expressivo de avaliações de usuários e a base de dados *Wikipedia Movie Plots* que fornece uma descrição longa do item objeto de estudo, nesse caso os filmes.

4.1.1 MovieLens

Na base de dados do *MovieLens* (HERLOCKER, 2000), disponibilizada pelo projeto de pesquisa *GroupLens Research Project* da *University of Minnesota*. Escolhemos um subconjunto composto por:

- 10.681 filmes;
- 71.567 usuários;
- 10.000.054 avaliações;

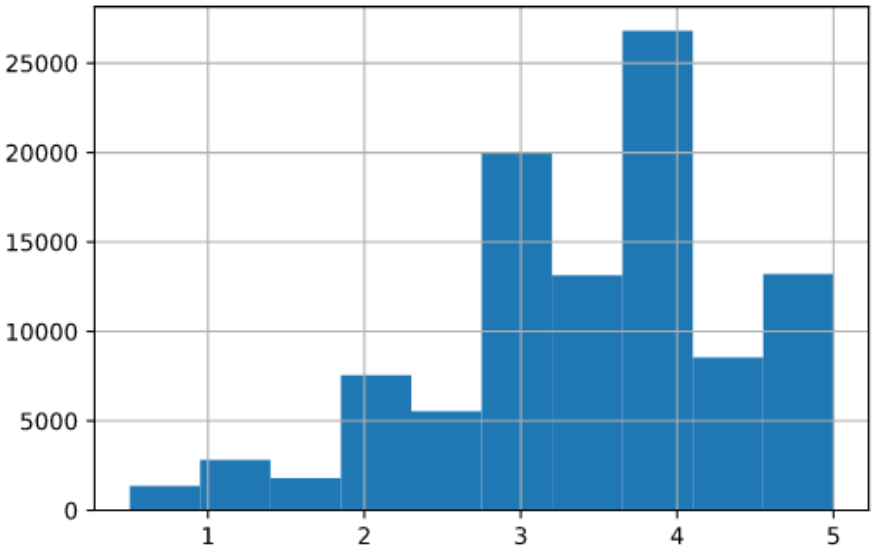
Cada item, no caso filmes, têm 7 características que auxiliam na sua definição.

- *MovieId*: Identificador do filme;
- *Title*: Título do filme;
- *Year*: Ano em que o filme foi lançado;

- *Genre*: Lista de gêneros do filme;
- *UserId*: Identificador do usuário que criou a avaliação;
- *Rating*: Avaliação do usuário;
- *Timestamp*: Momento em que o usuário realizou a avaliação;

As avaliações dos usuários estão em uma escala de “0” a “5”, podendo ter incrementos de “0,5”, sendo que quanto mais alto o valor, melhor é a aceitação do filme pelo usuário, ou seja, a nota “5” representa um filme como “muito bom” e a nota “0” como um filme “muito ruim”.

Figura 4.1 – Histograma dispersão das avaliações dos usuários.



Fonte: Elaborado pelo autor.

4.1.2 Wikipedia Movie Plots

Na base de dados da *Wikipedia Movie Plots*, disponibilizada pelo site *kaggle.com* como fonte de informação para uma competição de sistemas de recomendação. A base é composta pela descrição de 34.886 filmes com origem em todo o mundo.

Cada item, no caso filmes, têm 8 características que auxiliam na sua definição.

- *Release Year*: Ano em que o filme foi lançado;
- *Title*: Título do filme;
- *Origin*: Principal local de filmagem do filme;
- *Director*: Diretor ou diretores do filme;

- *Cast*: Ator ou atores do filme;
- *Genre*: Gêneros do filme;
- *Wiki Page*: URL da página da *Wikipedia*, origem da descrição do filme;
- *Plot*: Descrição longa do filme;

Neste trabalho exploram-se as características de *Genre, Director, Cast e Plot* e para definirmos o item filme utiliza-se a característica *Title*.

A Tabela 4.1 representa o estado inicial de um determinado item presente em nossa base de dados, nesse caso o filme *Toy Story*, após a retirada das colunas *Release Year*, *Origin* e *Wiki Page*, essas colunas não serão utilizadas na construção do nosso método.

Tabela 4.1 – Base de dados com a descrição do filme sem pré-processamento textual.

Title	**Genre**	**Director**	**Cast**	**Plot**
Toy Story	*animated film*	John Lasseter	Tim Allen, Tom Hanks	*In a world where toys are living things who pretend to be lifeless when humans are present, a group of toys, owned by six-year-old Andy Davis, are caught off-guard when Andy's birthday party is moved up a week, as Andy, his mother, and infant sister Molly, are preparing to move the following week. The toys' leader and Andy's favorite toy, a pull-string cowboy doll named Sheriff Woody, organizes the other toys, including Bo Peep the shepherdess, Mr. Potato Head, Rex the Dinosaur, Hamm the Piggy Bank, and Slinky Dog, into a scouting mission. Green army men, led by Sarge, spy on the party, and report the results to the others via baby monitors. The toys are relieved when the party appears to end with none of them having been replaced, but then Andy receives a surprise gift – an electronic toy space ranger action figure named Buzz Lightyear, who thinks he is an actual space ranger.*

Fonte: Elaborado pelo autor.

As descrições longas estão escritas em língua inglesa e a escolha deste conjunto de dados está relacionada a uma boa revisão já realizada nos textos.

4.2 Configuração dos experimentos

O algoritmo foi codificado utilizando a linguagem *Python 3.7.3* e todos os experimentos foram realizados em um computador com processador *Intel Core i7-6500U* de 2.5GHz com 16 GB de RAM, utilizando o sistema operacional *Windows 10*.

4.2.1 Preparação dos dados

O processo de preparação dos dados consiste numa série de tarefas destinadas a obter um conjunto final de dados no qual será criado e validado o método. Portanto, nessa etapa foram realizadas três tarefas de preparação:

1. Foi necessário realizar-se uma junção das duas bases de dados para obter no conjunto final apenas os itens que possuem avaliações dos usuários, sendo assim foram excluídos da base de dados da *Wikipedia Movie Plots* filmes que não possuíam avaliações.

2. Junção das características *Genre*, *Director*, *Cast* e *Plot* em apenas uma característica, sendo essa característica resultante utilizada para definir o item, denominou-se essa característica como *Bag_of_words*.

3. Realizar uma etapa de pré-processamento textual na qual se exclui da *Bag_of_words* os *stopwords* e os termos de pouca relevância, utilizando para isso o (*tf-idf*), a Seção 2.3 explica detalhadamente o funcionamento do processo de remoção utilizando a frequência do termo.

4. Também optou-se por utilizar apenas os usuários que tenham realizado ao menos 50 avaliações de itens, para que após a separação do conjunto de treino e teste, o conjunto de teste não ficasse tão pequeno e ter poucas opções para recomendação.

Sendo assim, após a etapa de preparação dos dados, nossa base de dados ficou composta de 4.936 itens, no caso, filmes e 6.236.611 avaliações dos usuários.

4.2.2 Divisão treino e teste

Neste primeiro momento de experimentação optou-se por utilizar a *hold-out-validation* para a fragmentação do conjunto de dados, detalhada na Subseção 3.3, outras técnicas de particionamento possuem uma complexidade maior de implementação e dificultar o processo de experimentação.

Sendo assim, os parâmetros utilizados em nossa fragmentação foram 70% para o conjunto de treinamento e 30% para teste. Ou seja, o usuário cujo Id = 69827 realizou 158 avaliações, seu conjunto de treinamento ficou com a quantidade de 111 itens e o conjunto de teste com 47 itens.

Aos itens presentes no conjunto de teste é realizado um procedimento de classificação de relevância do item, para que posteriormente, caso o item esteja presente na lista de recomendação essa classificação seja utilizada nas métricas de avaliação.

Para tal classificação utilizou-se como parâmetro o **rating** realizado pelo usuário para o item específico e como limiar o valor "3,5", ou seja, os itens previamente avaliados pelo usuário com valor menor que "3,5" foram classificados como "não relevantes" e os itens com valor maior ou igual a "3,5" foram classificados como "relevantes".

4.2.3 Perfis dicotômicos de usuário

Segue-se então que após as etapas de pré-processamento e divisão treino e teste é utilizado o conjunto de treinamento para criar o perfil dicotômico de cada usuário.

Aos itens resultantes realiza-se uma ordenação pela data que o usuário realizou a avaliação *timestamp*, assim pode-se construir um perfil do usuário utilizando avaliações efetuadas recentemente e obter um perfil do usuário mais atualizado, pois, como apresentado no trabalho de Satler et al. (2010), foi observado que os gostos e preferências dos usuários são modificados constantemente devido ao enorme número de novos conteúdos a que são submetidos.

Após a etapa de ordenação dos itens, tem-se também como parâmetro de entrada do método a quantidade de itens que será utilizada para a criação dos perfis dicotômicos, nos cenários de experimentação foram utilizados respectivamente os valores: (1) 1 e 15; (2) 5 e 15; (3) 15 e 15; (4) 15 e 5; (5) 15 e 1, ou seja, no primeiro cenário de experimentação é utilizado apenas 1 item com *rating* igual ou superior a 4, para a criação do perfil itens que o usuário "gosta" e 15 itens com *rating* igual ou inferior a 2 para a criação do perfil itens que o usuário "não gosta".

4.2.4 Recomendações

Tendo-se realizado as etapas de pré-processamento, divisão treino e teste, e criação dos perfis dicotômicos dos usuários, pode-se então realizar as recomendações considerando a função de importância dos perfis, conforme a Equação 3.1 e variando os valores de α e β de 0% a 100%.

O algoritmo seguinte apresenta uma configuração dos valores de entrada para a realização dos experimento e as recomendações sendo geradas para os seguintes valores de α e β, respectivamente: (1) 0% e 100%; (2) 25% e 75%; (3) 50% e 50%; (4) 75% e 25%; (5) 100% e 0%. Pode-se dizer que a primeira configuração, com valores "0%" e "100%, é o nosso principal *baseline*, uma vez que caracteriza os modelos comumente utilizados na literatura, com o perfil "gosta". Do lado oposto, a última configuração, com valores 100% e 0%, também será considerada *baseline*, por também utilizar apenas um dos perfis, no caso, o "não-gosta".

Temos então que cada cenário de experimentação é realizada a recomendação de uma lista de itens ao usuário, utilizando de uma configuração específica dos parâmetros de entrada e de uma porcentagem definida dos valores de α e β.

Algoritmo 1 Recommend

```
from sklearn.model_selection import train_test_split
alfa ← [1, 0.75, 0.5, 0.25, 0]
beta ← [0, 0.25, 0.5, 0.75, 1]
Input: alfa, beta, df_movies, df_ratings, users, threshold_like, threshold_not_like
Output: Lista de recomendação: df_recommendations
movies ← movies_user_rated(df_movies, df_ratings, users)
train, test ← train_test_split(user_movie_ratings, test_size = 0.3, shuffle=True)
profiles ← learning_profiles(train, threshold_like, threshold_not_like)
df_recommendations ← recommender(test, user_profile, alfa, beta, user)
return df_recommendations
```

4.3 Resultados

Para realizar os experimentos e gerar cada lista de recomendação foram feitas algumas variações na quantidade de itens utilizados para a criação dos **perfis dicotômicos**. O objetivo seria verificar quais cenários de entrada poderiam obter resultados relevantes quanto ao conteúdo textual dos itens e obter informações sobre como a construção dos perfis do usuário poderia inferir nas recomendações, visto que uma má escolha destas configurações poderia inferir na qualidade do método.

Conforme a Tabela 4.2, temos os cenários utilizados para a criação dos perfis do usuário e a quantidade de usuários obtidos após realizada a filtragem, considerando a quantidade miníma de *rating* em itens que o usuário "gosta" e quantidade miníma de *rating* em itens que o usuário "não-gosta".

Tabela 4.2 – Filtragem quantidade de *ratings* para criar perfis dicotômicos.

Min. itens perfil "gosta"	Min. itens perfil "não-gosta"	Qtd total usuários
15	1	29.189
15	5	23.233
15	15	12.755
5	15	13.174
1	15	13.210

Fonte: Elaborado pelo autor.

Em todos os experimentos, foram considerados os *Top-5* e *Top-20* itens da lista de recomendação de cada usuário para a avaliação. As métricas de avaliação utilizadas foram EPC e EILD, conforme Subseções 2.4.4 e 2.4.5, para considerar a novidade e NDCG, Subseção 2.4.3, para precisão. Em seguida, calcula-se a média das métricas obtidas para cada configuração em cada cenário de teste, o que nos oferece uma visão geral do desempenho. Quanto à confiabilidade das métricas, foi utilizado o Intervalo de Confiança, que nos fornece uma estimativa da variação e da consistência das médias calculadas, permitindo avaliar empates estatísticos entre as diferentes

configurações. A fórmula do intervalo de confiança utiliza como parâmetro o nível de confiança, neste trabalho utiliza-se o valor de 0,95, o que permite expressar com 95% de confiança que a média dos valores das métricas estão em uma determinada faixa.

A Tabela 4.3 mostra os resultados obtidos considerando a média e o Intervalo de Confiança das métricas, esses resultados nos indica que a porcentagem de α = 50% e β = 50% obtém os melhores valores para todas as métricas avaliadas, EPC, EILD e NDCG.

Tabela 4.3 – Média e Intervalo de Confiança das métricas. Maiores valores de médias estão destacadas em negrito (para todas as métricas, quanto maior, melhor).

	NDCG@5	**NDCG@20**	**EPC@5**	**EPC@20**	**EILD@5**	**EILD@20**
Cenário de teste: 1_15						
100% / 0%	0,704 ± 0,005	0,727 ± 0,003	0,423 ± 0,004	0,404 ± 0,003	0,213 ± 0,003	0,324 ± 0,003
75% / 25%	0,717 ± 0,005	0,736 ± 0,003	0,436 ± 0,004	0,414 ± 0,003	0,224 ± 0,003	0,336 ± 0,003
50% / 50%	**0,724** ± 0,005	**0,743** ± 0,003	**0,448** ± 0,004	**0,425** ± 0,003	**0,236** ± 0,003	**0,350** ± 0,003
25% / 75%	0,694 ± 0,005	0,723 ± 0,003	0,420 ± 0,004	0,405 ± 0,003	0,214 ± 0,003	0,333 ± 0,003
0% / 100%	0,649 ± 0,005	0,697 ± 0,003	0,381 ± 0,004	0,386 ± 0,003	0,186 ± 0,003	0,316 ± 0,003
Cenário de teste: 5_15						
100% / 0%	0,707 ± 0,005	0,729 ± 0,002	0,422 ± 0,004	0,405 ± 0,003	0,211 ± 0,003	0,324 ± 0,002
75% / 25%	0,718 ± 0,004	0,737 ± 0,003	0,435 ± 0,004	0,414 ± 0,003	0,222 ± 0,003	0,336 ± 0,002
50% / 50%	**0,723** ± 0,004	**0,744** ± 0,003	**0,446** ± 0,004	**0,424** ± 0,003	**0,234** ± 0,003	**0,349** ± 0,003
25% / 75%	0,694 ± 0,005	0,724 ± 0,003	0,419 ± 0,004	0,406 ± 0,003	0,213 ± 0,003	0,333 ± 0,003
0% / 100%	0,654 ± 0,005	0,700 ± 0,003	0,384 ± 0,004	0,387 ± 0,003	0,187 ± 0,003	0,317 ± 0,003
Cenário de teste: 15_15						
100% / 0%	0,715 ± 0,004	0,735 ± 0,003	0,427 ± 0,004	0,411 ± 0,003	0,214 ± 0,003	0,331 ± 0,003
75% / 25%	0,727 ± 0,005	0,744 ± 0,003	0,441 ± 0,004	0,421 ± 0,003	0,227 ± 0,003	0,343 ± 0,003
50% / 50%	**0,737** ± 0,004	**0,752** ± 0,003	**0,454** ± 0,004	**0,433** ± 0,003	**0,239** ± 0,003	**0,357** ± 0,003
25% / 75%	0,709 ± 0,005	0,733 ± 0,003	0,430 ± 0,004	0,415 ± 0,003	0,221 ± 0,003	0,341 ± 0,003
0% / 100%	0,668 ± 0,005	0,708 ± 0,003	0,394 ± 0,004	0,395 ± 0,003	0,194 ± 0,003	0,324 ± 0,003
Cenário de teste: 15_5						
100% / 0%	0,751 ± 0,003	0,768 ± 0,002	0,484 ± 0,004	0,466 ± 0,003	0,255 ± 0,003	0,378 ± 0,002
75% / 25%	0,763 ± 0,003	0,776 ± 0,001	0,496 ± 0,004	0,473 ± 0,003	0,266 ± 0,003	0,388 ± 0,002
50% / 50%	**0,770** ± 0,003	**0,781** ± 0,002	**0,506** ± 0,004	**0,482** ± 0,002	**0,277** ± 0,003	**0,399** ± 0,002
25% / 75%	0,749 ± 0,003	0,767 ± 0,002	0,486 ± 0,003	0,469 ± 0,002	0,263 ± 0,003	0,390 ± 0,002
0% / 100%	0,713 ± 0,003	0,746 ± 0,002	0,450 ± 0,003	0,451 ± 0,002	0,235 ± 0,003	0,374 ± 0,002
Cenário de teste: 15_1						
100% / 0%	0,769 ± 0,003	0,784 ± 0,002	0,513 ± 0,003	0,497 ± 0,002	0,278 ± 0,003	0,405 ± 0,002
75% / 25%	0,778 ± 0,003	0,790 ± 0,002	0,524 ± 0,003	0,502 ± 0,002	0,288 ± 0,003	0,414 ± 0,002
50% / 50%	**0,784** ± 0,003	**0,795** ± 0,002	**0,532** ± 0,003	**0,510** ± 0,002	**0,297** ± 0,003	**0,425** ± 0,003
25% / 75%	0,771 ± 0,003	0,786 ± 0,002	0,517 ± 0,003	0,499 ± 0,002	0,286 ± 0,003	0,416 ± 0,002
0% / 100%	0,738 ± 0,003	0,766 ± 0,002	0,482 ± 0,003	0,483 ± 0,002	0,260 ± 0,003	0,402 ± 0,002

Fonte: Elaborado pelo autor.

Obtidos os Intervalos de Confiança foi utilizado o Ranqueamento Fracionado, descrito na Seção 2.4.6, para classificar os resultados de cada métrica e identificar quais valores de α e β obtiveram melhor ranqueamento com significância estatística. Um ranqueamento geral foi definido pela soma de cada ranqueamento individual. O Ranqueamento Fracionado foi escolhido por diferenciar os empates estatísticos de uma forma a gerar um ranqueamento geral potencialmente mais justo.

A Tabela 4.4, demonstra o resultado obtido, apresentando o ranqueamento das configurações e variando o valor α e β para todas as métricas.

Os resultados dos ranqueamentos mostram que a porcentagem de α = 50% e β = 50% tem a melhor posição em todos os cenários de teste dos experimentos. Por outro lado, as configurações

Tabela 4.4 – Resultados do ranqueamento das configurações para todas as métricas. As configurações de cada cenário de teste foram ordenadas pelo ranqueamento geral.

α / β	NDCG@5	NDCG@20	EPC@5	EPC@20	EILD@5	EILD@20	Geral
Cenário de teste: 1_15							
50% / 50%	**1,5**	**1,0**	**1,0**	**1,0**	**1,0**	**1,0**	**6,5**
75% / 25%	**1,5**	2,0	2,0	2,0	2,0	2,5	**12,0**
25% / 75%	4,0	3,5	3,5	3,5	3,5	2,5	**20,5**
100% / 0%	3,0	3,5	3,5	3,5	3,5	4,0	**21,0**
0% / 100%	5,0	5,0	5,0	5,0	5,0	5,0	**30,0**
Cenário de teste: 5_15							
50% / 50%	**1,5**	**1,0**	**1,0**	**1,0**	**1,0**	**1,0**	**6,5**
75% / 25%	**1,5**	2,0	2,0	2,0	2,0	2,5	**12,0**
25% / 75%	4,0	3,5	3,5	3,5	3,5	2,5	**20,5**
100% / 0%	3,0	3,5	3,5	3,5	3,5	4,0	**21,0**
0% / 100%	5,0	5,0	5,0	5,0	5,0	5,0	**30,0**
Cenário de teste: 15_15							
50% / 50%	**1,0**	**1,0**	**1,0**	**1,0**	**1,0**	**1,0**	**6,0**
75% / 25%	2,0	2,0	2,0	2,0	3,0	2,5	**13,5**
25% / 75%	3,5	3,5	3,5	3,5	3,0	2,5	**19,5**
100% / 0%	3,5	3,5	3,5	3,5	3,0	4,0	**21,0**
0% / 100%	5,0	5,0	5,0	5,0	5,0	5,0	**30,0**
Cenário de teste: 15_5							
50% / 50%	**1,0**	**1,0**	**1,0**	**1,0**	**1,0**	**1,0**	**6,0**
75% / 25%	2,0	2,0	2,0	3,0	2,5	2,5	**14,0**
25% / 75%	3,5	3,5	3,5	3,0	2,5	2,5	**18,5**
100% / 0%	3,5	3,5	3,5	3,0	4,0	4,0	**21,5**
0% / 100%	5,0	5,0	5,0	5,0	5,0	5,0	**30,0**
Cenário de teste: 15_1							
50% / 50%	**1,0**	**1,0**	**1,0**	**1,0**	**1,0**	**1,0**	**6,0**
75% / 25%	2,0	2,0	2,0	3,0	2,5	2,5	**14,0**
25% / 75%	3,5	3,5	3,5	3,0	2,5	2,5	**18,5**
100% / 0%	3,5	3,5	3,5	3,0	4,0	4,5	**22,0**
0% / 100%	5,0	5,0	5,0	5,0	5,0	4,5	**29,5**

Fonte: Elaborado pelo autor.

que utilizam apenas um dos perfis (“gosta” ou “não-gosta”), figuram sempre nas duas últimas posições do ranqueamento geral. Demonstrando que, de fato, a utilização dos perfis dicotômicos contribui para melhores resultados na Filtragem Baseada em Conteúdo.

5 Considerações Finais

Este capítulo consolida os principais resultados e conhecimentos adquiridos ao longo deste trabalho. A seguir, na Seção 5.1 serão apresentadas as conclusões obtidas através da experimentação realizada utilizando o método proposto, enfatizando os resultados de sucesso e os casos que não foram tão eficientes conforme o domínio aplicado. Em seguida, na Seção 5.2 foram destacadas as sugestões de trabalhos futuros, indicando possíveis direções de melhorias e realizações de demais testes para a comprovação do método proposto.

5.1 Conclusão

Este trabalho surge da ideia inicial de explorarmos Sistemas de Recomendação que utilizem tanto das avaliações positivas quanto as avaliações negativas realizadas pelo usuário perante os itens acessados ou consumidos. Inicialmente foi feito um estudo das técnicas de Sistemas de Recomendação utilizadas no cotidiano, discutiram-se suas similaridades e limitações. Após apresentadas as diferentes técnicas de sistema de recomendação, optou-se por utilizar a técnica de Recomendação Baseada em Conteúdo, conforme justificada anteriormente, para a exploração das avaliações positivas e negativas previamente realizadas pelo usuário, no qual denominou-se **perfis dicotômicos**.

No decorrer deste trabalho, atingimos com sucesso os objetivos propostos, que visavam explorar as técnicas utilizadas na Recomendação Baseada em Conteúdo, implementar uma técnica comumente utilizada, sugerir adaptações ao método implementado, de forma a considerar a criação de perfis de usuário que já realizaram avaliações positivas e negativas dos itens.

Os resultados experimentais obtidos neste trabalho evidenciam que a utilização das avaliações negativas realizadas pelo usuário podem trazer muito conhecimento e melhor a eficácia na recomendação de novos itens ao usuário, o método proposto obteve resultados interessantes quando foram considerados os valores de α e β em 50%/50%, ou seja, realizar a recomendação de itens que estejam ao mesmo tempo, mais próximos dos itens que o usuário gosta e mais distante ao que o usuário não gosta, podem trazer melhorias nos resultados de recomendações quando considerados os critérios de *novidade*, *diversidade* e *precisão* das listas de itens recomendados.

5.2 Trabalhos Futuros

Com os resultados obtidos pode-se ficar entusiasmados em continuar com os estudos e realizar mais adaptações no método, são citados a seguir os pontos de melhorias identificados que podem trazer significativamente uma maior validação e utilização do método.

Em primeiro lugar é possível aplicar o método em mais domínios que propiciem a utilização de bases de dados de cunho textual, algumas sugestões seriam aplicar na recomendação de artigos científicos, notícias, receitas culinárias e descrição de produtos, por exemplo.

Melhorar os algoritmos que realizam a construção dos **perfis dicotômicos**, considerando a utilizando de pesos para as características que melhor representam o item. Utilizar a técnica de **embedding** de palavras para obter uma representação mais semântica do conteúdo textual.Considerar a utilização dos **perfis dicotômicos** nas demais técnicas de sistema de recomendação.

Outra sugestão seria utilizar de diferentes adaptações e diferentes algoritmos para realizar o cálculo de similaridade entre os itens, podendo também utilizar de diferentes técnicas de divisão dos dados em treino e teste.

Além disso, pode-se explorar também a utilização de diferentes técnicas meta-heurísticas a fim de otimizar a função objetivo e obter as melhores porcentagens de α e β na função que define o *score* do item.

Referências

ADOMAVICIUS, G.; TUZHILIN, A. Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering*, v. 17, n. 6, p. 734–749, June 2005. ISSN 1041-4347.

BASU, C.; HIRSH, H.; COHEN, W. Recommendation as classification: Using social and content-based information in recommendation. *AAAI Technical Report*, 1998.

BRIDGE, D.; GöKER, M. H.; MCGINTY, L.; SMYTH, B. Case-based recommender systems. *Knowl. Eng. Rev.*, Cambridge University Press, New York, NY, USA, v. 20, n. 3, p. 315–320, set. 2005. ISSN 0269-8889. Disponível em: <http://dx.doi.org/10.1017/S0269888906000567>.

BURKE, R. *Hybrid recommender systems: Survey and experiments. User Modeling and User-Adapted Interaction*. [S.l.: s.n.], 2002.

BUSATTO, C. O que tá valendo? um sistema web de recomendação de eventos. *Trabalho de Conclusão de Curso (Graduação em Ciência da Computação) - Universidade Federal do Rio Grande do Sul*, Porto Alegre, RS, 2013.

CABENA, P.; HADJINIAN, P.; STADLER, R.; VERHEES, J.; ZANASI, A. Discovering data mining: From concept to implementation. *Prentice Hall*, 1997.

CARTERETTE, B.; BENNETT, P. N. Evaluation measures for preference judgments. *ACM, New York, NY, USA*, 2008.

CAZELLA, S. C.; NUNES, M. A. S.; REATEGUI, E. B. A. Ciência da opinião: Estado da arte em sistemas de recomendação. *In: XXX CONGRESSO DA SOCIEDADE BRASILEIRA DE COMPUTAÇÃO - JORNADA DE ATUALIZAÇÃO EM INFORMÁTICA (JAI)*, Belo Horizonte, MG, 2010.

CAZELLA, S. C.; REATEGUI, E. B. A. Sistemas de recomendação. *In: XXV CONGRESSO DA SOCIEDADE BRASILEIRA DE COMPUTAÇÃO - A UNIVERSALIDADE DA COMPUTAÇÃO: UM AGENTE DE INOVAÇÃO E CONHECIMENTO*, São Leopoldo, RS, 2005.

D ZANKER M, F. A. J.; G, F. *Recommender Systems: An Introduction*. [S.l.]: Cambridge University Press, 2010.

FELDMAN, R.; SANGER, J. *The Text Mining Handbook: Advanced Aproaches in Analysing Unstructured Data*. [S.l.]: Cambridge University Press, 2006.

HERLOCKER, J. L. Understanding and improving automated collaborative filtering systems. *Tese de Doutorado (Doutorado em Ciência da Computação), University of Minnesota*, Minnesota, 2000.

JäRVELIN, K.; KEKäLäINEN, J. *Cumulated gain-based evaluation of ir techniques.* [S.l.: s.n.], 2002. v. 20(4):. 422–446 p.

KOREN, Y. *Factorization meets the neighborhood: a multifaceted collaborative filtering model.* [S.l.]: Proceeding of the 14h ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2008.

LEWIS, D. *Information Overload. Practical Strategies for Surviving in Today's Workplace*. [S.l.: s.n.], 1999. 30 p.

LOPS, P.; GEMMIS, M. de; SEMERARO, G. *Content-based Recommender Systems: State of the Art and Trends*. [S.l.: s.n.], 2011.

MA, H.; XIE, R.; MENG, L.; FENG, F.; DU, X.; SUN, X.; KANG, Z.; MENG, X. Negative sampling in recommendation: A survey and future directions. *ACM, New York, NY, USA*, 2018.

MANNIG, C. D.; RAGHAVAN, P.; SHUTZE, H. *An Introduction to Information Retrieval*. [S.l.]: Cambridge University Press, 2008.

MAYER-SCHöNBERGER, V.; CUKIER, K. *Big Data: A Revolution that Will Transform how We Live, Work, and Think*. [S.l.]: Houghton Mifflin Harcourt, 2013.

MELVILLE, P.; SINDHWANI, V. *Recommender Systems*. [S.l.: s.n.], 2017. 1056-1066 p.

MIKOLOV, T.; CORRADO, G.; CHEN, K.; DEAN, J. Efficient estimation of word representations in vector space. *International Conference on Learning Representations*, 2013.

PAZZANI, M. J.; BILLSUS, D. *Content-based recommendation systems. In THE ADAPTIVE WEB: METHODS AND STRATEGIES OF WEB PERSONALIZATION*. [S.l.]: Springer-Verlag, 2007. 325-341 p.

RICCI, F.; ROKACH, L.; SHAPIRA, B. *Introduction to Recommender Systems Handbook. In:Recommender Systems Handbook*. 2. ed.. ed. Boston, MA: [s.n.], 2011.

SALTON, G. *Automatic Text Processing: The Transformation, Analysis, and Retrieval of. Addison-Wesley*. [S.l.: s.n.], 1989. ISBN 0201122278.

SATLER, M. F.; ROMERO, F. P.; MENéNDEZ, V. H.; ZAPATA, A.; PIETRO, M. E. A fuzzy ontology approach to represent user profiles in e-learning enviroments. *IEEE Computer Society*, 2010.

SCHAFER, J. B. *Collaborative filtering recommender systems. In: (Ed.). The adaptive web*. Springer: [s.n.], 2007. ISBN 3540720782.

SCHAFER, J. B.; FRANKOWSKI, J. H. D.; SEN, S. *Collaborative filtering recommender systems. In P. Brusilovsky, A. Kobsa, and W. Nejdl, editors, The Adaptive Web: Methods and Strategies of Web Personalization*. [S.l.]: Springer-Verlag, 2006, 2009.

SHANI, G.; GUNAWARDANA, A. *Evaluating remmendation systems*. [S.l.]: In: Ricci, F.; Rokach, L.; Shapira, B; Kantor, P. B., eds. Recommender Systems Handbook, Springer US, 2011.

STERN, D. H.; HERBRICH, R.; GRAEPEL, T. *Matchbox: large scale online bayesian recommendations*. [S.l.]: In Proceedings of the 18th international conference on World wide web; New York, NY, USA, 2009. ACM, 2009.

TAKAHASHI, M. M. Estudo comparativo de algoritmos de recomendação. *Trabalho de Conclusão de Curso (Graduação em Ciência da Computação) - Universidade de São Paulo*, São Paulo, SP, 2015.

TOLKIEN, J. *The Fellowship of the Ring*. [S.l.]: George Allen & Unwin, 1954.

VARGAS, S.; CASTELLS, P. Rank and relevance in novelty and diversity metrics for recommender systems. *RecSys 2011*, 2011.

WANG, Y.; BANERJEE, C.; CHUCRI, S.; SOLDO, F.; BADAM, S.; CHI, E. H.; CHEN, M. Diversifying by intent in recommender systems. *arXiv preprint arXiv:2405.12327*, 2024.

WANG, Y.; HALPERN, Y.; CHANG, S.; FENG, J.; LE, E. Y.; LI, L.; LIANG, X.; HUANG, M.-C.; LI, S.; BEUTEL, A.; ZHANG, Y.; BI, S. Learning from negative user feedback and measuring responsiveness for sequential recommenders. Association for Computing Machinery, New York, NY, USA, 2023.

ZIGKOLIS, C.; KARAGIANNIDIS, S.; VAKALI, A. Dissimilarity features in recommender systems. *IEEE 13th International Conference on Data Mining Workshops.*, 2013.