

**RECOMENDAÇÃO ASSOCIATIVA DE TAGS NA
AUSÊNCIA DE INFORMAÇÃO PRÉVIA**

EDER FERREIRA MARTINS

RECOMENDAÇÃO ASSOCIATIVA DE TAGS NA
AUSÊNCIA DE INFORMAÇÃO PRÉVIA

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação do Instituto de Ciências Exatas da Universidade Federal de Minas Gerais como requisito parcial para a obtenção do grau de Mestre em Ciência da Computação.

ORIENTADOR: JUSSARA MARQUES ALMEIDA
COORIENTADOR: MARCOS ANDRÉ GONÇALVES

Belo Horizonte

Abril de 2013

© 2013, Eder Ferreira Martins.
Todos os direitos reservados.

M386r Martins, Eder Ferreira
Recomendação associativa de tags na ausência de
informação prévia / Eder Ferreira Martins. — Belo
Horizonte, 2013
xviii, 84 f. : il. ; 29cm

Dissertação (mestrado) — Universidade Federal de
Minas Gerais
Orientador: Jussara Marques Almeida
Coorientador: Marcos André Gonçalves

1. Computação - Teses. 2. Recuperação da
Informação. 3. Sistemas de recomendação.
4. Ferramentas de busca na Web.
I. Orientador II. Coorientador III. Título.

CDU 519.6*73(043)



UNIVERSIDADE FEDERAL DE MINAS GERAIS
INSTITUTO DE CIÊNCIAS EXATAS
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

FOLHA DE APROVAÇÃO

Recomendação associativa de tags na ausência de informação prévia

EDER FERREIRA MARTINS

Dissertação defendida e aprovada pela banca examinadora constituída pelos Senhores:

Jussara M. Almeida Gonçalves

PROFA. JUSSARA MARQUES DE ALMEIDA GONÇALVES - Orientadora
Departamento de Ciência da Computação - UFMG

Marcos André Gonçalves

PROF. MARCOS ANDRÉ GONÇALVES - Coorientador
Departamento de Ciência da Computação - UFMG

Nívio Ziviani

PROF. NÍVIO ZIVIANI
Departamento de Ciência da Computação - UFMG

Pável Pereira Calado

PROF. PÁVEL PEREIRA CALADO
Instituto Superior Técnico - Portugal

Ricardo Torres

PROF. RICARDO DA SILVA TORRES
Instituto de Computação - UNICAMP

Belo Horizonte, 19 de abril de 2013.

“Pensar é o trabalho mais difícil que existe. Talvez por isso tão poucos se dediquem a ele.”

(Henry Ford)

Resumo

A literatura contém uma grande variedade de estratégias de recomendação de *tags*, sendo que as que exploram padrões de coocorrência com *tags* previamente associadas aos objetos do sistema (incluindo *tags* disponíveis no próprio objeto alvo da recomendação), chamados de métodos associativos, têm produzido consistentemente resultados estado da arte. Entretanto, *tags* estão ausentes em uma parcela significativa dos objetos de várias aplicações populares da Web 2.0, o que levanta uma preocupação a respeito da eficácia de métodos associativos frente a sua ausência. Esse é um problema bem conhecido em sistemas de recomendação, chamado *cold start*, e é o foco desta dissertação. Mais especificamente, aborda-se o problema de recomendar *tags* para objetos sem *tags* previamente associadas. Tal estudo é iniciado pela avaliação do impacto do *cold start* sobre uma família de métodos associativos, usando para isso bases de dados coletadas de várias aplicações populares da Web 2.0. Os resultados indicam que a eficácia de tais métodos é muito prejudicada quando eles não podem utilizar *tags* previamente associadas ao objeto alvo da recomendação. É mostrado também que o uso de estratégias automáticas de filtragem de termos para gerar um conjunto inicial de *tags*, a partir do qual padrões de coocorrência podem ser obtidos, produz ganhos limitados. É proposto um novo método de recomendação, mais robusto em presença do *cold start*, que explora tanto o *feedback* positivo quanto o negativo, provido pelo usuário, para iterativamente selecionar *tags* para os métodos associativos. Os resultados, comparando-se algumas variações da estratégia proposta, baseada em *relevance feedback*, com os métodos originais baseados em coocorrência, mostram que estender os métodos associativos para incluir o *relevance feedback* gera ganhos significativos sobre o melhor método de referência considerado. É mostrado também que a estratégia proposta é eficaz mesmo na ausência de cooperação e de confiabilidade por parte do usuário, permanecendo tão boa, se não superior, ao melhor método de referência, mesmo que o *feedback* provido pelo usuário contenha ruído ou que ele selecione apenas uma parte das *tags* relevantes que são mostradas no processo de *relevance feedback*.

Palavras-chave: Recomendação de Tags, Ausência de Informação Prévia, Relevance Feedback.

Abstract

Among the existing tag recommendation strategies, those that have consistently produced state-of-the-art performance often exploit term co-occurrence patterns with tags previously assigned to the target object, possibly jointly with other textual features. However, these techniques have been evaluated only for objects with previously assigned tags. Instead, we here focus on the recommendation of tags for objects with empty tags, a variation of a well-known problem called *cold start*. We start by evaluating state-of-the-art co-occurrence based methods in face of *cold start*, using datasets collected from for popular Web 2.0 applications. Our results indicate that the effectiveness of these methods suffer greatly when they cannot rely on previously assigned tags in the target object. Moreover, we show that employing various automatic filtering strategies to generate a initial set of tags that enables the use of co-occurrence patterns produces limited gains. We then propose a new approach that exploits both positive and negative relevance feedback (RF) from users to iteratively select input tags to these methods. Our experimental results, comparing several variations of the proposed RF strategy against the original co-occurrence based methods and other state-of-the-art baselines in various scenarios, indicate that extending the methods to include relevance feedback from users leads to significant gains in precision over the best considered baseline in *cold start* scenarios, and also on objects that contain some initial tags (i.e., no *cold start*). We also show that our best RF-driven strategy is robust to the lack of user reliability and cooperation, remaining at least as good if not superior to the best baseline even if the feedback provided by user contains noise or if users select only a portion of the relevant tags that are shown to them in the feedback process.

Keywords: Tag recommendation, Cold start, Relevance feedback.

Lista de Figuras

2.1	Página de um objeto do Last.FM.	10
3.1	Algoritmo CTTR (Adaptado) [Lipczak et al., 2009].	28
3.2	Processo evolucionário do algoritmo de programação genética [Belém et al., 2011].	32
3.3	Exemplos de árvores.	34
4.1	Estratégia baseada em RF.	38
4.2	Variações da estratégia proposta.	40
4.3	<i>PG + RF</i> : treino x execução.	45
5.1	Convergência dos métodos baseados em RF.	58
5.2	Impácto do <i>feedback</i> negativo sobre a eficácia das recomendações.	59
5.3	Sensibilidade dos métodos baseados em RF à falta de cooperação por parte do usuário.	63
5.4	Sensibilidade dos métodos baseados em RF à introdução de ruído.	64

Lista de Tabelas

4.1	Métricas heurísticas utilizadas para a filtragem automática de termos de atributos textuais do objeto alvo da recomendação.	37
5.1	Descrição geral das bases de dados.	48
5.2	Valores médios de P@5 para o Sum^+wTS e $LATRE^+wTS$ em função da variação dos parâmetros θ_{min} e σ_{min} . Em negrito resultados com melhor compromisso entre eficiência e precisão.	51
5.3	P@5 média para o Sum^+wTS e $LATRE^+wTS$ em função da variação do parâmetro α . Em negrito melhores resultados.	51
5.4	Parametrização dos Algoritmos de Recomendação de <i>Tags</i>	52
5.5	Importância das diferentes fontes de informação para a eficácia dos métodos associativos Sum^+wTS e $LATRE^+wTS$. Valores médios de P@5 juntamente com intervalos de confiança de 95%.	53
5.6	Estratégias de filtragem automáticas para construir \mathcal{I}'_o : P@5 média juntamente com intervalos de confiança de 95%.	56
5.7	Estratégias de recomendação de <i>tags</i> baseadas em <i>Relevance feedback</i> aplicadas em objetos em <i>cold start</i> : Valores médios de P@5 juntamente com intervalos de confiança de 95%.	61
5.8	Estratégias de recomendação de <i>tags</i> baseadas em <i>Relevance feedback</i> aplicadas em objetos com <i>tags</i> : Valores médios de P@5 juntamente com intervalos de confiança de 95%.	62
5.9	Exemplos de recomendações no YouTube.	66
A.1	Importância das diferentes fontes de informação para a eficácia dos métodos associativos Sum^+wTS e $LATRE^+wTS$. Valores médios de revocação juntamente com intervalos de confiança de 95%.	81

A.2	Importância das diferentes fontes de informação para a eficácia dos métodos associativos Sum^+wTS e $LATRE^+wTS$. Valores médios de AP juntamente com intervalos de confiança de 95%.	82
A.3	Estratégias de filtragem automáticas para construir \mathcal{I}'_o : revocação média juntamente com intervalos de confiança de 95%.	82
A.4	Estratégias de filtragem automáticas para construir \mathcal{I}'_o : AP média juntamente com intervalos de confiança de 95%.	83
A.5	Estratégias de recomendação de <i>tags</i> baseadas em <i>Relevance feedback</i> aplicadas em objetos em <i>cold start</i> : Valores médios de revocação juntamente com intervalos de confiança de 95%.	84
A.6	Estratégias de recomendação de <i>tags</i> baseadas em <i>Relevance feedback</i> aplicadas em objetos em <i>cold start</i> : Valores médios de AP juntamente com intervalos de confiança de 95%.	84

Sumário

Resumo	ix
Abstract	xi
Lista de Figuras	xiii
Lista de Tabelas	xv
1 Introdução	1
1.1 Apresentação do Problema	3
1.2 Objetivos	4
1.3 Contribuições	5
1.4 Organização do Texto	7
2 Recomendação Associativa de Tags e o Problema de Cold Start	9
2.1 Recomendação Associativa de Tags	10
2.2 <i>Cold Start</i>	11
3 Trabalhos Relacionados	13
3.1 Qualidade de <i>Tags</i> e suas Aplicações na Web 2.0	13
3.2 Sistemas de Recomendação	15
3.3 <i>Cold Start</i> em Sistemas de Recomendação	19
3.4 <i>Relevance Feedback</i>	20
3.5 Recomendação de <i>Tags</i> : Métodos de Referência	21
3.5.1 Métricas de Relevância de <i>Tags</i>	22
3.5.2 Estratégias Associativas	26
3.5.3 Estratégia Baseada no Conteúdo de Outros Atributos Textuais .	28
3.5.4 Estratégia Baseada em <i>Relevance Feedback</i>	30
3.5.5 Estratégia Baseada em Programação Genética	31

4	Recomendação Associativa de <i>Tags</i> Robusta para o problema de <i>Cold Start</i>	35
4.1	Estratégias Baseadas em Filtragem de Termos	36
4.2	Estratégias de Recomendação Assistidas pelo Usuário	38
4.3	Explorando o <i>Feedback</i> do Usuário pela Aplicação de Programação Genética	42
4.4	Sumário dos Métodos de Recomendação	45
5	Avaliação Experimental	47
5.1	Bases de Dados	47
5.2	Metodologia de Avaliação	48
5.3	Parametrização	50
5.4	Impacto do <i>Cold Start</i> nos Métodos Associativos	53
5.5	Eficácia das Estratégias de Filtragem	55
5.6	Eficácia das Estratégias Assistidas pelo Usuário	57
5.6.1	Convergência dos Métodos Baseados em <i>Relevance Feedback</i>	58
5.6.2	Impacto do <i>Feedback</i> Negativo	59
5.6.3	<i>Relevance Feedback</i> em Cenários de <i>Cold Start</i>	60
5.6.4	Avaliação das Estratégias Baseadas em RF em Presença de <i>Tags</i>	61
5.6.5	Sensibilidade à Falta de Cooperação e Confiabilidade do Usuário	62
5.6.6	Exemplos de Recomendação	65
5.7	Resumo dos Resultados	67
6	Conclusões e Trabalhos Futuros	69
6.1	Conclusões	69
6.2	Trabalhos Futuros	70
	Referências Bibliográficas	73
	Apêndice A Revocação e Precisão Média das Estratégias de Recomendação	81

Capítulo 1

Introdução

A Web 2.0 surgiu como a consolidação de um conjunto de características que revolucionaram a maneira como a Internet é utilizada. Entre elas está o maior foco no usuário, que passou de frequentemente apenas consumidor do conteúdo disponibilizado para também participar como produtor desse conteúdo, muitas vezes denominado *mídia social*. Aplicações da Web 2.0 apresentam uma “arquitetura de participação” [O’Reilly, 2005], que permite que os usuários agreguem valor à aplicação na medida em que eles a utilizam, ao contrário de *sites* tradicionais, nos quais apenas os administradores podem modificar o conteúdo disponibilizado.

Diversas aplicações construídas sob os conceitos da Web 2.0 estão entre as mais populares da Web atualmente¹. Exemplos incluem o Youtube e o Yahoo! Video², dois sistemas de compartilhamento de vídeos, o Last.FM³ (ou simplesmente LastFM), uma estação de rádio *online* que possibilita a interação social em torno de tópicos relacionados a música, o Flickr⁴, uma aplicação de compartilhamento e organização de fotos, e o CiteULike⁵, um serviço de gerência e descoberta de referências bibliográficas voltado para a comunidade acadêmica. Fazendo uso da distribuição de conteúdo gerado por usuários e do estabelecimento de redes sociais, estas aplicações oferecem uma maior quantidade e diversidade de conteúdo em relação às aplicações da Web tradicional. O Youtube, por exemplo, é atualmente uma das maiores bases de dados, e o segundo sistema de busca mais utilizado do mundo [Figueiredo et al., 2013].

Aplicações da Web 2.0 são compostas por coleções de *objetos*, que podem estar armazenados em diversos tipos de mídias (áudio, vídeo, imagens). Cada objeto tem

¹<http://www.alexa.com/topsites> acessado em 01/04/2013

²<http://youtube.com> e <http://video.yahoo.com> acessado em 01/04/2013

³<http://last.fm> acessado em 01/04/2013

⁴<http://www.flickr.com/> acessado em 01/04/2013

⁵<http://www.citeulike.org> acessado em 01/04/2013

diversas fontes de informação associadas, denominadas *atributos*. Tais atributos podem ser de diversos tipos. *Atributos de conteúdo* são fontes de informação que podem ser extraídas do próprio objeto, tais como o histograma de cores de uma imagem [Swain & Ballard, 1991]. *Atributos textuais* são blocos de texto frequentemente associados ao objeto pelos usuários, tais como título, descrição e *tags*. *Atributos sociais*, por sua vez, refletem o contexto social em que o objeto está inserido, ou seja, quem o criou, quais os usuários o acessaram e quais relações entre usuários foram estabelecidas por meio do objeto.

Apesar da grande quantidade de conteúdo multimídia disponível em aplicações Web 2.0, a maioria dos serviços de Recuperação de Informação (RI), tais como máquinas de busca, sistemas de recomendação e classificação, fazem uso apenas dos atributos textuais associados ao conteúdo. Isso ocorre porque conteúdo textual é mais simples de ser processado e compreendido automaticamente do que mídias mais ricas, como imagens e vídeos, cujo processamento apresenta maior custo computacional. Além disso, tal processamento pode ser inviável devido tanto ao grande volume de dados nas aplicações Web 2.0 quanto também à qualidade pobre de grande parte desses dados [Boll, 2007].

Entretanto, por serem gerados pelos próprios usuários sem nenhum controle editorial, atributos textuais não oferecem garantias de qualidade (nem sintática, nem semântica) para outros usuários e para serviços de RI. Em particular, Figueiredo et al. [2013] realizaram uma extensa caracterização da qualidade de quatro atributos textuais, a saber, título, descrição, comentários e *tags*, em quatro aplicações diferentes, Youtube, Last.FM, Yahoo Video e Citeulike. A caracterização foi feita objetivando avaliar o potencial de cada atributo como fonte de dados para serviços de RI mais eficazes. Ela foi baseada em métricas de uso e heurísticas para estimar o poder descritivo e o poder discriminativo dos atributos. Os autores verificaram que alguns atributos textuais, tais como *tags* e comentários, ainda têm uso limitado em algumas aplicações Web 2.0, sendo ausentes em parcelas significativas dos objetos ou apresentando pequena quantidade de conteúdo. Nesse mesmo estudo, foi mostrado que, a despeito disso, *tags* estão entre os atributos com maior potencial para dar suporte a serviços de RI, tais como classificação automática. Além disso, *tags* constituem uma boa alternativa para organização, disseminação e recuperação de conteúdo.

Nesse contexto, um serviço que pode aumentar a quantidade e melhorar a qualidade do conteúdo das *tags* é a recomendação automática de *tags* por meio da sugestão de termos que idealmente descrevam o conteúdo de forma mais precisa e completa. Entretanto, os sistemas de recomendação presentes em algumas aplicações ainda são

bastante limitados. Por exemplo, no sistema Delicious⁶, as *tags* recomendadas a um usuário são uma interseção do vocabulário desse usuário e de todas as *tags* associadas ao conteúdo alvo da recomendação. No LastFM, por outro lado, são recomendadas as *tags* mais frequentemente associadas a um conteúdo. Note que tais estratégias de recomendação só sugerem *tags* que já foram associadas a um conteúdo.

Por outro lado, a literatura é rica em propostas de novos métodos de recomendação de *tags* baseados nas mais diversas estratégias [Sigurbjörnsson & van Zwol, 2008; Lipczak et al., 2009; Belém et al., 2011; Graham & Caverlee, 2008]. Entretanto, a maioria dessas propostas assume cenários ideais em que todos os dados necessários para que elas funcionem estejam disponíveis (e.g., todos os objetos possuem *tags* como em [Sigurbjörnsson & van Zwol, 2008]), sem no entanto analisar o impacto da ausência de parte desses dados, como ocorre comumente em cenários reais [Figueiredo et al., 2013]. Assim, torna-se necessário o estudo e desenvolvimento de estratégias para mitigar o efeito da falta de informação em métodos de recomendação já existentes, assim como o projeto de novos métodos de recomendação robustos à ausência dessa informação.

Portanto, se por um lado, recomendação de *tags* é um serviço de auxílio ao usuário que apresenta um grande potencial para melhorar a qualidade de *tags* e, consequentemente, a eficácia dos serviços de RI que exploram esse atributo, por outro, a maioria das soluções previamente propostas não consideram aspectos práticos, particularmente a falta de informação prévia para os recomendadores. Logo, faz-se necessário um estudo mais aprofundado sobre o assunto que leve em conta tais aspectos.

1.1 Apresentação do Problema

A literatura contém uma grande variedade de estratégias de recomendação de *tags*, sendo que as que exploram padrões de co-ocorrência com *tags* previamente associadas aos objetos do sistema (incluindo *tags* disponíveis no próprio objeto alvo da recomendação), chamados de *métodos associativos* têm produzido consistentemente resultados estado da arte [Heymann et al., 2008; Sigurbjörnsson & van Zwol, 2008; Garg & Weber, 2008; Menezes et al., 2010; Belém et al., 2011]. Em Belém et al. [2011], por exemplo, foram propostos algoritmos de recomendação que superaram os métodos encontrados na literatura até então. Naquele trabalho, foram considerados apenas objetos contendo *tags* previamente associadas e foram explorados como fonte de informação para os métodos além das próprias *tags*, métricas de relevância e termos extraídos de outros

⁶<http://delicious.com> acessado em 01/04/2013

campos textuais para recomendar *tags* que descrevam de uma forma mais precisa o objeto.

A observação feita em [Figueiredo et al., 2013] de que *tags* estão ausentes em uma parcela significativa dos objetos de várias aplicações populares da Web 2.0 levanta uma preocupação a respeito da eficácia de métodos associativos frente a ausência de *tags*. Este é um problema bem conhecido em sistemas de recomendação chamado *cold start*. Este problema é caracterizado pelo fato de o sistema de recomendação não conseguir fazer inferências sobre usuários ou itens sobre os quais ainda não tenha reunido informações suficientes [Schein et al., 2002], o que acarreta uma drástica redução da acurácia do mesmo.

O comportamento dos métodos associativos em um cenário de *cold start* ainda não foi avaliado. Visto que tais métodos podem sofrer uma degradação significativa da sua eficácia, uma vez que uma de suas fontes de informação (às vezes a única como em [Menezes et al., 2010]) não está disponível, torna-se necessário avaliá-los em tal cenário, já que como reportado em [Figueiredo et al., 2013], ele ocorre em uma fração significativa (em torno de 18%) dos objetos em várias aplicações da Web 2.0. Desse modo um dos objetivos desta dissertação é avaliar e propor métodos associativos de recomendação de *tags* que sejam mais robustos à falta de informação prévia e por conseguinte sejam mais eficazes em cenários reais.

Cabe ressaltar que a pesquisa existente em recomendação de *tags* pode ser dividida em duas correntes. Na primeira, a recomendação é *centrada no objeto*, e visa sugerir termos que o descrevam bem. Já na segunda corrente, chamada *personalizada*, o objetivo é sugerir termos que não só descrevam bem o objeto, mas também que sejam relevantes ao usuário alvo da recomendação, já que usuários diferentes podem optar por termos diferentes para descrever o mesmo objeto [Jäschke et al., 2007]. O foco dessa dissertação é em recomendação centrada no objeto. O desenvolvimento de métodos para a recomendação personalizada mais robustos à falta de informação será foco de trabalho futuro.

1.2 Objetivos

Os principais objetivos dessa dissertação são listados a seguir:

- Avaliar o impacto da falta de informação prévia (*cold start*) na eficácia de métodos de recomendação associativa de *tags* considerados estado-da-arte.
- Propor estratégias que visam minimizar o efeito do *cold start* sobre métodos associativos de recomendação de *tags*.

- Propor novos métodos de recomendação de *tags*, mais robustos ao *cold start*.
- Avaliar experimentalmente as estratégias propostas, comparando o desempenho destas estratégias entre si e com métodos de recomendação estado-da-arte em diferentes cenários e coleções de dados reais, coletadas de aplicações populares.

1.3 Contribuições

O estudo realizado pode ser dividido em várias partes. A primeira etapa consiste na quantificação do impacto do *cold start* sobre uma família de métodos estado-da-arte baseados em padrões de co-ocorrência de *tags* considerando quatro bases de dados coletadas de aplicações Web 2.0 populares. Nosso estudo revelou que a eficácia desses métodos é reduzida em até 84% em tal cenário. Essa perda é tão grande que, nesse cenário, os métodos de recomendação associativos são superados inclusive por métodos mais simples que não exploram padrões de co-ocorrência, diferentemente do que ocorre quando há *tags* previamente associadas ao objeto alvo da recomendação tal como reportado em [Belém et al., 2011]. Assim, torna-se necessário o desenvolvimento de técnicas que possam amenizar o impacto do *cold start* sobre estes métodos.

Na segunda parte do trabalho, foram exploradas estratégias de filtragem de modo a construir um conjunto alternativo de *tags*, as quais seriam usadas como entrada para os métodos associativos para obter padrões de co-ocorrência. Para tal, o trabalho focou nos métodos associativos estado-da-arte propostos em [Belém et al., 2011]. Foram avaliadas diversas estratégias para gerar o conjunto inicial de *tags* pela filtragem tanto de termos de objetos similares no conjunto de treino quanto de termos obtidos de outros atributos textuais do objeto alvo da recomendação (e.g., título e descrição). Entretanto os ganhos dessas estratégias são limitados, o que nos motivou a buscar novas soluções para o problema.

A principal contribuição desta dissertação é uma extensão dos métodos propostos em [Belém et al., 2011] para incluir as preferências por *tags* específicas manifestadas pelo usuário durante o processo de recomendação. Tais preferências são tratadas como um retorno implícito sobre a relevância de *tags* (*relevance feedback*). A ideia geral da estratégia proposta pode ser descrita em 3 passos: (i) um conjunto inicial de *tags* é apresentado para o usuário; (ii) o usuário seleciona *tags* relevantes nesse conjunto; (iii) as *tags* selecionadas como relevantes são usadas como entrada para o método de recomendação enquanto que as *tags* não relevantes (não selecionadas) são inseridas em uma lista negra para prevenir que elas sejam recomendadas novamente nas próximas iterações. Tal processo é repetido até que um resultado satisfatório seja obtido, ou seja

até que o usuário decida parar de adicionar tags ao objeto. Observe que a estratégia proposta leva em conta tanto as *tags* que foram selecionadas pelo usuário (feedback positivo), quanto as que não foram (feedback negativo) e requer apenas um pequeno esforço extra por parte do usuário⁷ que é recompensado pela melhoria da qualidade das tags recomendadas. Isso corresponde a um cenário em que o usuário deseja atribuir tags com o máximo de qualidade (i.e., relevância) para promover seu objeto no sistema, mas realizando o menor esforço possível.

Foram avaliadas algumas variações da estratégia assistida pelo usuário (i.e., estratégia baseada em *relevance feedback*) mencionada acima, desde o uso de simples heurísticas até a aplicação de Programação Genética (PG) [Banzhaf et al., 1998], um algoritmo de aprendizado de máquina, ao problema. Os resultados obtidos mostraram que a estratégia proposta reduz efetivamente o impacto do *cold start*, melhorando muito a eficácia dos métodos de recomendação originais baseados em co-ocorrência. Verificou-se, também, que a estratégia proposta aumenta em até 43% a eficácia dos métodos baseados em co-ocorrência mesmo em cenários nos quais o problema de *cold start* não ocorre.

Por fim, foi investigada a robustez da estratégia proposta, baseada em RF, à falta de cooperação do usuário. Em outras palavras, foi avaliado o impacto na eficácia das recomendações da adição de ruído (e.g., termos irrelevantes) e também o impacto do não assinalamento de termos relevantes por parte do usuário. Os resultados mostraram que a solução proposta permanece tão boa quanto, se não melhor que, outros métodos encontrados na literatura, mesmo que 10% do *feedback* provido pelo usuário seja composto por ruído.

De modo resumido, as principais contribuições deste trabalho são:

- Avaliação de vários métodos estado-da-arte de recomendação de *tags* associativos em um cenário de *cold start*.
- Proposição de várias estratégias de filtragem para gerar um conjunto alternativo de *tags* de modo a amenizar o *cold start* sobre uma família de métodos estado da arte.
- Proposição de métodos baseados em *relevance feedback* e programação genética que produzem recomendações de qualidade superior a métodos encontrados na literatura.

⁷Tipicamente tal esforço envolve apenas clicar em 2 ou 3 tags de uma lista de tags apresentadas ao usuário (geralmente 5 tags).

- Proposição de um arcabouço para avaliar a robustez de métodos baseados em *relevance feedback* à falta de cooperação do usuário.
- Avaliação da robustez dos métodos propostos baseados em *relevance feedback* à falta de cooperação do usuário.

1.4 Organização do Texto

O restante desta dissertação está organizado da seguinte forma: O capítulo 2 apresenta uma definição formal do problema aqui tratado bem como a terminologia empregada no estudo. O capítulo 3 contém uma revisão aprofundada da literatura no que diz respeito a métodos de recomendação e do impacto da ausência de informação sobre os mesmos, assim como sobre uso de *relevance feedback* e técnicas de aprendizado de máquina no projeto de métodos de recomendação de *tags*. Em particular a seção 3.5 apresenta métricas de relevância de *tags* bem como métodos de recomendação de *tags* previamente propostos que aqui são usados como ponto de partida e também para comparação com nossas propostas. O capítulo 4 descreve nossas estratégias para superar o *cold start* em sistemas de recomendação de *tags*. A avaliação experimental desses métodos é discutida no capítulo 5. Conclusões e direções de trabalhos futuros são apresentados no capítulo 6.

Capítulo 2

Recomendação Associativa de Tags e o Problema de Cold Start

Este capítulo apresenta uma contextualização do problema de recomendação de *tags* na ausência de informação prévia. A Seção 2.1 define a recomendação associativa de *tags*. Em seguida, a Seção 2.2 descreve o problema do *cold start* alvo do estudo aqui desenvolvido.

Sistemas de recomendação estão comumente presentes em aplicações da Web 2.0 e podem ser utilizados em diferentes contextos tais como recomendação de produtos em um site de comércio eletrônico, recomendação de artigos em uma biblioteca digital, recomendação de usuários de uma rede social. Nesta dissertação, os itens que estamos interessados em recomendar são *tags*, palavras-chave utilizadas para descrever de forma sucinta o conteúdo de um objeto de uma aplicação da Web 2.0.

O termo *objeto* é usado para referir a uma instância de uma mídia em uma dada aplicação Web 2.0. Exemplos de objetos são vídeos no Youtube, páginas web no Bibsonomy¹, fotos no Flickr e artistas no Last.FM. Há várias fontes de informação associadas a um objeto, aqui denominadas *atributos*. Em particular, esta dissertação explora *atributos textuais*, que compreendem blocos de texto associados ao objeto, e que geralmente apresentam tópicos ou funções bem definidas [Fernandes et al., 2007]. Mais precisamente, são explorados os atributos textuais título, descrição e *tags*. A Figura 2.1 mostra a página de um artista no Last.FM contendo alguns atributos textuais comumente associados a objetos em aplicações da Web 2.0. Na figura aparecem em destaque os atributos textuais título, *tags* e descrição estudados nesta dissertação como fontes de informação para a tarefa de recomendação de *tags*.

¹<http://bibisonomy.org> acessado em 01/03/2013

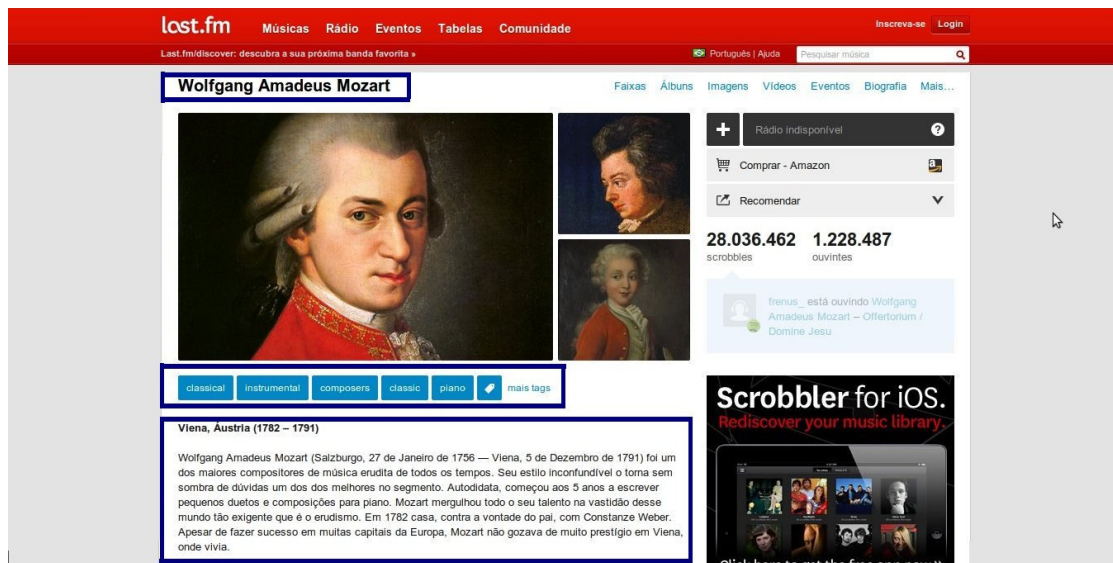


Figura 2.1. Página de um objeto do Last.FM.

2.1 Recomendação Associativa de Tags

Um sistema de recomendação de tags é geralmente utilizado para auxiliar usuários, fornecendo uma lista de recomendações que idealmente descrevam um dado objeto, ou seja, uma lista de tags que sejam relevantes para aquele objeto. Para produzir essa lista, o sistema de recomendação pode explorar diferentes fontes de informação. Por exemplo, o conteúdo de páginas editadas ou visualizadas por um usuário na Web, ou os produtos que um usuário adquire em um *site* de comércio eletrônico são evidências das preferências desse usuário.

Várias definições do problema de recomendação de tags são possíveis, dependendo das fontes de informação que são consideradas para os recomendadores. Seguindo a modelagem de [Belém et al., 2011; Menezes et al., 2010], essa dissertação foca em um tipo particular de recomendação em que são consideradas *tags* previamente associadas ao objeto e também os atributos textuais como fontes de informação para os métodos de recomendação. Mais formalmente, a tarefa de recomendar tags para um objeto alvo o pode ser definida como:

Definição. *Dado um conjunto \mathcal{I}_o de tags que já foram assinaladas para um objeto alvo o , e um conjunto $\mathcal{F}_o = \{\mathcal{F}_o^1, \mathcal{F}_o^2, \dots, \mathcal{F}_o^n\}$ dos outros atributos textuais associados a o , onde \mathcal{F}_o^i é o conjunto de termos associados ao atributo i , gere um conjunto de termos candidatos \mathcal{C}_o e recomende os k termos de \mathcal{C}_o de maior relevância.*

Cabe destacar que esta dissertação não estuda a recomendação personalizada, como pode ser visto pela definição acima. Tal escolha se deve ao fato de que um dos

objetivos desta dissertação é melhorar a qualidade das *tags*, tornando-as mais completas e precisas para dar suporte a outras tarefas de RI tais como busca e classificação, cenário no qual a recomendação centrada no objeto é suficientemente eficaz.

Logo, o problema de recomendação de *tags* é modelado como um problema de geração e ordenação de múltiplos termos candidatos por relevância. Em outras palavras, objetiva-se desenvolver uma função capaz de selecionar termos de um vocabulário e ordena-los de tal forma que termos que representem recomendações de mais alta qualidade (mais apropriadas para o conteúdo do objeto) apareçam primeiro que termos que representem recomendações de menor qualidade. Para desenvolver tal função, nesta dissertação serão utilizadas métricas que estimam a relevância de um termo como *tag*. A hipótese na qual esta proposta se baseia é que termos de maior qualidade (conforme capturado pelas métricas utilizadas) são mais relevantes para a tarefa de recomendação de *tags*. Logo, a solução para este problema é a função que melhor estime a relevância dos termos candidatos à recomendação e, conseqüentemente, sugira como *tags* os termos mais adequados ou mais relevantes a um conteúdo.

Muitos métodos anteriores de recomendação de *tags*, e particularmente aqueles que apresentam os melhores resultados na literatura, exploram padrões de coocorrência minerados sobre *tags* previamente associadas ao objeto alvo (e.g., *tags* em \mathcal{I}_o) e outros termos de uma coleção de objetos [Belém et al., 2011; Menezes et al., 2010; Sigurbjörnsson & van Zwol, 2008]. Essas técnicas são comumente referidas na literatura como técnicas de recomendação associativa de *tags*, já que elas aprendem padrões de coocorrência explorando regras de associação sobre um conjunto de treino $\mathcal{D} = \{\langle \mathcal{I}_d, \mathcal{F}_d \rangle\}$, onde \mathcal{I}_d ($\mathcal{I}_d \neq \emptyset$) é o conjunto de todas as *tags* e \mathcal{F}_d são os outros atributos textuais associados ao objeto $d \in \mathcal{D}$. Os padrões aprendidos são então usados para recomendar *tags* para objetos em um conjunto de teste $\mathcal{O} = \{\langle \mathcal{I}_o, \mathcal{F}_o \rangle\}$.

A próxima seção aborda um problema que pode acontecer em sistemas de recomendação, chamado *cold start* descrevendo seu efeito sobre tais sistemas.

2.2 *Cold Start*

Para que qualquer recomendador possa sugerir itens que sejam relevantes, é necessário que as fontes de informação que ele considera estejam disponíveis em tempo de recomendação. Quando tais fontes estão ausentes, a eficácia do recomendador pode sofrer degradação significativa, podendo inclusive tornar inútil seu uso. Tal problema, conhecido na literatura como *cold start*, é usualmente definido como uma falta de informação sobre usuários ou itens [Schein et al., 2002], e pode ocorrer em qualquer sistema

de recomendação, sendo motivo de vários estudos recentes [Preisach et al., 2010; Ness et al., 2009].

Do ponto de vista dos métodos associativos de recomendação de *tags*, uma instância do problema do *cold start* ocorre quando o objeto alvo o não possui tags previamente associadas, em outras palavras $\mathcal{I}_o = \emptyset$, enquanto que outros atributos textuais \mathcal{F}_o^i podem não estar vazios. Observe que essa definição não exige que todos os objetos do sistema não tenham tags associadas a eles. Entretanto o foco é na recomendação de tags a objetos que não possuam tags associadas e, para tal, pode-se utilizar a informação contida nos objetos que possuem tags. Observe ainda, que essa definição permite que um objeto em *cold start* possua outros atributos textuais associados (que não tags) com algum conteúdo. Tal definição corresponde a um cenário no qual um usuário está adicionando um novo objeto ao sistema e deseja associar algumas tags a ele, embora já tenham preenchido outros atributos.

O problema do *cold start*, como definido aqui, apresenta desafios para os métodos associativos de recomendação de tags, uma vez que eles não podem utilizar a informação de \mathcal{I}_o . Note que outras variações do problema podem ocorrer como no caso de novos objetos apenas com atributos textuais vazios. Entretanto, o foco desta disseminação é em um cenário no qual alguns dos atributos textuais de o (e.g., título e descrição) já foram preenchidos com algum conteúdo, e objetiva-se ajudar o usuário pela recomendação das primeiras tag(s) que serão atribuídas a o .

No próximo capítulo, é feita uma revisão aprofundada da literatura e são apresentados de maneira detalhada quatro métodos de recomendação que foram escolhidos com referências para este estudo. No capítulo 4, são apresentadas estratégias para amenizar o problema do *cold start* bem como são propostos novos métodos mais robustos ao mesmo.

Capítulo 3

Trabalhos Relacionados

Este capítulo discute os principais trabalhos relacionados, bem como apresenta quatro métodos de recomendação de *tags* que são considerados como referências para a avaliação das estratégias propostas nessa dissertação. Mais especificamente, a seção 3.1 apresenta trabalhos que exploram o uso de *tags* para melhorar outros serviços de RI. Já a seção 3.2 discute vários trabalhos sobre recomendação de itens e *tags*. Trabalhos que abordam o problema do *cold start* em sistemas de recomendação são mostrados na seção 3.3. A seção 3.4 apresenta trabalhos que utilizam estratégias de *relevance feedback* em vários contextos de RI. A seção 3.5 apresenta inicialmente uma série de métricas de relevância de *tags* que são utilizadas pelos métodos de referência, bem como pelos métodos propostos nesta dissertação. Em seguida, quatro métodos de referência são apresentados, sendo dois deles métodos associativos, um terceiro que explora apenas outros atributos textuais e, finalmente, um que é baseado em *relevance feedback*. Por fim, é apresentado o arcabouço de programação genética que nesta dissertação é estendido para incluir *relevance feedback* de modo a tornar o método mais robusto ao *cold start*.

3.1 Qualidade de *Tags* e suas Aplicações na Web 2.0

Muitos trabalhos avaliam a qualidade, sob diferentes perspectivas, de diversos atributos textuais em aplicações da Web 2.0 [Figueiredo et al., 2013; Li et al., 2008a]. Em particular, alguns trabalhos focam no atributo textual *tags* e a sua aplicação para melhorar serviços como busca [Schenkel et al., 2008], recomendação [Sigurbjörnsson & van Zwol, 2008], agrupamento [Chen et al., 2009; Ramage et al., 2009] e indexação [Song

et al., 2008]. No entanto, não há um consenso a respeito da qualidade de *tags* (assim como de outros atributos) gerados por usuários. Por exemplo, Li et al. [2008b] afirmam que *tags* refletem os interesses dos usuários e portanto são de boa qualidade. Por outro lado, Marshall [2009], que faz uma análise semântica manual dos atributos textuais associados a fotos do Flickr, conclui que *tags* são empregadas majoritariamente para uso pessoal, sendo pouco descritivas do conteúdo a que foram associadas em relação a outros atributos textuais, como o título e a legenda das imagens.

Figueiredo et al. [2013] avaliam não apenas *tags*, como também outros três atributos textuais a saber, título, descrição e comentários, com funcionalidades semelhantes em quatro aplicações diferentes, Youtube, Yahoo Video, LastFM e CiteULike. Os autores realizam uma ampla caracterização do uso desses atributos, que permite compará-los em relação a diferentes aspectos, como utilização, poder descritivo e poder discriminativo. Nesse estudo, os autores concluem que atributos textuais colaborativos (por exemplo *tags* em aplicações como o Last.FM e Yahoo Video) são ausentes em parcelas significativas dos objetos nas aplicações estudadas, mas tendem a apresentar maior quantidade de conteúdo quando presentes. Eles também verificam uma diversidade significativa de conteúdo entre os atributos, o que motiva seu uso combinado em serviços de RI. Quanto ao poder descritivo e discriminativo, os autores concluem que título e *tags* são os atributos de maior qualidade segundo as métricas heurísticas propostas por eles.

Ainda no contexto de caracterização do uso dos atributos textuais, vários trabalhos propõem métricas para avaliar a relevância de um termo associado a um atributo textual de um objeto [Figueiredo et al., 2013; Lipczak et al., 2009; Sigurbjörnsson & van Zwol, 2008]. Algumas dessas métricas serão vistas em mais detalhes na Seção 3.5.1.

Venetis et al. [2011] examinam a criação de nuvens de *tags* para explorar e sumarizar um conjunto de objetos. Eles apresentam um conjunto de métricas que capturam propriedades estruturais das nuvens de *tags*, e apresentam três algoritmos para realizar a seleção de *tags* para representar um conjunto de objetos.

No contexto de classificação e agrupamento, Ramage et al. [2009] contrastam o uso de *tags* associadas por usuários com o uso do conteúdo textual completo contido nas páginas da Web correspondentes. Os autores mostram que combinar ambos os atributos textuais leva a uma melhoria na qualidade dos resultados de dois algoritmos de agrupamento. Outro trabalho com um resultado similar é o de Chen et al. [2009], que propõe a combinação de atributos sonoros e *tags* de músicas do LastFM para classificação das músicas, melhorando os resultados obtidos pelo uso isolado dos atributos sonoros.

Além dos trabalhos centrados nos atributos textuais, vários estudos focam no uso

de técnicas de RI em conteúdo de mídias ricas (imagens, vídeos, áudios) em aplicações da Web 2.0. Exemplos incluem o uso de atributos visuais e textuais para agrupar imagens similares no Flickr [Rege et al., 2008] e o uso de atributos do conteúdo de vídeos para anexar propagandas relevantes a um vídeo [Mei et al., 2007]. Entretanto, estas técnicas foram aplicadas em amostras pequenas de dados, e sua escalabilidade para a Web 2.0 ainda é um problema em aberto. Além disso, ainda existe uma lacuna semântica entre conceitos humanos representados por exemplo por uma consulta e atributos visuais/sonoros extraídos de mídias ricas. Logo, é de fundamental importância a disponibilidade de anotações textuais (e.g., *tags*) associadas ao conteúdo multimídia de aplicações da Web 2.0 [Sigurbjörnsson & van Zwol, 2008].

A próxima seção discute trabalhos que abordam métodos de recomendação, focando especificamente naqueles que abordam a recomendação de *tags*.

3.2 Sistemas de Recomendação

Sistemas de recomendação têm se tornado extremamente comuns nos dias de hoje. Alguns exemplos de destaque incluem os sistemas de recomendação de produtos do Amazon.com¹ e de filmes do Netflix². Nesse contexto, diversos algoritmos têm sido desenvolvidos para dar suporte aos mais diferentes serviços de recomendação, tais como recomendação de filmes [Koren, 2008], livros [Givon & Lavrenko, 2009] e *tags* [Belém et al., 2011]. Por exemplo, Koren [2008] propõe um método baseado na decomposição de matrizes em valores singulares que descobre tópicos latentes para realizar recomendações de filmes para o Netflix. Já Mei et al. [2011] propõem um sistema de recomendação de vídeos que usa uma abordagem multimodal, combinando informação do conteúdo do vídeo, metadados e *relevance feedback* fornecido pelo usuário.

Particularmente, recomendação de *tags* tem atraído a atenção de pesquisadores, uma vez que *tags* têm se mostrado uma boa alternativa não só para gerar recomendações de melhor qualidade [Guy et al., 2010; Wetzker et al., 2009], mas também para melhorar o desempenho de outros serviços de RI, tais como classificação automática [Figueiredo et al., 2013] e busca [Li et al., 2008a]. Tal serviço consiste em técnicas para sugerir *tags* a um usuário, auxiliando-o na descrição do conteúdo de um objeto alvo da recomendação.

A maioria das estratégias de recomendação de *tags* previamente propostas exploram termos extraídos dos metadados do objeto alvo e padrões de coocorrência entre *tags* para sugerir novas *tags* para um objeto alvo. Em particular, algumas delas exploram padrões de coocorrência de *tags* para expandir um conjunto inicial \mathcal{I}_o associado a

¹<http://www.amazon.com/> acessado em 01/04/2013

²<http://www.netflix.com> acessado em 01/04/2013

um objeto o alvo da recomendação [Heymann et al., 2008; Sigurbjörnsson & van Zwol, 2008; Garg & Weber, 2008; Menezes et al., 2010; Belém et al., 2011]. Para isso, tais trabalhos utilizam regras de associação, isto é, implicações do tipo $X \rightarrow y$, onde X é um conjunto de *tags* e y é uma *tag* candidata, selecionando as regras mais importantes a partir de um limiar de confiança (i.e., probabilidade condicional de y ser selecionado como *tag* dado que X foi selecionado).

Sigurbjörnsson & van Zwol [2008] analisam como os usuários associam *tags* a fotos no sistema Flickr e que tipos de *tags* são empregadas neste sistema, utilizando uma amostra de 52 milhões de páginas da aplicação. As estratégias de recomendação propostas pelos autores exploram métricas globais de coocorrência de termos. Além das métricas de coocorrência, os autores consideram diferentes métricas para ordenar as *tags* recomendadas por relevância. Em particular, eles propuseram utilizar a frequência da *tag* em uma amostra de dados de treino como métrica de relevância.

Por questões de eficiência, a maioria dos métodos associativos computam padrões de coocorrência apenas entre duas *tags*. Em outras palavras o antecedente X contém apenas uma *tag*. Em contraste, Menezes et al. [2010] propõem o LATRE - *Lazy Associative Tag Recommendation* - que computa tais regras de uma maneira sob demanda, permitindo uma geração eficiente de regras mais complexas e potencialmente melhores.

Belém et al. [2011] propõem métodos de recomendação de *tags* que exploram de uma maneira conjunta coocorrência de termos com *tags* previamente associadas ao objeto alvo, termos extraídos de múltiplos atributos textuais do objeto alvo e métricas de relevância de termos. Os autores mostram que tal abordagem supera em muito tanto o LATRE quanto o melhor método proposto em [Sigurbjörnsson & van Zwol, 2008], em muitas bases de dados.

Em uma outra direção, alguns esforços de recomendação de *tags* não fazem uso de *tags* previamente associadas ao objeto alvo, dando ênfase em outras fontes de dados. Por exemplo, Lipczak et al. [2009] propõem CTTR - *Co-occurrence and Text-based Tag Recommender* - que extrai termos de outros atributos textuais (e.g., título e descrição) do objeto alvo da recomendação, expande esses termos extraídos utilizando regras de associação, e ordena tais termos pelo seu uso como *tags* em um conjunto de treino. Um estudo similar é apresentado em [Wang et al., 2009], embora os autores utilizem a métrica $TF \times IDF$, tradicionalmente empregada em tarefas de RI, para extrair e ordenar por relevância os termos mais importantes do conteúdo textual do objeto. Lu et al. [2009], bem como Zhang et al. [2009] propagam *tags* entre objetos que tenham conteúdo textual similar. Já Graham & Caverlee [2008] propõem Plurality, um método simples que combina um modelo vetorial com *relevance feedback* provido por usuários.

Algumas técnicas exploram o conteúdo de objetos multimídia diretamente. Por

exemplo, uma abordagem baseada em conteúdo para expandir um conjunto de *tags* é descrita em [Siersdorfer et al., 2009]. Os autores utilizam conteúdo duplicado em vídeos para criar ligações em um grafo e propagar *tags* entre vídeos similares. Já Wu et al. [2009] adicionam informações retiradas do conteúdo da imagem para ordenar *tags* no Flickr.

Outras técnicas focam no problema da recomendação personalizada. Por exemplo, abordagens que utilizam os relacionamentos de amizade entre usuários para melhorar serviços de recomendação de *tags* têm sido empregados com sucesso em diversos trabalhos. Guan et al. [2009] constroem um grafo bipartido de usuários, documentos e *tags* para gerar recomendações personalizadas. Os autores modelam a recomendação como um problema de “*query and ranking*”, e propõem um algoritmo baseado em grafos para ordenar objetos de tipos diferentes (documentos, *tags* e usuários) inter-relacionados. Quando um usuário aplica um conjunto de *tags*, o documento e o usuário são tratados como consultas, representando relevância e personalização, respectivamente. Konstas et al. [2009] levam em consideração os relacionamentos de amizade inerentes à rede social estabelecida entre usuários do LastFM, e mostram que essa informação melhora os métodos tradicionais de recomendação baseados apenas no conteúdo dos objetos.

Rendle & Lars [2010] apresentam o método de recomendação personalizado de *tags* PITF - *Pairwise Interactions Tensor Factorization*. Nele, um tensor representando os eventos de atribuição de *tags* é decomposto em matrizes que representam as interações entre usuários, objetos e *tags*. O critério de otimização BRP - *Bayesian Personalized Rank* - é empregado para aprendizagem do modelo. Garg & Weber [2008] propõem um método de recomendação de *tags* personalizado que explora o conhecimento sobre o comportamento passado de cada usuário, combinando informação local (relativas ao histórico de uso de *tags* de um usuário específico) e global (relativa ao histórico de todos os usuários do sistema). Já Jäschke et al. [2007] propõem *FolkRank*, uma adaptação do conhecido *PageRank*, para realizar recomendações personalizadas de *tags*. A intuição por trás do *FolkRank* é que um objeto que recebe *tags* relevantes de usuários importantes se torna importante, ou seja, relevante como recomendação. O mesmo acontece, simetricamente, com *tags* e usuários, ou seja, uma *tag* é relevante se ela foi associada a objetos importantes por usuários importantes. Assim, define-se um grafo cujos vértices se reforçam mutuamente por meio da propagação de pesos, que resulta em uma ordenação final desses vértices, particularmente de *tags*. Essa dissertação trata apenas de recomendação de *tags* centrada no objeto deixando a recomendação personalizada para trabalhos futuros.

Poucos trabalhos exploram técnicas baseadas em *learning to rank* (L2R) para a tarefa de recomendação de *tags* [Cao et al., 2009; Wu et al., 2009; Belém et al., 2011].

Esse tipo de estratégia é baseada no aprendizado de um modelo que permite ordenar *tags* por relevância a partir de um conjunto de atributos (e.g., métricas de relevância). O aprendizado ocorre com base em um conjunto de treino contendo para cada *tag* de exemplo t associada a um objeto um rótulo indicando o seu grau de relevância a esse objeto e os valores dos atributos correspondentes a t .

Os métodos de recomendação de *tags* baseados em L2R previamente propostos são: o trabalho de Cao et al. [2009], baseado no algoritmo RankSVM [Cristianini & Shawe-Taylor, 2000], o trabalho de Wu et al. [2009], baseado no algoritmo RankBoost [Freund et al., 2003] e o trabalho de Belém et al. [2011] baseado no uso de programação genética. Entretanto, o primeiro não considera coocorrências de *tags*, e o segundo não considera os atributos textuais de um objeto para a tarefa de recomendação. Já Belém et al. [2011] propõem a aplicação de programação genética [Poli, 2002] e do RankSVM ao problema de recomendação de *tags*, entretanto ele considera apenas cenários nos quais objetos alvo de recomendação possuam *tags* previamente associadas. Nesta dissertação, esse trabalho será expandido para incluir o *relevance feedback* como uma fonte adicional de informação para o algoritmo de programação genética de modo a tornar tal arcabouço mais robusto em cenários de *cold start*.

Em suma, a literatura contém um grande número de trabalhos que focam no problema de recomendação de *tags* sem considerar nenhuma restrição com relação a *tags* previamente associadas aos objetos. Desses métodos, aqueles que exploram padrões de coocorrência com *tags* previamente associadas ao objeto alvo da recomendação, chamados métodos associativos, têm obtido consistentemente resultados que são muito competitivos e frequentemente superiores a outras alternativas. Entretanto eles foram avaliados apenas para objetos com *tags* previamente associadas. Para objetos sem nenhuma *tag* (uma instância do problema *cold start*), em contraste, sua eficácia pode ser prejudicada, uma vez que eles não poderão explorar os padrões de coocorrência. Assim, uma contribuição dessa dissertação é uma avaliação do impacto do *cold start* sobre esses métodos que são tratados como base de comparação e são apresentados em mais detalhes na seção 3.5. Por outro lado, métodos que não utilizam *tags* previamente associadas podem ser mais robustos a esse problema. Em particular, os métodos CTTR [Lipczak et al., 2009] e Plurality [Graham & Caverlee, 2008] são aqui considerados como referência para comparação com nossas propostas e portanto são apresentados de maneira mais detalhada na seção 3.5. Plurality é de particular interesse uma vez que ele também explora *relevance feedback*, assim como uma de nossas estratégias. Entretanto, nesse quesito ele explora apenas o *feedback* positivo provido pelos usuários, já o método proposto nesta dissertação explora tanto o *feedback* positivo quanto o negativo.

3.3 *Cold Start* em Sistemas de Recomendação

O *cold start* é um problema prevalente em sistemas de recomendação, relacionado à questão de que o sistema não pode fazer inferências para usuários ou itens sobre os quais ainda não tenha reunido informações suficientes. Muitos trabalhos anteriores tentam solucionar, ou pelo menos amenizar, o problema usando as mais diversas estratégias [Lam et al., 2008; Gunawardana & Meek, 2009; Sahebi & William, 2011; Liu et al., 2011].

Algumas das abordagens existentes para lidar com tal problema se apoiam em técnicas de aprendizado de máquina para melhorar filtros colaborativos. Com esse intuito, Bobadilla et al. [2012] propõem uma nova métrica de similaridade gerada a partir do uso de uma rede neural. Já Zhou et al. [2011] propõem uma técnica de levantamento de preferências de usuários baseada em uma árvore de decisão para construir uma entrevista inicial que é aplicada a novos usuários com o objetivo de evitar que eles permaneçam em *cold start*.

Outras técnicas para lidar com o *cold start* exploram atributos textuais dos objetos, como por exemplo *tags*, para aumentar a acurácia de sistemas de recomendação de itens. Por exemplo, Givon & Lavrenko [2009] mostram como prever de forma mais precisa preferências dos usuários usando um conjunto de *tags* associado a um livro em um sistema de recomendação de livros. Já Said et al. [2009] investigam o problema da recomendação de itens durante os primeiros meses de existência do CiteULike a fim de melhorar os resultados de recomendação durante esse período inicial, quando o sistema possuía pouca informação sobre os objetos. Eles apresentam uma abordagem probabilística para recomendação de itens, estendendo modelos propostos anteriormente, como o *probabilistic latent semantic analysis* (PLSA) [Hofmann, 2013].

Finalmente, muitos trabalhos anteriores combinam o uso de um filtro colaborativo com o uso de atributos textuais para amenizar os efeitos do *cold start*. Por exemplo, Sun et al. [2011] lidam com o problema do *cold start* em sistemas de recomendação de itens combinando o uso de uma árvore de decisão com informações retiradas do conteúdo textual do objeto para construir uma nova métrica de similaridade entre objetos.

Em contraste, existem poucos trabalhos que tratam do problema do *cold start* no contexto específico de Recomendação de *tags*. Ness et al. [2009] descrevem como *stacked generalization*, uma técnica baseada na análise do conteúdo de áudio, ajuda a melhorar a eficácia de um sistema de recomendação de *tags* para conteúdo musical frente ao *cold start*. Este tipo de método, entretanto, requer algoritmos especializados para cada tipo de mídia (e.g., imagens, vídeos), o que tipicamente tem uma alta complexidade computacional.

Preisach et al. [2010] propõem um algoritmo semi-supervisionado puramente baseado em grafos para realizar recomendações personalizadas. A ideia básica do algoritmo consiste em propagar *tags* pelo grafo reduzindo assim o efeito do *cold start*. A abordagem proposta não explora padrões de coocorrência, sendo baseada apenas nas relações entre usuários e objetos. Essa dissertação não aborda recomendação personalizada, logo o estudo proposto em [Preisach et al., 2010] não pode ser diretamente comparado aos métodos propostos nesta dissertação.

Em suma, *cold start* é um problema largamente conhecido e trabalhado em sistemas de recomendação de itens. Entretanto, os poucos trabalhos que estudam tal problema no contexto específico de recomendação de *tags* ou utilizam algoritmos especializados, que dependem do tipo de mídia da aplicação, ou analisam o problema do ponto de vista de recomendação personalizada. Assim um dos objetivos desta dissertação é contribuir com um estudo que aborda uma parte pouco explorada do problema, o impacto de *cold start* na recomendação de *tags* centrada no objeto.

3.4 *Relevance Feedback*

Vários trabalhos tentam explorar o usuário e a potencial assistência que ele pode dar a vários métodos de busca e recomendação. Em particular, o *Relevance feedback* é largamente utilizado em muitas tarefas de RI, principalmente em busca [Lv & Zhai, 2009; Xu & Akella, 2008; Giacinto, 2007; Wang et al., 2011]. Nesse contexto, Xu & Akella [2008] propõem utilizar um “*mixture model*”, isto é, um modelo estatístico que aproxima a distribuição de probabilidade de diferentes tipos de indivíduos em uma população, em conjunto com o *relevance feedback* provido pelo usuário a fim de melhorar a acurácia de buscas em que a consulta inicial retorna muitos documentos de baixa qualidade. Um trabalho similar é apresentado por Wang et al. [2011], que propõem um método em que o *relevance feedback* provido pelo usuário é usado para ajustar os pesos de uma função de ordenação de modo a melhorar os resultados de determinadas consultas. Já Giacinto [2007] propõe uma abordagem baseada em uma busca pelo vizinho mais próximo que utiliza o *relevance feedback* do usuário para refinar buscas de imagens baseadas em conteúdo.

Ainda no contexto de busca, alguns trabalhos exploram também o *feedback* negativo (informação sobre conteúdo não relevante) provido pelo usuário. Por exemplo, Algarni et al. [2009] analisam dois aspectos relativos ao uso do *feedback* negativo: a seleção de exemplos negativos interessantes e a revisão do modelo de busca para acomodar a informação obtida do *feedback* negativo. Já Ferreira et al. [2011] propõem um

método baseado em programação genética que usa ambos os tipos de *feedback* para melhorar a busca de imagens.

No contexto de recomendação de itens, alguns trabalhos exploram o *relevance feedback* para refinar recomendações tornando-as mais relevantes para o usuário. Por exemplo, Mei et al. [2011] propõem um método multimodal de recomendação de vídeos que combina, usando uma ponderação linear, as informações extraídas do conteúdo visual, sonoro e textual com *relevance feedback*. Já Yang et al. [2007] propõem um sistema de recomendação de vídeos que utiliza *relevance feedback* para realizar um ajuste de pesos em seu modelo, enquanto Wu et al. [2008] usam o *feedback* negativo para melhorar recomendações em sistemas de pergunta e resposta. Para isso, os autores propõem uma adaptação de um *Probabilistic Latent Semantic Analysis* (PLSA) [Hofmann, 2013] que leva em consideração tanto o *feedback* positivo quanto negativo provido pelo usuário.

Pelo que sabemos, Plurality [Graham & Caverlee, 2008] é o único método anteriormente publicado que explora *relevance feedback* em recomendação de *tags*, entretanto como discutido na seção 3.2 e detalhado no capítulo 3.5, ele explora apenas o *feedback* positivo. Entretanto, tanto o *feedback* positivo quanto o negativo têm sido continuamente usados em vários outros contextos de recomendação [Ferreira et al., 2011; Yang et al., 2007]. Em todo caso, o uso de *feedback* positivo e negativo para resolver problemas de *cold start* no contexto de sistemas de recomendação de *tags* (principalmente naqueles baseados em padrões de coocorrência) é uma contribuição original de nosso trabalho. Mais ainda, nenhum trabalho anterior realizou a análise da robustez dos recomendadores de *tags* à falta de cooperação e confiabilidade dos usuários, uma outra contribuição desta dissertação.

3.5 Recomendação de *Tags*: Métodos de Referência

Esta seção apresenta quatro métodos de recomendação de *tags* estado-da-arte tomados neste trabalho como referências para a comparação com as novas propostas. Tais métodos são apresentados da mesma forma como foram propostos nos trabalhos originais, considerando as mesmas fontes de informação que os autores dos métodos consideraram. Dois desses métodos foram propostos recentemente e superaram várias alternativas em cenários sem *cold start* [Belém et al., 2011], além de serem tomados como ponto de partida para o desenvolvimento das soluções propostas. Os outros dois métodos focam no uso de outros atributos textuais para realizar a recomendação de *tags*. Por não explorarem *tags* previamente associadas ao objeto, eles podem ser

mais robustos ao *cold start*. Um desses métodos é de especial interesse uma vez que ele também explora o *relevance feedback* dos usuários. Logo, esses dois métodos são naturalmente boas bases de comparação para a avaliação das estratégias propostas neste trabalho. Ao fim dessa seção, também é apresentada a estratégia baseada em programação genética proposta em [Belém et al., 2011] que será estendida na seção 4.3 para usar o *relevance feedback*. Antes da apresentação detalhada dos métodos de referência, esta seção apresenta várias métricas de relevância de *tags* exploradas por eles. Tais métricas também são utilizadas como parte das soluções propostas para amenizar os efeitos do *cold start* na eficácia de métodos associativos, conforme será discutido na Seção 4.1.

3.5.1 Métricas de Relevância de *Tags*

Uma série de métricas heurísticas foram propostas para avaliar a relevância de termos candidatos à recomendação [Belém et al., 2011; Sigurbjörnsson & van Zwol, 2008; Lipczak et al., 2009; Venetis et al., 2011].

Algumas dessas métricas exploram padrões de coocorrência de *tags* aprendidos a partir de *regras de associação*, isto é, implicações do tipo $X \rightarrow y$, onde o *antecedente* X é um conjunto de *tags* e o *consequente* y é um termo candidato à recomendação. A importância de uma regra de associação é estimada com base no **suporte** (σ), que é o número de coocorrências de X e y no conjunto de treino \mathcal{D} , e na **confiança** (θ), a probabilidade condicional de que o termo y seja associado como *tag* a um objeto $d \in \mathcal{D}$ dado que todas as *tags* em X são também associadas a d .

Dado que o número de regras mineradas a partir de \mathcal{D} pode ser muito grande e algumas dessas regras podem não ser úteis para recomendação, limiares de suporte e confiança (σ_{min} e θ_{min} , respectivamente) são usados como limites inferiores para selecionar apenas as regras mais frequentes e/ou confiáveis. Esta seleção pode melhorar tanto a eficiência quanto a eficácia do recomendador.

Em tempo de recomendação, são selecionadas as regras cujos antecedentes estejam incluídos em \mathcal{I}_o , isto é no conjunto de *tags* disponíveis no objeto o alvo de recomendação. Para cada termo c que aparece como consequente em qualquer uma das regras selecionadas, sua relevância como *tag* para o é estimada pela soma das confianças de todas as regras contendo c , isto é [Sigurbjörnsson & van Zwol, 2008]:

$$Sum(c, o, \ell) = \sum_{X \subseteq \mathcal{I}_o} \theta(X \rightarrow c), \quad (X \rightarrow c) \in \mathcal{R}, |X| \leq \ell \quad (3.1)$$

onde \mathcal{R} é um conjunto de regras de associação computadas *offline* sobre um conjunto

de treino \mathcal{D} dados limiares σ_{min} e θ_{min} , e ℓ é o tamanho limite para o número de elementos no antecedente de uma regra de associação.

Sigurbjörnsson & van Zwol [2008] também propõem outra métrica que utiliza os padrões de coocorrência, chamada *Vote*. Para um dado candidato c e um objeto alvo o , $Vote(c, o)$ é o número de regras de associação cujo antecedente é um termo em \mathcal{I}_o e cujo consequente é um termo candidato c , isto é:

$$Vote(c, o) = \sum_{x \in \mathcal{I}_o} j, \text{ onde } j = \begin{cases} 1, & \text{se } (x \rightarrow t) \in \mathcal{R} \\ 0, & \text{caso contrário} \end{cases} \quad (3.2)$$

Outras métricas tentam capturar a capacidade de um termo candidato c de discriminar um objeto dos demais. Termos que são muito comuns e gerais (por exemplo, “vídeo” no YouTube) ou muito raros e portanto potencialmente muito específicos, tendem a ter baixo poder de discriminação. Tais termos não são bons candidatos para a recomendação. A métrica *estabilidade* (*Stab*) [Sigurbjörnsson & van Zwol, 2008] tenta capturar este aspecto, dando mais importância a termos com frequências intermediárias, ou seja:

$$Stab(c, k_s) = \frac{k_s}{k_s + |k_s - \log(f_c^{tag})|} \quad (3.3)$$

onde f_c^{tag} é o número de objetos em \mathcal{D} que contém c como *tag*, e k_s representa a “frequência ideal” de um termo e deve ser ajustada de acordo com a coleção.

Algumas métricas tentam combinar a capacidade de um termo candidato c de discriminar um objeto dos demais com a capacidade das regras de associação de gerar bons candidatos a recomendação. Um exemplo dessas métricas é o $Vote^+$ [Sigurbjörnsson & van Zwol, 2008] construído a partir de *Vote* ponderando cada voto pelo valor de *Stab* de ambos antecedente e consequente. Ou seja:

$$Vote^+(c, o, k_x, k_c, k_r) = \sum_{x \in \mathcal{I}_o} j \times Stab(x, k_x) \times Stab(c, k_c) \times Rank(c, x, k_r),$$

$$\text{onde } j = \begin{cases} 1, & \text{se } x \rightarrow c \in \mathcal{R} \\ 0, & \text{caso contrário} \end{cases} \quad (3.4)$$

Por outro lado, pode-se argumentar que a recomendação de termos muito frequentes em \mathcal{D} pode ser uma boa estratégia quando nenhuma informação é fornecida sobre o , o que seria o caso de um novo objeto com atributos textuais vazios. A métrica *Cobertura* (*Cov*), proposta em [Venetis et al., 2011] para estimar a qualidade de uma *tag* para uma nuvem de *tags*, captura esse efeito, uma vez que dá mais importância às

tags mais freqüentes:

$$Cov(c) = \frac{f_c}{|\mathcal{D}|} \quad (3.5)$$

onde f_c representa o número de objetos em \mathcal{D} que contém c em pelo menos um atributo textual (e não somente *tags*), e $|\mathcal{D}|$ é o número de objetos em \mathcal{D} .

É possível argumentar que recomendar termos mais infreqüentes (desde que não sejam muito raros) seja desejável, porque eles têm um potencial maior para discriminar objetos em diferentes categorias, tópicos ou níveis de relevância. Este aspecto pode ser capturado pela métrica *Inverse Feature Frequency* (IFF) [Figueiredo et al., 2013], uma adaptação do tradicional *Inverse Document Frequency* (IDF) [Baeza-Yates & Ribeiro-Neto, 2011] que considera a frequência do termo em um atributo textual específico (no nosso caso, *tags*). Dado o número de objetos no conjunto de treino $|\mathcal{D}|$, o *IFF* de um termo candidato c é definido como:

$$IFF(c) = \log \frac{|\mathcal{D}| + 1}{f_c^{tag} + 1} \quad (3.6)$$

onde f_c^{tag} é o número de objetos em \mathcal{D} para os quais c aparece como *tag*. Note que c pode ser extraído de outros atributos textuais. O valor 1 é somado ao numerador e ao denominador para tratar de novos termos que não aparecem como *tag* no conjunto de treino.

Como argumentado em [Belém et al., 2011], um bom poder descritivo também é desejável para uma *tag*. Para capturar essa propriedade, os autores consideram quatro métricas heurísticas, a saber: espalhamento (*term spread* ou *TS*), frequência do termo (*term frequency* ou *TF*), espalhamento ponderado do termo (*weighted term spread* ou *wTS*) e frequência ponderada do termo (*weighted term frequency* ou *wTF*).

O pressuposto por trás da heurística *TS* é que quanto maior o número de atributos textuais de o que contém c , mais relacionado o termo candidato c está ao conteúdo de o . Assim, o espalhamento de um termo candidato c em um objeto o , $TS(c, o)$, é dado pelo número de atributos textuais (exceto *tags*) associado a o que contêm c , ou seja:

$$TS(c, o) = \sum_{\mathcal{F}_o^i \in \mathcal{F}_o} j, \text{ onde } j = \begin{cases} 1 & \text{se } c \in \mathcal{F}_o^i \\ 0 & \text{caso contrário} \end{cases} \quad (3.7)$$

Diferentemente da métrica *TS*, que leva a estrutura do objeto em consideração, a frequência do termo c em um objeto o , $TF(c, o)$ considera todos os atributos textuais como uma única lista de termos e conta todas as ocorrências de c nela. Isto é, dado $tf(c, \mathcal{F}_o^i)$, o número de ocorrências de c em um dado atributo textual i de o , $TF(c, o)$

é dada por:

$$TF(c, o) = \sum_{\mathcal{F}_o^i \in \mathcal{F}_o} tf(c, \mathcal{F}_o^i) \quad (3.8)$$

Belém et al. [2011] propõem extensões destas heurísticas, referidas como *wTS* e *wTF* respectivamente, que consideram que os diferentes atributos textuais podem apresentar diferentes capacidades descritivas [Figueiredo et al., 2013]. As métricas *wTS* e *wTF* ponderam a ocorrência de c em um atributo textual \mathcal{F}_o^i de o pelo poder descritivo médio desse atributo em todos os objetos do conjunto de treino, conforme definido a seguir.

Para estimar o poder descritivo médio de um atributo textual \mathcal{F}^i , Belém et al. [2011] usam uma heurística proposta anteriormente chamada espalhamento médio do atributo (*average feature spread* ou *AFS*) [Figueiredo et al., 2013]. Seja o *espalhamento do atributo* \mathcal{F}_d^i (*feature instance spread* ou *FIS*(\mathcal{F}_d^i)) associado ao objeto $d \in \mathcal{D}$ a média dos valores de *TS* sobre todos os termos em \mathcal{F}_d^i . *AFS*(\mathcal{F}^i) é definido como a média dos valores de *FIS*(\mathcal{F}_d^i) computada sobre todas as instâncias de \mathcal{F}^i associadas a objetos em \mathcal{D} . As métricas *wTS* e *wTF* são então definidas como:

$$wTS(c, o) = \sum_{\mathcal{F}_o^i \in \mathcal{F}_o} j, \text{ onde } j = \begin{cases} AFS(\mathcal{F}^i) & \text{se } c \in \mathcal{F}_o^i \\ 0 & \text{caso contrário} \end{cases} \quad (3.9)$$

$$wTF(c, o) = \sum_{\mathcal{F}_o^i \in \mathcal{F}_o} tf(c, \mathcal{F}_o^i) \times AFS(\mathcal{F}^i) \quad (3.10)$$

Finalmente, outro aspecto importante para a recomendação de *tags* é a previsibilidade dos termos candidatos, uma vez que termos mais previsíveis são geralmente candidatos mais confiáveis à recomendação [Lipczak et al., 2009]. Diferentes heurísticas para estimar a previsibilidade dos termos foram propostas na literatura [Heymann et al., 2008; Lipczak et al., 2009]. Nesta dissertação, foi adotada a métrica proposta por Lipczak et al. [2009], que mede a previsibilidade de um termo c que ocorre em um atributo textual \mathcal{F}_o^i do objeto o como a probabilidade condicional de que c seja usado como uma *tags* dado que ele também aparece em algum atributo textual \mathcal{F}^i . Esta probabilidade é calculada sobre todos os objetos do conjunto \mathcal{D} . Isto é:

$$p(c \in \mathcal{I}_o | c \in \mathcal{F}_o^i) = \frac{\sum_{d \in \mathcal{D}} I(c \in \mathcal{I}_d \wedge c \in \mathcal{F}_d^i)}{\sum_{d \in \mathcal{D}} I(c \in \mathcal{F}_d^i)} \quad (3.11)$$

onde $I(x) = 1$ se x é verdadeiro e 0 caso contrário. Nesta dissertação, a previsibilidade de que um termo c , que ocorre em pelo menos um dos n atributos textuais (exceto

$tags$) de o , seja uma tag , é computada como:

$$Pred(c, o) = \sum_{i=1}^n p(c \in \mathcal{I}_o | c \in \mathcal{F}_o^i) \quad (3.12)$$

Já Heymann et al. [2008] medem a previsibilidade de um termo a partir do cálculo da entropia do mesmo. A entropia de um termo c em relação ao atributo textual $tags$, $H^{tags}(c)$, é definida como:

$$H^{tags}(c) = - \sum_{(c \rightarrow i) \in \mathcal{R}} \theta(c \rightarrow i) \log(\theta(c \rightarrow i)) \quad (3.13)$$

Se um termo ocorre consistentemente com certas $tags$, ele é mais previsível, apresentando baixa entropia. Por outro lado, termos que ocorrem indiscriminadamente com muitas outras $tags$ são menos previsíveis, apresentando entropia mais alta. Em outras palavras, $H^{tags}(c)$ mede a concentração dos valores de confiança de todas as regras de associação cujo antecedente é c . Se um termo é ausente no conjunto de treino, associa-se um valor de entropia arbitrariamente alto a ele, visto que, nesse caso, o resultado não é um número real. A entropia de um termo pode ser particularmente útil como critério de desempate, visto que é melhor recomendar termos mais “consistentes” ou menos “confusos”. Enquanto a entropia foi usada em Heymann et al. [2008] apenas para avaliar recomendações, aqui ela é empregada em funções de recomendação.

As seções seguintes mostram como as métricas apresentadas nesta seção foram exploradas em quatro estratégias estado-da-arte tomadas como referência para nosso estudo.

3.5.2 Estratégias Associativas

Os dois métodos de recomendação associativa utilizados como referência e ponto de partida para o desenvolvimento de novas soluções são Sum^+wTS e $LATRE^+wTS$, os dois melhores métodos de recomendação propostos em [Belém et al., 2011]. Ambos métodos exploram conjuntamente $tags$ previamente associadas ao objeto alvo de recomendação, múltiplos atributos textuais e algumas métricas de relevância de $tags$.

Sum^+wTS estende um método de recomendação anterior, chamado Sum^+ [Sigurbjörnsson & van Zwol, 2008], pela introdução da métrica wTS (Eq. 3.9). Sum^+ estima a relevância de cada termo candidato c para um objeto o em função dos padrões de coocorrência de $tags$ previamente associadas a o e da estabilidade do termo

candidato c (Eq. 3.3), sendo definida como:

$$Sum^+(c, o, k_x, k_c, k_r) = \sum_{x \in \mathcal{I}_o} \theta(x \rightarrow c) \times Stab(x, k_x) \times Stab(c, k_c) \times Rank(c, o, k_r) \quad (3.14)$$

onde k_x , k_c e k_r são parâmetros de configuração. $Rank(c, o, k_r)$ é um fator de suavização definido como $k_r / (k_r + p(c, o))$, onde $p(c, o)$ é a posição de c quando se consideram os candidatos ordenados de acordo com a confiança das regras de associação.

Sum^+wTS estima a relevância de um termo candidato c pela combinação linear da saída das métricas Sum^+ e wTS , que leva em conta o poder descritivo de c . Ou seja:

$$Sum^+wTS(c, o, k_x, k_c, k_r, \alpha) = \alpha Sum^+(c, o, k_x, k_c, k_r) + (1 - \alpha)wTS(c, o) \quad (3.15)$$

O parâmetro α ($0 \leq \alpha \leq 1$) é usado como fator de ponderação. Note que, assim como a métrica Sum (Eq. 3.1) em que é baseada, Sum^+ é computada sobre todos os candidatos gerados a partir das regras de associação com *tags* previamente associadas ao objeto o . Em contraste, wTS é computado também para os termos extraídos dos outros atributos textuais de o . Como em [Belém et al., 2011], as regras de associação são geradas pelo uso do algoritmo *Apriori* [Agrawal & Srikant, 1994].

Sum^+ e Sum^+wTS , assim como outros métodos baseados em coocorrência [Sigurbjörnsson & van Zwol, 2008; Garg & Weber, 2008], restringem o tamanho das regras de associação para apenas uma *tag* no antecedente ($\ell=1$) por razões de desempenho. Em contraste, *LATRE* - *Lazy Associative Tag Recommender* [Menezes et al., 2010] - é capaz de gerar eficientemente regras de associação maiores. Isso porque as regras são geradas sob demanda, diferentemente de outros métodos (i.e., *Apriori*), que computam todas as regras antecipadamente sobre o conjunto de treino, possivelmente incluindo regras que não sejam úteis durante o processo de recomendação. *LATRE* ordena cada termo candidato c pela soma das confianças de todas as regras contendo c , isto é, ele usa a métrica Sum (Eq. 3.1) com $\ell \geq 1$, explorando assim apenas padrões de coocorrência.

De modo similar ao Sum^+wTS , o método $LATRE^+wTS$, estende *LATRE* pela introdução da métrica wTS :

$$LATRE^+wTS(c, o, \ell, \alpha) = \alpha Sum(c, o, \ell) + (1 - \alpha)wTS(c, o) \quad (3.16)$$

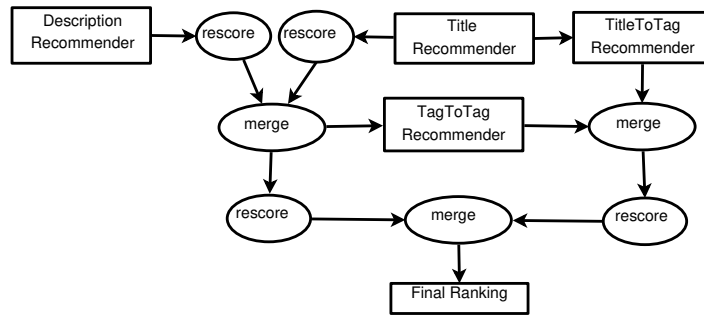


Figura 3.1. Algoritmo CTTR (Adaptado) [Lipczak et al., 2009].

3.5.3 Estratégia Baseada no Conteúdo de Outros Atributos Textuais

O terceiro método utilizado como referência é o *Co-occurrence and Text based Tag Recommender* (CTTR), que explora termos extraídos de outros atributos textuais e uma métrica de relevância de *tags*, mas não considera *tags* previamente associadas ao objeto, e assim não sofre os efeitos de *cold start*. CTTR é uma adaptação do vencedor do ECML Discovery Challenge 2009 [Lipczak et al., 2009], que em adição aos dois aspectos anteriormente mencionados leva em conta também o histórico de atribuição de *tags* para prover recomendações personalizadas. Como esta dissertação não trata recomendações personalizadas, estatísticas dos usuários não são incluídas no método CTTR.

A estrutura básica do CTTR é apresentada na Figura 3.1. O primeiro passo é a extração de potenciais candidatos à recomendação a partir dos atributos textuais (título e descrição) associados ao objeto alvo o . A cada termo candidato $c \in \mathcal{F}_o^i$ é atribuído um peso s_c^i igual à taxa de utilização de c como *tag* no conjunto de treino. Essa taxa é a razão entre o número de objetos do conjunto de treino em que o termo c foi utilizado tanto como *tag* quanto no atributo textual i e o número total de vezes em que c foi associado ao atributo i .

Em seguida, os conjuntos de termos candidatos gerados a partir do título e da descrição são intercalados. Segundo Lipczak et al. [2009], o título tende a fornecer recomendações mais precisas que outros atributos textuais, e isso deve ser refletido na etapa de intercalação. Para isso, os autores propõem uma estratégia de (re)ponderação chamada *leading precision rescorer*. Tal estratégia usa a precisão média na primeira posição da lista ordenada de recomendações, $avgP@1$, calculada para o conjunto de dados de treino, como a nova pontuação para o primeiro candidato da lista de termos candidatos a recomendação. O peso dos candidatos seguintes são modificados proporcionalmente. Seja s_j o peso original do j -ésimo termo candidato na lista de

recomendações, o novo peso s'_j do j -ésimo termo será:

$$s'_j = \frac{avgP@1 \times s_j}{s_1} \quad (3.17)$$

Depois de reajustar os pesos, os termos candidatos são combinados por meio de um procedimento de intercalação, como segue. Seja $S_c = \{s_c^1, s_c^2, \dots, s_c^n\}$ o conjunto de diferentes pontuações para o mesmo termo candidato c . A função de intercalação é definida como:

$$intercala(S_c) = 1 - \prod_{s_c^j \in S_c} (1 - s_c^j) \quad (3.18)$$

Os termos extraídos no primeiro passo são então “expandidos” por meio de regras de associação. Entretanto, diferentemente do método Sum^+ , *tags* previamente associadas ao objeto alvo não são exploradas. Em vez disso, termos extraídos de outros atributos textuais são usados como antecedentes. Para isso, Lipczak et al. [2009] distinguem dois tipos de relacionamentos de coocorrência: (1) entre *tags* (antecedente e consequente são *tags* do conjunto de treino), que define o conjunto $R_{Tag, Tag}$, e (2) entre termos do título de um objeto e suas *tags*, nos quais o antecedente é um termo do título e o consequente é uma *tag* do mesmo objeto, que define o conjunto $R_{Titulo, Tag}$. No momento da recomendação, os conjuntos de regras relacionados aos termos extraídos são combinados. Termos do título são utilizados como antecedentes na equação abaixo para encontrar *tags* candidatas relacionadas a eles:

$$S_{Titulo, Tag}(c, o) = 1 - \prod_{x \in \mathcal{F}_o^{titulo}} (1 - \theta(X \rightarrow c) \times s_x^{titulo}) \quad (3.19)$$

onde $\theta(X \rightarrow c) \in R_{Titulo, Tag}$ e s_x^{titulo} é a taxa de utilização do termo x como *tag*, sendo x proveniente do título. De modo similar, o conjunto de termos resultante da intercalação entre título e descrição é usado como antecedente na seguinte equação:

$$S_{Tag, Tag}(c, o) = 1 - \prod_{x \in \cup_i \mathcal{F}_o^i} (1 - \theta(X \rightarrow c) \times s_x) \quad (3.20)$$

onde $\theta(X \rightarrow c) \in R_{Tag, Tag}$ e $s_x = intercala(\{p_x^{titulo}, p_x^{descricao}\})$ é o peso resultante da intercalação título-descrição mencionada acima. Note que s_x pode ser interpretado como uma métrica de relevância que captura a importância dos termos nos atributos textuais de um objeto, de modo similar ao TF , TS e suas variações, discutidas na Seção 3.5.1.

No passo final, os pesos obtidos a partir das regras de associação ($S_{Titulo, Tag}$

e $S_{Tag,Tag}$) assim como os pesos atribuídos aos termos do título e da descrição do objeto alvo são reajustados e intercalados, resultando na ordenação final dos termos por relevância.

3.5.4 Estratégia Baseada em *Relevance Feedback*

O último método estado-da-arte usado como referência nesta dissertação é o Plurality [Graham & Caverlee, 2008], que explora *relevance feedback* obtido do usuário e, como o CTTR, não utiliza *tags* previamente associadas ao objeto alvo da recomendação. Começando com algumas sugestões iniciais obtidas a partir de um filtro colaborativo, Plurality iterativamente as refina pela incorporação do feedback provido pelo usuário.

As sugestões iniciais de *tags* para um objeto alvo o são produzidas a partir de uma busca pelos objetos mais similares a o contidos na base de treino \mathcal{D} . Cada objeto $d \in \mathcal{D}$ é modelado como um *bag of terms* extraídos de todos os seus atributos textuais (incluindo as *tags* de d). A similaridade entre cada objeto $d \in \mathcal{D}$ e o é então computada usando a métrica do cosseno [Baeza-Yates & Ribeiro-Neto, 2011]:

$$Sim(d, o) = \frac{\vec{d} \bullet \vec{o}}{|d| \times |o|} = \frac{\sum_{i=1}^{|\mathcal{V}|} w_{i,d} \times w_{i,o}}{\sqrt{\sum_{i=1}^{|\mathcal{V}|} w_{i,d}^2} \times \sqrt{\sum_{i=1}^{|\mathcal{V}|} w_{i,o}^2}} \quad (3.21)$$

onde $|\mathcal{V}|$ é o tamanho do vocabulário em \mathcal{D} , e o peso $w_{i,d}$ é uma variação do $TF \times IDF$, e.g., $w_{i,d} = \sqrt{freq(t_i, d)} \times (1 + \log(\frac{|\mathcal{D}|}{df(t_i)+1}))$, onde $freq(t_i, d)$ é a frequência do termo t_i do objeto d e $df(t_i)$ é igual ao número de objetos em \mathcal{D} contendo o termo.

Para cada termo t contido em um dos top- n_{obj} objetos com maior similaridade com o , Plurality atribui o peso:

$$termScore(t, o) = \sum_{i=1}^{n_{obj}} Sim(o, d_i)^4 \times freq_{tag}(t, d_i) \quad (3.22)$$

onde $freq_{tag}(t, d_i)$ é o número de vezes que t foi aplicado como *tag* ao objeto $d_i \in \mathcal{D}^3$. As top- n_{tags} *tags* com os maiores pesos são então selecionadas como sugestões iniciais e mostradas aos usuários, que fornece *feedback* sobre a relevância dos mesmos com relação ao objeto ao qual se pretende fazer a recomendação. *Tags* julgadas relevantes são então expandidas utilizando-se padrões de coocorrência com outras *tags* em \mathcal{D} . As novas *tags* retornadas para o usuário são aquelas que co-ocorrem mais frequentemente com qualquer uma das *tags* julgadas relevantes na iteração anterior. Este processo é

³Note que em muitas aplicações, uma mesma *tag* pode ser atribuída multiplas vezes a um mesmo objeto por usuários diferentes.

repetido até que um resultado satisfatório ou um número máximo de iterações M , seja atingido.

Em outras palavras, seja $\mathcal{I}_{\mathcal{D}}$ um conjunto de todas as *tags* associadas a todos os objetos de \mathcal{D} , ou seja, $\mathcal{I}_{\mathcal{D}} = \{t \mid t \in \mathcal{I}_d \text{ e } d \in \mathcal{D}\}$. Para cada par $t_i, t_j \in \mathcal{I}_{\mathcal{D}}$, Plurality pré-computa um valor de coocorrência para o par (t_i, t_j) como segue. Cada *tag* t_i é representada por um vetor \vec{t}_i , com dimensão $|\mathcal{D}|$, em que cada posição contém o número de vezes que a *tag* t_i foi assinalada ao objeto $d \in \mathcal{D}$. O valor computado das coocorrências para um par (t_i, t_j) captura a similaridade entre t_i e t_j com relação as atribuições realizadas aos objetos de \mathcal{D} , i.e.:

$$\text{pairScore}(t_i, t_j) = \frac{\vec{t}_i * \vec{t}_j}{|t_i| \times |t_j|} \quad (3.23)$$

Dado T_{rel} , o conjunto de *tags* julgadas relevantes pelo usuário, o peso das coocorrências sobre uma *tag* candidata c é definido como:

$$\text{tagScore}(c) = \sum_{t_i \in T_{rel}} \text{pairScore}(c, t_i) \quad (3.24)$$

As n_{tags} *tags* com maiores valores de *tagScore* são retornadas para o usuário para que ele possa novamente prover seu feedback. Assim, Plurality explora apenas *tags* julgadas como relevantes pelos usuários (e.g., feedback positivo) para gerar novas recomendações. Este é um ponto em que os métodos propostos nesta dissertação diferem do Plurality, já que eles exploram tanto o *feedback* positivo quanto o negativo.

3.5.5 Estratégia Baseada em Programação Genética

Além dos métodos descritos na seção 3.5.2, Belém et al. [2011] também propõem uma estratégia que usa programação genética [Poli, 2002] para gerar uma função de recomendação de *tags*. Nesta dissertação tal estratégia será estendida para se tornar mais robusta ao problema do *cold start*. Antes de apresentar a estratégia de Belém et al. [2011] é mostrado o funcionamento geral de algoritmos baseados em programação genética.

3.5.5.1 Programação Genética

Programação genética (PG) [Poli, 2002] é um processo iterativo que implementa um mecanismo de busca global. A ideia básica da PG é gerar uma solução mais eficiente ou eficaz para um problema a partir de um conjunto inicial de possíveis soluções que são combinadas, usando para isso ideias aprendidas do processo evolucionário biológico.

No arcabouço de PG, cada possível solução, chamada indivíduo, é representada por uma árvore que é composta por métricas relacionadas ao problema. Em cada geração do processo evolucionário, os indivíduos são avaliados de acordo com uma função de *Fitness*, também relacionada ao problema. O processo é repetido até que um valor de *Fitness* dado como meta seja atingido ou até que um número máximo de gerações seja alcançado.

A Figura 3.2 mostra em mais detalhes o processo evolucionário usado pelo algoritmo de PG. Inicialmente, um conjunto de indivíduos é gerado de modo aleatório. Em seguida o algoritmo iterativamente combina os indivíduos de modo a gerar soluções melhores. O valor da *Fitness* é usado como critério para selecionar os melhores indivíduos, que transmitirão suas características para gerações futuras a partir de operações tais como cruzamento e mutação. No final, o indivíduo com melhor valor de *Fitness*, que normalmente pertence à última geração do processo evolucionário, é escolhido como solução para o problema.

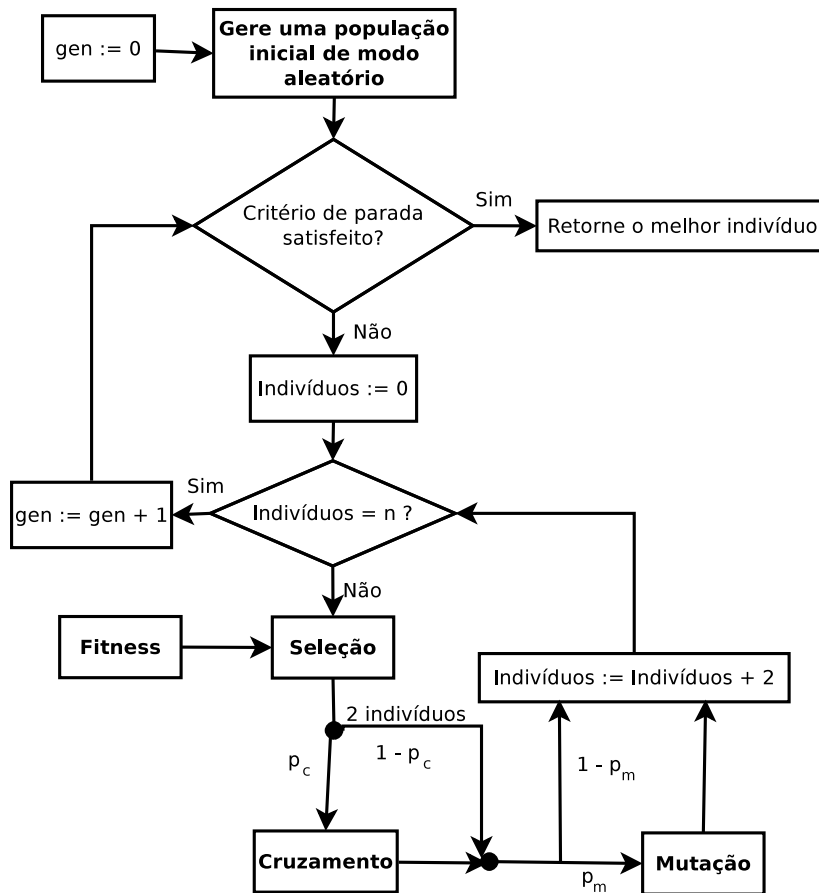


Figura 3.2. Processo evolucionário do algoritmo de programação genética [Belém et al., 2011].

A cada iteração do processo evolucionário, uma nova população de tamanho n é gerada. Para isso, dois indivíduos da geração atual são escolhidos e com probabilidade p_c trocam “material genético” na operação de *cruzamento* para gerar um novo indivíduo. A escolha de tais indivíduos é feita pelo método de *seleção por torneio*, ou seja, k indivíduos da população são sorteados, com reposição, e o indivíduo que apresentar maior valor de *Fitness* é escolhido. De cada uma das duas árvores que representam os indivíduos é escolhido um nó, e os dois nós juntamente com as subárvores abaixo deles são trocados. O objetivo dessa operação é combinar boas soluções rumo às direções mais promissoras do espaço de busca. Em seguida, com probabilidade p_m , aplica-se a operação de *mutação*, com o objetivo de aumentar a diversidade da população. Isto é útil, por exemplo, para evitar que o processo evolucionário de busca fique preso em mínimos locais. A mutação de um indivíduo é feita selecionando-se aleatoriamente um dos nós da árvore e substituindo-o (juntamente com sua subárvore correspondente) por uma nova subárvore gerada aleatoriamente, sem exceder a profundidade máxima permitida d .

3.5.5.2 Programação Genética Aplicada à Recomendação

Belém et al. [2011] foram os primeiros a propor o uso de PG no problema de recomendação de *tags*. A solução proposta pelos autores consiste em representar cada indivíduo por uma árvore correspondente à função de valoração de um termo candidato à recomendação.

Mais especificamente, uma árvore é composta de terminais (folhas) e não terminais (nós internos). Belém et al. [2011] escolheram as operações de soma (+), subtração (−), multiplicação (\times), divisão (/) e logaritmo natural (\ln) como não-terminais. Para assegurar a propriedade de fechamento, foram implementados divisão e logaritmo protegidos, de forma que esses operadores retornem o valor padrão 0 quando suas entradas estiverem fora de seus respectivos domínios. Os autores utilizaram como terminais um conjunto $L_{métricas}$ de métricas de relevância e um conjunto de constantes (uniformemente distribuídas entre 0 e 1). As métricas de relevância consideradas para compor o conjunto $L_{métricas}$ foram: *Sum*, *LATRE*, *IFF*, *Stab*, *TS*, *TF*, *wTS*, *wTF*, *H*, *Sum+*, *Vote* e *Vote+* discutidas na seção 3.5.1. A Figura 3.3 mostra exemplos de possíveis árvores e suas funções correspondentes.

A *Fitness* de um indivíduo representa a qualidade das recomendações produzidas pela função de valoração correspondente. Em [Belém et al., 2011], os autores avaliaram a qualidade de uma recomendação em função da precisão nas k primeiras posições dos termos recomendados ($P@k$, conforme definido na Equação 5.1), utilizando $k = 5$.

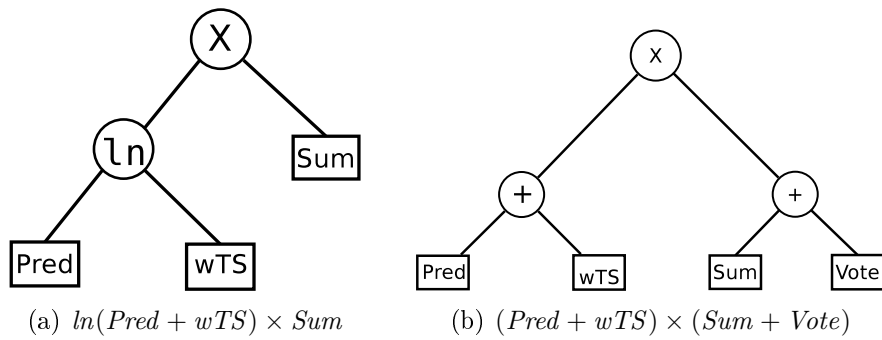


Figura 3.3. Exemplos de árvores.

Sendo assim, a *Fitness* de uma função de valoração foi definida como a média de $P@k$ sobre todas as recomendações produzidas pela função em uma amostra de objetos de tamanho s extraída de um conjunto de validação \mathcal{V} .

A PG é um método não-linear efetivo que tem sido empregado com sucesso quando há um grande espaço de busca e um objetivo a ser otimizado, produzindo resultados próximos do ótimo em várias aplicações [Banzhaf et al., 1998]. De fato, a PG foi aplicada a várias tarefas e Recuperação de Informação, como classificação, busca e recuperação de imagens (por exemplo, [Yeh et al., 2007]). Nesta dissertação a proposta de Belém et al. [2011] é estendida para explorar o uso de PG em conjunto com *relevance feedback*. Além disso, aqui mais métricas são consideradas para compor o conjunto $L_{metrics}$ e uma geração de termos candidatos mais abrangente é considerada, como será discutido na Seção 4.3.

Capítulo 4

Recomendação Associativa de *Tags* Robusta para o problema de *Cold Start*

Os métodos de recomendação associativos, em particular o Sum^+wTS e $LATRE^+wTS$, têm demonstrado eficácia superior a outras alternativas em cenários nos quais o objeto alvo de recomendação possua algumas *tags* previamente associadas a ele [Belém et al., 2011]. Entretanto como pode ser visto nas Equações 3.15 e 3.16, tanto o Sum^+wTS quanto o $LATRE^+wTS$ reduzem à aplicação da métrica wTS sobre os termos candidatos quando $\mathcal{I}_o = \emptyset$, isto é na instância de *cold start* estudada nessa dissertação. Nesse caso, esses métodos são capazes de recomendar apenas *tags* extraídas dos outros atributos textuais do objeto o , já que os padrões de coocorrência não podem ser explorados. Isso limita o potencial desses métodos para descobrir novas *tags*, o que em última instância pode causar uma perda de precisão desses métodos em tal cenário. Uma forma simples de tentar reduzir esse impacto é construir um conjunto alternativo \mathcal{I}'_o de *tags* iniciais e usá-lo no lugar de \mathcal{I}_o como entrada para os métodos associativos de recomendação.

Diferentes estratégias podem ser adotadas para se construir \mathcal{I}'_o . Por exemplo, filtrar termos de outras fontes, como outros atributos textuais do objeto alvo, ou de outros objetos presentes na base de treino \mathcal{D} , e então usar tais termos para gerar \mathcal{I}'_o .

Esta dissertação considera ambas as abordagens descritas acima. Mais especificamente estendemos dois métodos associativos estado-da-arte a saber, o Sum^+wTS e o $LATRE^+wTS$ de modo a torná-los mais robustos em cenários nos quais o *cold start* ocorra. Outra contribuição é a extensão desses métodos, bem como a proposição de um novo método híbrido, de modo a considerar uma nova fonte de informação, o *relevance*

feedback, que como será mostrado no capítulo 5 auxilia a amenizar o impacto do *cold start*.

Neste capítulo, são apresentadas as extensões discutidas acima, assim como a estratégia baseada em *relevance feedback*. A Seção 4.1 apresenta estratégias automáticas para filtragem de termos de outras fontes para construir um conjunto inicial de *tags* \mathcal{I}'_o alternativo sobre o qual os padrões de coocorrência podem ser obtidos. Em seguida, a Seção 4.2 apresenta uma estratégia que explora o uso do *relevance feedback* como uma nova fonte de informação para construir o conjunto \mathcal{I}'_o .

4.1 Estratégias Baseadas em Filtragem de Termos

Primeiramente, é apresentada uma estratégia baseada na filtragem de termos provenientes de outros atributos textuais do objeto o , usando como critério de filtragem as métricas heurísticas apresentadas na Tabela 4.1 e discutidas na Seção 3.5.1. Tal estratégia pode ser vista no Algoritmo 1. Para cada termo $t \in \mathcal{F}_o$, é atribuído um peso dado pela métrica usada (linha 2). São selecionados os k_{filter} termos com os maiores pesos para serem usados como \mathcal{I}'_o (linha 5). Para armazenar os pesos atribuídos aos termos uma tabela hash chamada *utility_as_tag* é utilizada (linha 2). Por fim \mathcal{I}'_o é fornecido ao método de recomendação *rec* juntamente com os outros atributos textuais do objeto o (linha 6). Em nossa avaliação experimental consideramos como o método *rec* a ser usado (linha 6) tanto o Sum^+wTS quanto o $LATRE^+wTS$.

Algoritmo 1 Filtragem automática de termos a partir de outros atributos textuais

Entrada: Objeto alvo o , número k_{filter} de termos a serem filtrados, métrica heurística h , método associativo de recomendação *rec*

Saída: Uma lista de *tags* R_o recomendadas para o .

- 1: **for each term** $t \in \bigcup_{i=1}^n F_o^i$ **do**
 - 2: $utility_as_tag[t] = h(t, o)$
 - 3: **end for**
 - 4: Ordene *utility_as_tag* em ordem decrescente
 - 5: $\mathcal{I}'_o =$ primeiros k_{filter} termos em *utility_as_tag*
 - 6: **return** *rec*(o, \mathcal{I}'_o)
-

Uma alternativa para a construção automática do conjunto \mathcal{I}'_o é a extração de *tags* de outros objetos da base de treino \mathcal{D} , utilizando-se para isso um filtro colaborativo inspirado pela primeira etapa do Plurality [Graham & Caverlee, 2008]. A estratégia pode ser vista no Algoritmo 2, isto é, a similaridade entre o e cada objeto $d \in \mathcal{D}$ é calculada (linha 2), os top- n_{obj} objetos mais similares a o são selecionados (linhas 3), usando a Equação 3.21. Então, de todas as *tags* presentes nesses objetos, são escolhidas

Tabela 4.1. Métricas heurísticas utilizadas para a filtragem automática de termos de atributos textuais do objeto alvo da recomendação.

Nome	Sigla	Equação
Frequência do Termo	TF	3.8
Espalhamento do Termo	TS	3.7
Frequência Ponderada do Termo	wTF	3.10
Espalhamento Ponderado do Termo	wTS	3.9
Previsibilidade do Termo	Pred	3.12
Cobertura	Cov	3.5

as k_{filter} com os maiores pesos (linhas 11-12), sendo esses pesos calculados pela Equação 3.22 (linha 8). Novamente a estrutura $utility_as_tag$ é utilizada e como método de recomendação rec tanto o Sum^+wTS quanto o $LATRE^+wTS$ são considerados (linha 13).

Por fim, é possível ainda utilizar outro método de recomendação que não utilize $tags$ previamente associadas ao objeto alvo e que apresente eficácia razoável para gerar um conjunto inicial \mathcal{I}'_o , que possa então ser usado como entrada de um método associativo de recomendação. Nesta dissertação, é testado o uso do método $CTTR$ para esse fim.

Algoritmo 2 Filtragem automática de termos a partir de objetos similares em \mathcal{D}

Entrada: Objeto alvo o , número $top-n_{obj}$ de objetos similares, número k_{filter} de termos a serem filtrados, método associativo de recomendação rec , conjunto de objetos de treino \mathcal{D}

Saída: Uma lista de $tags$ R_o recomendadas para o .

- 1: **for each object** $d \in \mathcal{D}$ **do**
 - 2: $similarObjScores[d] = Sim(d, o)$ (Equação 3.21)
 - 3: **end for**
 - 4: Ordene $similarObjScores$ em ordem decrescente
 - 5: $mostSimObj =$ primeiros $top-n_{obj}$ objetos em $similarObjScores$
 - 6: **for each object** $d \in mostSimObj$ **do**
 - 7: **for each tag** $t \in d$ **do**
 - 8: $utility_as_tag[t] = termScore(t, o)$ (Equação 3.22)
 - 9: **end for**
 - 10: **end for**
 - 11: Ordene $utility_as_tag$ em ordem decrescente
 - 12: $\mathcal{I}'_o =$ primeiros k_{filter} termos em $utility_as_tag$
 - 13: **return** $rec(o, \mathcal{I}'_o)$
-

4.2 Estratégias de Recomendação Assistidas pelo Usuário

Em comum, as estratégias apresentadas na Seção 4.1 são baseadas em métodos automáticos para gerar \mathcal{I}'_o . Entretanto, como será visto na Seção 5.5, elas produzem apenas ganhos marginais sobre o melhor método de referência. Assim, a principal contribuição desta dissertação é propor e avaliar uma nova estratégia baseada em *relevance feedback* (RF) para construir e refinar \mathcal{I}'_o .

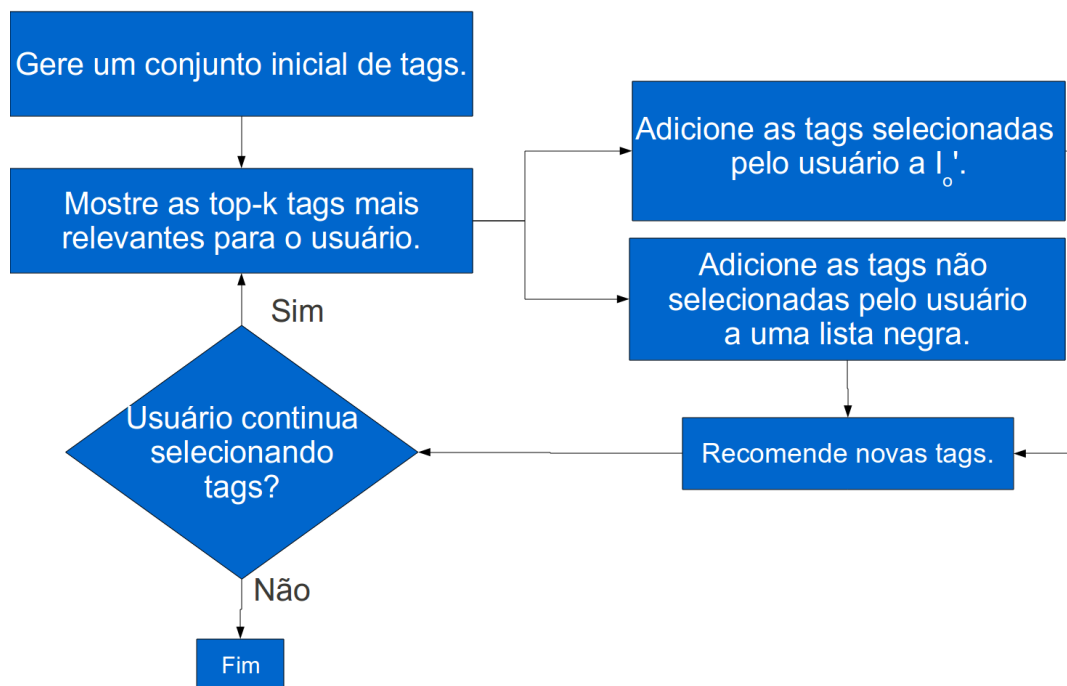


Figura 4.1. Estratégia baseada em RF.

A estratégia proposta é apresentada na Figura 4.1 e detalhada no Algoritmo 3, e consiste nos seguintes passos. Primeiro, é gerado um conjunto inicial de *tags* usando algum método de recomendação, em conjunto ou não com as estratégias de filtragem propostas na seção anterior (linha 4). A seguir, as k_{rf} *tags* mais relevantes (onde a relevância é medida pelo método usado na linha 4) são mostradas para o usuário (linha 7), que seleciona aquelas que ele considera relevantes. As *tags* selecionadas são então usadas para construir \mathcal{I}'_o (linha 8). Tal conjunto é dado como entrada para um método de recomendação associativa que por sua vez, gera novas recomendações (linha 15). Diferentemente de outros métodos de recomendação como o Plurality [Graham

Algoritmo 3 Recomendação baseada em *relevance feedback* do usuário

Entrada: Objeto alvo o , número k_{rf} de *tags* mostradas ao usuário em cada iteração, número máximo M de iterações do processo de RF e o método associativo de recomendação *rec*

Saída: Uma lista de *tags* R_o recomendadas para o objeto o

Seja R_o^i o conjunto das *tags* recomendadas na i^{th} iteração.

```

1:  $R_o = \emptyset$ 
2:  $BlackList_o = \emptyset$ 
3:  $i = 0$ 
4:  $R_o^i =$  gere um conjunto inicial de tags para ser mostrado para o usuário
5: while  $i \leq M$  do
6:    $R_o^i = R_o^i - BlackList_o$ 
7:   Mostre as  $k_{rf}$  tags mais relevantes em  $R_o^i$  para o usuário
8:   Seja  $\mathcal{I}'_o$  as tags selecionadas como relevantes pelo usuário
9:   Seja  $\mathcal{N}_o$  as tags mostradas para o usuário mas não selecionadas por ele
10:   $R_o = R_o \cup \mathcal{I}'_o$ 
11:   $BlackList_o = BlackList_o \cup \mathcal{N}_o$ 
12:  if  $(i \geq 1)$  and  $(R_o^i = R_o^{i-1})$  then break
13:   $\mathcal{I}'_o = R_o$ 
14:   $i++$ 
15:  if  $i \leq M$  then  $R_o^i = rec(o, \mathcal{I}'_o)$ 
16: end while
17: return  $R_o$ 

```

& Caverlee, 2008], a estratégia proposta explora o feedback negativo implícito dado pelo usuário, isto é, *tags* que não foram selecionadas pelo usuário como relevantes são mantidas em uma lista negra (linha 11). *Tags* nessa lista não são consideradas como candidatas para a recomendação (linha 6) e, assim, não são mostradas para o usuário novamente. O processo é repetido até as recomendações convergirem para o mesmo conjunto em iterações consecutivas ou um número máximo de iterações M seja alcançado¹.

Abordagens alternativas podem ser aplicadas tanto para gerar o primeiro conjunto de recomendações mostrados para o usuário (linha 4 do Algoritmo 3) quanto para gerar as recomendações subsequentes (linha 15 do Algoritmo 3). Por exemplo, é possível aplicar simplesmente o mesmo método associativo de recomendação (*rec*) com o conjunto \mathcal{I}_o vazio ou com \mathcal{I}'_o gerado a partir de alguma estratégia de filtragem, como as descritas na Seção 4.1. Em última instância, qualquer método de recomendação que se sai bem em *cold start* pode ser usado. A Figura 4.2 ilustra os seis métodos de recomendação considerados nessa dissertação, obtidos a partir de diferentes escolhas para

¹Na prática, o processo pode ser interrompido pelo usuário quando um resultado satisfatório for obtido e ele parar de fornecer feedback.



Figura 4.2. Variações da estratégia proposta.

os métodos de recomendação usados nas linhas 4 e 15 do Algoritmo 3, a saber: o uso do Sum^+wTS tanto nas linhas 4, quanto 15 (Figura 4.2 (a)); o uso do $LATRE^+wTS$ tanto nas linhas 4, quanto 15 (Figura 4.2 (d)); a aplicação das estratégias de filtragem ao método associativo usado na linha 4 (Figura 4.2 (b) e (e)); o uso de um método de recomendação que se sai bem em *cold start* para gerar as primeiras recomendações (Figura 4.2 (c) e (f)). O ponto dessa estratégia é tentar melhorar a acurácia dos métodos associativos de recomendação em *cold start* fornecendo a eles um bom conjunto

inicial \mathcal{I}'_o de tags, do qual eles podem obter boas *tags* candidatas a partir da geração de padrões de coocorrência. Assim, é importante que algumas das *tags* mostradas para o usuário sejam realmente relevantes. De outro modo, não haverá *feedback* positivo e o método associativo de recomendação não poderá se beneficiar dele. Note que, depois do primeiro *feedback* dado pelo usuário, o conjunto \mathcal{I}'_o passa a ser composto por todas as *tags* marcadas como relevantes pelo usuário em todas as iterações (linha 13). De modo similar, o conjunto final de recomendações consiste em todas as *tags* marcadas como relevantes pelo usuário em todas as iterações (R_o no algoritmo).

O processo de RF diminui o impacto do *cold start* pois ele introduz uma nova fonte de informação (i.e., o usuário), presumivelmente mais segura, aos métodos associativos de recomendação. Comparado ao Plurality [Graham & Caverlee, 2008], que também explora o RF, nesta dissertação tal estratégia é utilizada conjuntamente com métodos estado-da-arte associativos de recomendação que, diferentemente do Plurality, distinguem entre diferentes atributos textuais e levam em conta a diferença de qualidade dos termos extraídos desses atributos para estimar a qualidade de uma tag candidata. Mais ainda, explorando apenas as *tags* marcadas como relevantes pelo usuário, Plurality leva em conta apenas o *feedback* positivo, ignorando o *feedback* negativo implícito que um usuário necessariamente fornece quando ele não seleciona uma *tag* como relevante.

É interessante notar que o problema da assimetria na quantidade de *feedback* positivo e negativo provido pelo usuário, que normalmente existe quando o *feedback* negativo é usado não ocorre no Algoritmo 3. Isso ocorre porque o número k_{rf} de *tags* mostradas ao usuário em cada iteração é pequeno, assim não há grande diferença entre o tamanho do conjunto de *tags* marcadas como relevantes e o conjunto das *tags* não selecionadas pelo usuário.

Note que a estratégia proposta requer pouco esforço extra do usuário, já que tanto o *feedback* positivo quanto o negativo são obtidos de forma implícita a partir das *tags* recomendadas que são ou não selecionadas pelo usuário. A única ação que o usuário precisa tomar é clicar nas *tags* que ele considera relevante, um esforço que ele normalmente tem que fazer para usar qualquer sistema de recomendação. Mais ainda, como será mostrado na Seção 5.6, a convergência do processo é bem rápida. Posto de outro modo, a estratégia proposta é especialmente útil em um cenário em que o usuário deseja atribuir tags com o máximo de qualidade (i.e., relevância) para promover seu objeto no sistema, mas realizando o menor esforço possível.

A próxima seção apresenta um refinamento da estratégia descrita acima baseado na aplicação de programação genética ao problema.

4.3 Explorando o *Feedback* do Usuário pela Aplicação de Programação Genética

Esta dissertação também investiga os benefícios de utilizar técnicas de aprendizado de máquina, mais especificamente programação genética, conjuntamente com o processo de *relevance feedback*. Para tanto, estendemos o arcabouço proposto por [Belém et al., 2011], descrito em mais detalhes na Seção 3.5.5, de modo a incluir tanto uma nova fonte de informação (*relevance feedback*) quanto novas métricas de relevância que sejam mais apropriadas para um cenário onde o objeto alvo da recomendação não possua *tags*. A ideia básica da estratégia proposta é usar a PG para “aprender” uma *boa função de recomendação* para cada iteração do processo de *relevance feedback*. A ideia por trás de tal estratégia é fazer com que o método de recomendação se adapte as diferentes quantidades de informação que estão disponíveis a cada iteração, uma vez que, algumas métricas podem ser melhores quando pouca informação é conhecida e podem não funcionar tão bem quando muita informação está disponível e vice-versa.

De modo análogo a [Belém et al., 2011], uma lista $L_{metrics}$ de métricas de relevância é utilizada. Como no trabalho original, tal lista inclui *Sum*, *IFF*, *Stab*, *TS*, *TF*, *wTS*, *wTF*, *H*, *Sum+*, *Vote* e *Vote+*. Em particular as métricas *Sum* e *Sum+* são usadas com $\ell = 1$. Além dessas métricas, são também incluídas as métricas $S_{Titulo,Tag}$ (Equação 3.19), $S_{Tag,Tag}$ (Equação 3.20), *Pred* (Equação 3.12), e *termScore* (Equação 3.22), que não foram consideradas em [Belém et al., 2011].

A inclusão das novas métricas de relevância leva a um aumento no número de termos candidatos, em relação à proposta original feita em [Belém et al., 2011], o que favorece o método proposto em cenários nos quais ele não pode se valer das *tags* previamente associadas ao objeto alvo de recomendação. As métricas $S_{Titulo,Tag}$, $S_{Tag,Tag}$ e *Pred* em conjunto geram os mesmos candidatos que o método *CTTR*, enquanto que a métrica *termScore* gera os mesmos candidatos que a primeira iteração do *Plurality*. Assim os termos candidatos \mathcal{C}_o de cada objeto alvo o incluem todos os termos gerados pelo *CTTR*, todos os termos extraídos dos outros atributos textuais e todas as *tags* extraídas dos objetos mais semelhantes a o pertencentes a \mathcal{D} . Mais ainda, em cenários nos quais existam *tags* previamente associadas, \mathcal{C}_o ainda contém os candidatos gerados pelo *Sum+*. Para cada candidato $c \in \mathcal{C}_o$, para cada objeto o , são computadas todas as métricas em $L_{metrics}$ usando o conjunto de treino \mathcal{D} , e os atributos textuais associados a o .

O Algoritmo 4 mostra com mais detalhes o processo de treino. Como pode ser visto, um conjunto de validação \mathcal{V} é usado. Nesse conjunto, para cada objeto $v \in \mathcal{V}$ são

Algoritmo 4 *PG + RF*: Processo de Treino

Entrada: Um conjunto de treino \mathcal{D} , um conjunto de validação \mathcal{V} , número k_{rf} de *tags* mostrados ao usuário em cada iteração, número máximo M de iterações do processo de RF, número máximo g de iterações do processo evolucionário do GP.

Saída: Uma lista de funções F de recomendação

```

1:  $i = 0$ 
2: for each  $o \in \mathcal{V}$  do
3:   Seja  $R_o^i$  as tags recomendadas na  $i$ -ésima iteração.
4:    $R_o = \emptyset$ 
5:    $BlackList_o = \emptyset$ 
6:   Calcule o vetor de valores de métricas  $M_{c,o}$  para  $o$  usando  $\mathcal{D}$ 
7: end for
8: while  $i \leq M$  do
9:   Gere uma função de recomendação usando a PG descrita na Figura 3.2 juntamente com o conjunto  $M_c$  de valores das métricas em  $L_{metrics}$ 
10:  Seja  $F_i$  a função gerada pelo treino da PG usando o algoritmo descrito na Figura 3.2
11:  Adicione  $F_i$  a lista de funções  $F$ 
12:  for each  $o \in \mathcal{V}$  do
13:     $R_o^i = F_i(o, M_{c,o})$ 
14:     $R_o^i = R_o^i - BlackList_o$ 
15:    Mostre as  $k_{rf}$  tags mais relevantes em  $R_o^i$  para o usuário
16:    Seja  $\mathcal{T}'_o$  o conjunto das tags selecionadas pelo usuário como relevantes
17:    Seja  $\mathcal{N}_o$  o conjunto das tags mostradas para o usuário mas não selecionadas
18:     $R_o = R_o \cup \mathcal{T}'_o$ 
19:     $BlackList_o = BlackList_o \cup \mathcal{N}_o$ 
20:     $\mathcal{T}'_o = R_o$ 
21:    Recalcule o vetor de valores de métricas  $M_{c,o}$  para  $o$  com base em  $\mathcal{T}'_o$ 
22:  end for
23:   $i++$ 
24: end while
25: return  $F$ 

```

conhecidas as *tags* relevantes que são utilizadas para o cálculo da função de *Fitness*. Mais especificamente, cada candidato c é representado por um vetor de valores de métricas $M_c \in \mathbb{R}^m$, onde m é o número de métricas consideradas. Também atribui-se um rótulo binário Y_c para cada candidato c de cada objeto v no conjunto de validação \mathcal{V} , indicando se c é uma recomendação relevante para v ($Y_c=1$) ou não ($Y_c=0$). Note que aqui o conjunto de treino é usado apenas para extrair regras de associação e computar métricas, enquanto o conjunto de validação \mathcal{V} é o responsável pelo processo de aprendizado das soluções (vide discussão na Seção 5.2). Logo, os rótulos Y_c são atribuídos somente a objetos em \mathcal{V} . Em cada iteração do processo é aplicado o GP para encontrar uma função de recomendação para aquela iteração. Assim ao final do

processo de treino há uma função diferente para cada iteração do RF.

Algoritmo 5 *PG+RF*: Processo de Recomendação.

Entrada: Objeto alvo o , conjunto de treino \mathcal{D} , número k_{rf} de *tags* mostradas ao usuário, número máximo de iterações M do RF, uma lista F de funções geradas pelo Algoritmo 4

Saída: Uma lista de *tags* R_o recomendadas para o

- 1: Seja R_o^i as *tags* recomendadas na i -ésima iteração.
 - 2: $R_o = \emptyset$
 - 3: $\mathcal{I}'_o = \emptyset$
 - 4: $BlackList_o = \emptyset$
 - 5: $i = 0$
 - 6: **while** $i \leq M$ **do**
 - 7: Calcule o vetor de valores de métricas $M_{c,o}$ para o com base em \mathcal{I}'_o e \mathcal{D}
 - 8: $R_o^i = F_i(o, M_{c,o})$
 - 9: $R_o^i = R_o^i - BlackList_o$
 - 10: Mostre as k_{rf} *tags* mais relevantes em R_o^i para o usuário
 - 11: Seja \mathcal{I}'_o o conjunto das *tags* selecionadas pelo usuário como relevantes
 - 12: Seja \mathcal{N}_o o conjunto das *tags* mostradas para o usuário mas não selecionadas
 - 13: $R_o = R_o \cup \mathcal{I}'_o$
 - 14: $BlackList_o = BlackList_o \cup \mathcal{N}_o$
 - 15: **if** $(i \geq 1)$ **and** $(R_o^i = R_o^{i-1})$ **then break**
 - 16: $\mathcal{I}'_o = R_o$
 - 17: $i++$
 - 18: **end while**
 - 19: **return** R_o
-

O Algoritmo 5 mostra como é feita a recomendação usando as funções aprendidas no treino. Inicialmente o vetor com os valores de métricas $M_{c,o}$ para o é calculado com base em \mathcal{I}'_o e \mathcal{D} (linha 7) então a função corresponde a iteração i (F_i) é usada para gerar as recomendações (linha 8), as k_{rf} *tags* mais relevantes são mostradas para o usuário (linha 11), que seleciona aquelas que ele considera relevantes. As *tags* selecionadas são então usadas para construir \mathcal{I}'_o (linhas 13 e 16). Tal conjunto é usado juntamente com \mathcal{D} para recalculer $M_{c,o}$ na iteração seguinte (linha 7).

A Figura 4.3 mostra de uma maneira esquemática o método baseado em PG descrito nos algoritmo 4 e 5. É possível notar que o processo é muito similar ao descrito no Algoritmo 3 a única real diferença é que no Algoritmo 5 as funções usadas para recomendação foram aprendidas de modo automático pelo Algoritmo 4, o que acaba por levar a recomendações de melhor qualidade, como será mostrado na seção 5.6. Devemos destacar, ainda, que o processo de treino do algoritmo *PG + RF* é feito inteiramente offline (Figura 4.3 (a)) e que em tempo de execução são necessárias apenas as funções aprendidas no treino (Figura 4.3 (b)).

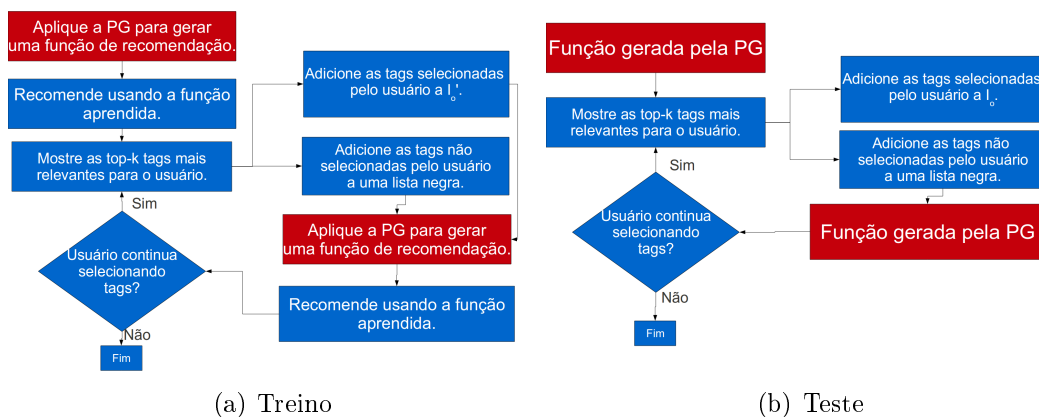


Figura 4.3. *PG + RF*: treino x execução.

Por fim é interessante destacar a flexibilidade do arcabouço proposto uma vez que ele é capaz de acoplar diferentes funções permitindo que o sistema de recomendação possa se adaptar melhor a diferentes cenários.

4.4 Sumário dos Métodos de Recomendação

Métodos de Referência	
Acrônimo	Equação/Descrição
Sum^+wTS	<p>Estima a relevância de um termo candidato c pela combinação linear da saída das métricas Sum^+ e wTS, que leva em conta o poder descritivo de c. Ou seja:</p> $Sum^+wTS(c, o, k_x, k_c, k_r, \alpha) = \alpha Sum^+(c, o, k_x, k_c, k_r) + (1 - \alpha)wTS(c, o) \quad (4.1)$
$LATRE^+wTS$	<p>Estima a relevância de um termo candidato c pela combinação linear da saída das métricas $LATRE$ e wTS, que leva em conta o poder descritivo de c. Ou seja:</p> $LATRE^+wTS(c, o, \ell, \alpha) = \alpha Sum(c, o, \ell) + (1 - \alpha)wTS(c, o) \quad (4.2)$
$CTTR$	Explora termos extraídos de outros atributos textuais e uma métrica de relevância de <i>tags</i> , mas não considera <i>tags</i> previamente associadas ao objeto, e assim não sofre os efeitos de <i>cold start</i> .
Plurality	Explora <i>relevance feedback</i> obtido do usuário e não utiliza <i>tags</i> previamente associadas ao objeto alvo da recomendação. Começando com algumas sugestões iniciais obtidas a partir de um filtro colaborativo, Plurality iterativamente as refina pela incorporação do feedback provido pelo usuário.

Extensões Propostas	
Acrônimo	Equação/Descrição
$PG + RF$	Combina o arcabouço de programação genética com o arcabouço de <i>relevance feedback</i> . Para tanto, estende o arcabouço proposto por [Belém et al., 2011], descrito em mais detalhes na Seção 3.5.5, de modo a incluir tanto uma nova fonte de informação (<i>relevance feedback</i>) quanto novas métricas de relevância que sejam mais apropriadas para um cenário onde o objeto alvo da recomendação não possua <i>tags</i> .
Sum^+wTS	Estende o método Sum^+wTS para incluir o uso do <i>relevance feedback</i> como mostrado na Figura 4.2 (a).
$CF + Sum^+wTS$	Estende o método Sum^+wTS para incluir o uso do <i>relevance feedback</i> como mostrado na Figura 4.2 (b).
$CTTR + Sum^+wTS$	Estende o método Sum^+wTS para incluir o uso do <i>relevance feedback</i> como mostrado na Figura 4.2 (c).
$LATRE^+wTS$	Estende o método $LATRE^+wTS$ para incluir o uso do <i>relevance feedback</i> como mostrado na Figura 4.2 (d).
$CF + LATRE^+wTS$	Estende o método $LATRE^+wTS$ para incluir o uso do <i>relevance feedback</i> como mostrado na Figura 4.2 (e).
$CTTR + LATRE^+wTS$	Estende o método $LATRE^+wTS$ para incluir o uso do <i>relevance feedback</i> como mostrado na Figura 4.2 (f).

Capítulo 5

Avaliação Experimental

Este capítulo apresenta primeiramente os conjuntos de dados utilizados para avaliar as estratégias de recomendação (Seção 5.1), bem como a metodologia de avaliação utilizada (Seção 5.2). A seguir é descrito o procedimento adotado para realizar o ajuste dos parâmetros dos métodos de recomendação (Seção 5.3). Já os principais resultados são apresentados nas Seções 5.4-5.6

5.1 Bases de Dados

Os métodos de recomendação de *tags* foram avaliados em quatro bases de dados, cada uma contendo *título*, *tags* e *descrição* associados a objetos coletados das aplicações Bibsonomy¹, LastFM, YouTube e YahooVideo. As três últimas bases são as mesmas usadas em [Belém et al., 2011] e incluem os atributos textuais associados a 2,8 milhões de artistas do LastFM, a mais de 9 milhões de vídeos do YouTube e a mais de 160 mil vídeos do Yahoo Video. Já os dados do Bibsonomy estão disponíveis publicamente² e são os mesmos dados utilizados em estudos recentes [Lipczak et al., 2009].

São considerados apenas objetos com atributos textuais em língua inglesa, sendo utilizado o algoritmo de Porter³ para remoção de afixos em cada palavra de cada atributo coletado. A remoção de afixos foi feita para evitar recomendações triviais como plurais e outras pequenas variações de uma mesma palavra. Também foram removidas *stopwords*, bem como termos que são ou muito frequentes (com mais de 100,000 ocorrências na coleção de dados) ou muito raros (com menos de 30 ocorrências),

¹Note que o Bibsonomy possui outros atributos textuais, entretanto neste trabalho consideramos apenas o título a descrição e as *tags* associados aos seus objetos.

²Bibsonomy: <http://www.kde.cs.uni-kassel.de/bibsonomy/dumps> acessado em 01/03/2013

³<http://tartarus.org/~martin/PorterStemmer/> acessado em 01/03/2013

Tabela 5.1. Descrição geral das bases de dados.

Base de dados	# tags/objeto	# medio de tags/objeto	# objetos
Bibsonomy	2-120	4.9	483,932
Last.FM	2-507	7.8	235,492
YahooVideo	2-52	10.7	146,868
YouTube	2-77	8.9	5.5 million

visto que tais termos raramente constituem boas recomendações [Sigurbjörnsson & van Zwol, 2008].

A Tabela 5.1 resume as bases de dados, mostrando o número de objetos e o intervalo para o número de *tags* por objeto em cada base.

5.2 Metodologia de Avaliação

De forma similar a [Belém et al., 2011; Garg & Weber, 2008; Rendle & Lars, 2010; Menezes et al., 2010; Guan et al., 2009], foi adotada uma metodologia de avaliação automática: um subconjunto das *tags* associadas a cada objeto de teste foi usado como gabarito para a recomendação, ou seja, como *tags* relevantes para aquele objeto. Como tal, tais *tags* foram desconsideradas para o cálculo de métricas de relevância. Esta metodologia foi adotada porque o processo de avaliação manual é caro e pode ser afetado pela subjetividade dos julgamentos humanos. É importante destacar que os resultados obtidos de acordo com a metodologia proposta representam limites inferiores, já que algumas das *tags* recomendadas podem não aparecer no gabarito mas serem de fato relevantes para o objeto.

Nossa avaliação foi feita sobre amostras de 150,000 objetos do YouTube, Last.FM e Bibsonomy e 120,000 objetos do YahooVideo. Cada amostra foi dividida em 5 partes de igual tamanho, que são utilizadas num procedimento experimental de validação cruzada. Ou seja, três partes são usadas como conjunto de treinamento \mathcal{D} , que por sua vez é usado para a extração das regras de associação e computação das outras métricas, uma parte é usada para a validação e ajuste de parâmetros, e a última parte é usada como o conjunto teste \mathcal{O} . Este processo é repetido cinco vezes, utilizando-se como conjunto de teste uma parte diferente a cada vez. Resultados apresentados são médias dos cinco conjuntos de testes, juntamente com os intervalos de confiança de 95% correspondes.

Seguindo a metodologia proposta, para cada objeto $o = \langle \mathcal{I}_o, \mathcal{F}_o, \mathcal{Y}_o \rangle$ nos conjuntos de validação e de teste, metade de suas *tags* são selecionadas aleatoriamente e incluídas em \mathcal{I}_o . A outra metade é incluída em \mathcal{Y}_o , o gabarito de o . Para avaliação dos métodos

em *cold start*, *tags* em \mathcal{I}_o foram ignoradas. Isso foi feito, ao invés de utilizá-las como parte da resposta esperada, de modo a manter o mesmo gabarito para os cenários com e sem *cold start*, possibilitando assim a comparação entre os mesmos. Também foram utilizados título e descrição como atributos textuais em \mathcal{F}_o , para cada objeto o .

Como em [Ferreira et al., 2011], o gabarito \mathcal{Y}_o foi usado para simular o *feedback* dado pelo usuário, usado tanto pela estratégias baseadas em RF proposta quanto pelo Plurality. Em outras palavras em cada iteração do processo de RF, as *tags* recomendadas consideradas como relevantes (e.g., linha 8 no Algoritmo 3), são aquelas que estão em \mathcal{Y}_o . Inicialmente é simulado um usuário ideal, isto é um usuário que seleciona todas as *tags* relevantes mostradas a ele a cada iteração (*tags* que estão em \mathcal{Y}_o), posteriormente esta premissa é relaxada, de modo a avaliar a sensibilidade das estratégias baseadas em *relevance feedback* ao nível de cooperação e confiabilidade do usuário. Ressaltamos que as *tags* do conjunto \mathcal{Y}_o que não foram “selecionados pelo usuário” não são utilizadas no cálculo de métricas de relevância de *tags* nem exploradas, de forma alguma, pelo recomendador. A única informação adicional que o gabarito fornece para as estratégias baseadas em RF é se cada tag t recomendada na iteração atual é relevante ou não. Esta metodologia simula o *feedback* real dado pelos usuários e não afeta ou distorce nossa avaliação automática.

Como mencionado na Seção 4.3, o processo de validação cruzada para a abordagem baseada em programação genética é ligeiramente diferente do tradicional, já que o aprendizado da função de valoração e a seleção de parâmetros são feitas sobre o conjunto de validação \mathcal{V} . Isso foi feito para se evitar *overfitting*, que pode ocorrer se as soluções forem aprendidas no mesmo conjunto que foi utilizado para a geração das regras de associação, isto é, no conjunto de treino \mathcal{D} , pois nesse caso poderia haver uma superestimativa das métricas derivadas das regras de associação.

O projeto experimental proposto aqui é justo porque: (1) nenhuma informação privilegiada do conjunto de teste (para o qual os resultados são reportados) é utilizada; (2) todos os parâmetros são ajustados em um mesmo conjunto de validação; (3) os intervalos de confiança reportados em nossos experimentos são evidências da pouca variabilidade e convergência tanto do GP quanto das estratégias baseadas em RF.

Nesta dissertação foram usadas três métricas amplamente utilizadas na literatura [Baeza-Yates & Ribeiro-Neto, 2011] para avaliar a qualidade das recomendações a saber $P@k$, revocação e a precisão média. Uma delas ($P@k$) também foi utilizada pelo arcabouço de Programação Genética (seções 3.5.5 e 4.3), para o cálculo da *Fitness*.

A métrica $P@k$ (Eq. 5.1) é a principal métrica utilizada nessa dissertação para avaliar as recomendações. Ela mede a fração das *tags* relevantes entre as primeiras k *tags* recomendadas. Em outras palavras, seja \mathcal{Y}_o o conjunto de *tags* relevantes do

objeto o , \mathcal{C}_o as recomendações (ordenadas) produzidas pelo método avaliado e \mathcal{C}_o^k os primeiros k elementos de \mathcal{C}_o . $P@k$ é definida como:

$$P@k(\mathcal{C}_o, \mathcal{Y}_o) = \frac{|\mathcal{C}_o^k \cap \mathcal{Y}_o|}{\min(k, |\mathcal{Y}_o|)} \quad (5.1)$$

Considera-se aqui $P@k$ com $k = 5$, também foram medidas a revocação e a precisão média (*average precision* ou *AP*). Revocação é a fração do conjunto de *tags* relevantes para um objeto que foram recomendadas, enquanto o *AP* considera a ordem em que as *tags* foram recomendadas, enfatizando as recomendações no topo da ordenação [Baeza-Yates & Ribeiro-Neto, 2011]:

$$Revoc(\mathcal{C}_o, \mathcal{Y}_o) = \frac{|\mathcal{C}_o \cap \mathcal{Y}_o|}{|\mathcal{Y}_o|} \quad (5.2)$$

$$AP(\mathcal{C}_o, \mathcal{Y}_o) = \frac{1}{|\mathcal{Y}_o|} \sum_{k=1}^{|\mathcal{C}_o|} (P@k(\mathcal{C}_o, \mathcal{Y}_o) \times rel(k)) \quad (5.3)$$

onde $rel(k) = 1$ se o k -ésimo candidato retornado em \mathcal{C} é relevante e $rel(k) = 0$ caso contrário.

5.3 Parametrização

A avaliação experimental é iniciada pelo ajuste dos parâmetros dos métodos avaliados. Para isso é usado o conjunto de validação \mathcal{V} e partindo da premissa que os métodos tenham todas as fontes de informação disponíveis (cenário sem *cold start*) uma vez que alguns métodos exploram mais de uma fonte de informação e usam parâmetros (e.g., α no *Sum⁺wTS*) para ponderar as mesmas.

A eficácia do método *Sum⁺wTS* foi avaliada em função dos parâmetros k_r , k_x e k_c . Para tanto o valor de α foi fixado em 1 já que k_r , k_x e k_c influenciam apenas a primeira parte do método (*Sum⁺*). Eles foram variados sequencialmente nos valores 1, 5, 10, 20, 50, sendo os melhores resultados obtidos para $k_r = k_x = k_c = 5$, nas quatro bases de dados.

O valor do parâmetro ℓ , que controla o tamanho das regras de associação geradas pelo *LATRE⁺wTS*, foi fixado com 3 em todas as bases como feito em [Belém et al., 2011].

Os parâmetros θ_{min} e σ_{min} impactam diretamente o número de regras de associação geradas pelo *Sum⁺wTS* e *LATRE⁺wTS* e conseqüentemente o tempo de processamento dos mesmos. Assim é necessário buscar um bom compromisso entre tempo

Tabela 5.2. Valores médios de P@5 para o Sum^+wTS e $LATRE^+wTS$ em função da variação dos parâmetros θ_{min} e σ_{min} . Em negrito resultados com melhor compromisso entre eficiência e precisão.

σ_{min}	θ_{min}	Bibsonomy		Last.FM		YahooVideo		YouTube	
		Sum^+wTS	$LATRE^+wTS$	Sum^+wTS	$LATRE^+wTS$	Sum^+wTS	$LATRE^+wTS$	Sum^+wTS	$LATRE^+wTS$
1	0.00	0.356	0.359	0.414	0.408	0.493	0.618	0.250	0.291
1	0.01	0.356	0.359	0.414	0.408	0.493	0.618	0.250	0.291
1	0.10	0.346	0.349	0.413	0.407	0.491	0.615	0.245	0.285
1	0.20	0.309	0.312	0.411	0.405	0.485	0.608	0.228	0.265
1	0.30	0.268	0.271	0.384	0.378	0.471	0.590	0.206	0.239
1	0.40	0.232	0.234	0.341	0.336	0.445	0.558	0.184	0.214
1	0.50	0.205	0.207	0.260	0.256	0.414	0.519	0.176	0.437
2	0.00	0.345	0.348	0.414	0.408	0.493	0.618	0.241	0.281
2	0.01	0.345	0.348	0.414	0.408	0.493	0.617	0.241	0.281
2	0.10	0.335	0.338	0.414	0.408	0.491	0.615	0.236	0.275
2	0.20	0.297	0.300	0.411	0.405	0.485	0.607	0.217	0.252
2	0.30	0.256	0.258	0.384	0.378	0.470	0.590	0.193	0.224
2	0.40	0.221	0.223	0.341	0.336	0.445	0.558	0.170	0.198
2	0.50	0.194	0.195	0.292	0.287	0.414	0.518	0.149	0.173

de execução e precisão das recomendações. Quanto menor o valor de θ_{min} e/ou σ_{min} , maior o número de regras e portanto maior o tempo de processamento. Os resultados de experimentos com diferentes valores de θ_{min} e σ_{min} são mostrados na Tabela 5.2, novamente, o valor de α foi fixado em 1 para estes experimentos. Os resultados indicam que, na maioria dos casos, a precisão decresce com o aumento desses dois limiares, uma vez que menos regras de associação são geradas com tal aumento. Assim os valores de θ_{min} e σ_{min} foram escolhidos de modo que a perda de precisão em relação a $\theta_{min} = \sigma_{min} = 0$ fosse menor que 3%. Os valores escolhidos para cada base são mostrados em negrito na tabela.

O impacto do parâmetro α , que pondera cada uma das métricas dos métodos Sum^+wTS e $LATRE^+wTS$ é mostrado na Tabela 5.3. Observe que valores intermediários de α levam a recomendações melhores, beneficiando-se do uso combinado de regras de associação e de outros atributos textuais. Observe ainda que o peso dado aos candidatos gerados a partir do uso de regras de associação é maior que o dado aos candidatos extraídos dos outros atributos textuais do objeto.

Tabela 5.3. P@5 média para o Sum^+wTS e $LATRE^+wTS$ em função da variação do parâmetro α . Em negrito melhores resultados.

α	Bibsonomy		Last.FM		YahooVideo		YouTube	
	Sum^+wTS	$LATRE^+wTS$	Sum^+wTS	$LATRE^+wTS$	Sum^+wTS	$LATRE^+wTS$	Sum^+wTS	$LATRE^+wTS$
0.00	0.277	0.264	0.111	0.117	0.492	0.492	0.400	0.397
0.50	0.360	0.376	0.242	0.244	0.621	0.600	0.464	0.463
0.60	0.374	0.387	0.264	0.260	0.658	0.614	0.477	0.466
0.70	0.402	0.403	0.307	0.285	0.690	0.647	0.494	0.472
0.80	0.434	0.422	0.374	0.321	0.707	0.695	0.502	0.482
0.90	0.443	0.438	0.409	0.370	0.662	0.732	0.456	0.488
0.95	0.416	0.435	0.417	0.401	0.573	0.697	0.368	0.454
0.99	0.380	0.413	0.415	0.411	0.509	0.634	0.284	0.335
1.00	0.369	0.399	0.413	0.406	0.496	0.616	0.271	0.304

Tabela 5.4. Parametrização dos Algoritmos de Recomendação de *Tags*.

Parametro	Bibsonomy	Last.FM	YahooVideo	YouTube
σ_{min}	1	2	2	1
θ_{min}	0.1	0.2	0.2	0.1
$\alpha(Sum^+wTS)$	0.9	0.95	0.8	0.8
$\alpha(LATRE^+wTS)$	0.9	0.99	0.9	0.9
$\ell(LATRE^+wTS)$	3	3	3	3
k_r	5	5	5	5
k_x	5	5	5	5
k_c	5	5	5	5
k_s	5	5	5	5
k_{filter}	1	10	5	3
n_{tags}	5	5	5	5
k_{rf}	5	5	5	5
n_{obj}	10	10	10	10

Para a estratégia baseada em programação genética, foram conduzidos experimentos no conjunto de validação \mathcal{V} . Inicialmente foram fixados o número de indivíduos usados na seleção por torneio $k = 2$, a profundidade máxima da árvore que representa um indivíduo $d = 7$, a probabilidade de cruzamento $p_c = 0.6$ e a probabilidade de mutação $p_m = 0.1$, valores frequentemente usados na literatura [Banzhaf et al., 1998]. O tamanho n da população foi variado no intervalo $[50, 200]$, sendo selecionado o valor 200, visto que uma população mais ampla permite uma melhor cobertura do espaço de busca. Para esse tamanho de população, o algoritmo convergiu antes de 200 iterações, valor então fixado como o número máximo g de iterações. Computar a *Fitness* durante o processo evolucionário sobre todos os objetos do conjunto de validação é proibitivo, uma vez que as bases de dados são razoavelmente grandes e o processo de computação da *Fitness* razoavelmente caro, sendo assim a *Fitness* foi computada sobre uma amostra variável S contendo 500 objetos, isto é, a cada iteração são escolhidos aleatoriamente S objetos do conjunto de validação sobre os quais a *Fitness* é então computada naquela iteração.

Para as estratégias de filtragem, o parâmetro k_{filter} foi testado para valores no intervalo $[1,20]$, foram escolhidos os valores 3, 10, 5 e 1, para o YouTube, Last.FM, YahooVideo e Bibsonomy, respectivamente, já que eles levaram aos melhores resultados sobre o conjunto de validação para a maioria das heurísticas testadas.

Tanto para o Plurality quanto para as novas estratégias baseadas em RF, foi escolhido o valor 5 para n_{tags} e k_{rf} , por consistência com a principal métrica de avaliação adotada, $P@5$. Além disso, como sugerido pelos autores do Plurality, foi atribuído $n_{obj} = 10$ [Graham & Caverlee, 2008].

A Tabela 5.4 sumariza os valores escolhidos para os parâmetros dos métodos de referência bem como para as estratégias de filtragem.

Nas próximas seções é feita uma avaliação do impacto do *cold start* sobre os métodos de referência (Seção 5.4), bem como das estratégias de filtragem propostas (Seção 5.5) e das estratégias baseadas em RF (Seção 5.6).

5.4 Impacto do *Cold Start* nos Métodos Associativos

Esta Seção mostra o impacto do *cold start* sobre os métodos associativos de recomendação Sum^+wTS e $LATRE^+wTS$, que exploram padrões de coocorrência com *tags* previamente associadas ao objeto o , bem como termos extraídos de outros atributos textuais. A eficácia destes métodos, em *cold start*, é comparada com a do CTTR, que explora somente termos extraídos de outros atributos textuais. Objetiva-se com esta comparação mostrar a importância de cada fonte de informação *tags* e outros atributos textuais disponível para os métodos de recomendação.

Tabela 5.5. Importância das diferentes fontes de informação para a eficácia dos métodos associativos Sum^+wTS e $LATRE^+wTS$. Valores médios de $P@5$ juntamente com intervalos de confiança de 95%.

Métodos originais				
Estratégia	Bibsonomy	Last.FM	YahooVideo	YouTube
<i>CTTR</i>	0.275 ± 0.002	0.260 ± 0.001	0.465 ± 0.004	0.376 ± 0.002
Sum^+wTS	0.436 ± 0.002	0.417 ± 0.002	0.707 ± 0.002	0.502 ± 0.003
$LATRE^+wTS$	0.435 ± 0.002	0.411 ± 0.001	0.733 ± 0.003	0.489 ± 0.003
Métodos associativos sem considerar \mathcal{F}_o				
Estratégia	Bibsonomy	Last.FM	YahooVideo	YouTube
Sum^+	0.326 ± 0.002	0.411 ± 0.001	0.484 ± 0.003	0.245 ± 0.002
<i>LATRE</i>	0.349 ± 0.002	0.405 ± 0.001	0.608 ± 0.003	0.285 ± 0.004
Métodos associativos em cenário de <i>cold start</i>				
Estratégia	Bibsonomy	Last.FM	YahooVideo	YouTube
Sum^+wTS	0.207 ± 0.002	0.065 ± 0.001	0.419 ± 0.004	0.359 ± 0.002
$LATRE^+wTS$				

A Tabela 5.5 mostra os valores médios de $P@5$ juntamente com intervalos de confiança de 95% para cada uma das bases de dados, em três diferentes cenários. O primeiro consiste nos métodos de referência como definidos originalmente, com todas as fontes de informação que eles exploram. O segundo cenário consiste nos métodos associativos de recomendação (Sum^+wTS e $LATRE^+wTS$) explorando apenas padrões

de coocorrência com *tags* do objeto alvo o (cenário no qual os dois métodos convergem para suas formas originais Sum^+ [Sigurbjörnsson & van Zwol, 2008] e $LATRE$ [Menezes et al., 2010], respectivamente). O último cenário consiste nos dois métodos na ausência de *tags* previamente associadas (*cold start*), que é simulado ignorando as *tags* em \mathcal{I}_o (i.e., $\mathcal{I}_o = \emptyset$). Os resultados dos diferentes cenários são mostrados em diferentes blocos na Tabela 5.5. Como o CTTR não explora *tags* previamente associadas (i.e., o CTTR não usa as *tags* presentes em \mathcal{I}_o), seus resultados são os mesmos com e sem *cold start*.

Observa-se que, consistentemente com os resultados obtidos em [Belém et al., 2011], tanto o Sum^+wTS quanto o $LATRE^+wTS$ superam as outras estratégias (incluindo o CTTR) em todas as bases na ausência de *cold start*. Entretanto, em *cold start*, a precisão de ambos os métodos (terceiro bloco da Tabela 5.5) é bastante prejudicada. Em tal cenário, as perdas do Sum^+wTS em $P@5$ médio são de 47% no Bibsonomy, 41% no YahooVideo e 28% no YouTube. Perdas similares são observadas para o $LATRE^+wTS$ nas três bases. O *cold start* é ainda mais danoso a ambos os métodos no Last.FM, causando uma perda em $P@5$ médio de 84%. A grande perda de precisão nesta base é possivelmente explicada pelo fato que a sobreposição entre os outros atributos textuais e *tags* tende a ser menor [Figueiredo et al., 2013], o que acaba por causar uma grande concentração em valores de TS , e por conseguinte de wTS , baixos. Esta concentração faz com que seja difícil distinguir termos “bons” de “ruins” usando apenas estas métricas.

Como mencionado anteriormente, na ausência de *tags* no objeto alvo o (*cold start*), tanto o Sum^+wTS quanto o $LATRE^+wTS$ são capazes de explorar como candidatos a recomendação apenas termos extraídos dos outros atributos textuais do objeto o (através da aplicação da métrica WTS sobre tais termos), isso limita o seu potencial de descobrir novos termos candidatos o que em última instância causa a perda de precisão observada na Tabela 5.5. O impacto do *cold start* sobre os métodos de recomendação associativa é tão forte que, nesse cenário, eles são superados pelo CTTR, que, por sua vez, não é afetado pela falta de *tags*. Isso acontece em todas as bases de dados, sendo que no Last.FM a precisão do CTTR passa a ser mais de 300% superior a dos outros métodos. Este resultado é completamente divergente daquele reportado em [Belém et al., 2011], e também observado aqui quando os métodos são aplicados a objetos que já contém algumas *tags* ($\mathcal{I}_o \neq \emptyset$).

A Tabela 5.5 também mostra que, em geral, coocorrência com *tags* previamente associadas ao objeto alvo representam uma fonte de informação mais importante para suportar recomendações mais efetivas que termos extraídos de outro atributos textuais tanto para o Sum^+wTS quanto para o $LATRE^+wTS$. Isso pode ser visto pela comparação dos resultados do Sum^+ e $LATRE$ com o resultado apresentado no terceiro

bloco da tabela. No Last.FM, em particular, ambos os métodos se beneficiam muito do uso dos padrões de coocorrência. De fato, nesta aplicação, tanto o Sum^+ quanto o $LATRE$ superam o CTTR conseguindo ganhos de até 160% em $P@5$ média sobre ele. A única exceção fica por conta da coleção do YouTube, para a qual o uso termos extraídos de outro atributos textuais (terceiro bloco da Tabela 5.5) é 46% e 25% superior ao Sum^+ e $LATRE$, respectivamente. Isso pode ser explicado pelo fato de que, diferentemente do YahooVideo, Last.FM e Bibsonomy para as quais as *tags* são atribuídas de forma colaborativa e assim têm um vocabulário mais diversificado, no YouTube, *tags* podem ser associadas apenas pelo criador do vídeo. Logo diferentes atributos textuais associados a um mesmo objeto (incluindo *tags*) tendem a ter mais termos em comum no YouTube [Figueiredo et al., 2013], o que é explorado pelo wTS para melhor discriminar entre diferentes termos.

Conclusões muito similares são obtidas quando se consideram a revocação ou o AP . Por questões de apresentação tais resultados são mostrados no Apêndice A.

Resumindo, a despeito da eficácia superior em objetos contendo algumas tags, tanto o Sum^+wTS quanto o $LATRE^+wTS$ têm grandes perdas de performance no cenário de *cold start*. As próximas seções focam nesse cenário mostrando os resultados das várias estratégias propostas para aliviar seu impacto e melhorar a eficácia dos métodos associativos de recomendação.

5.5 Eficácia das Estratégias de Filtragem

Os resultados mostrados na Seção 5.4 motivam a procura por estratégias para melhorar a eficácia dos métodos associativos de recomendação na presença de *cold start*. Nesta dissertação, foram propostos métodos automáticos e métodos assistidos pelo usuário para gerar um conjunto alternativo \mathcal{I}'_o sobre o qual padrões de coocorrência possam ser obtidos. Esta Seção discute os resultados obtidos com as estratégias de filtragem automática propostas e discutidas na Seção 4.1. Resultados das estratégias assistidas pelo usuário são discutidos na próxima Seção.

A Tabela 5.6 mostra os resultados para cada uma das abordagens mencionadas na Seção 4.1, ou seja: (1) uso de outro recomendador - $CTTR$ - que tem um desempenho razoável em *cold start* para gerar \mathcal{I}'_o ; (2) filtragem de termos de outros atributos textuais do objeto usando uma das heurísticas discutidas na Seção 3.5.1 como critério de filtragem; (3) uso de um filtro colaborativo (CF) para extrair *tags* de objetos similares na base de treino. A tabela mostra os resultados obtidos pelo uso de cada uma das abordagens em conjunto com o Sum^+wTS e $LATRE^+wTS$. Os melhores

Tabela 5.6. Estratégias de filtragem automáticas para construir \mathcal{I}'_o : P@5 média juntamente com intervalos de confiança de 95%.

Métodos associativos em cenário de <i>cold start</i>				
Estratégias de filtragem aplicadas ao Sum^+wTS (\mathcal{I}'_o gerado a partir de \mathcal{F}_o)				
Estratégia	Bibsonomy	Last.FM	Yahoo Video	YouTube
<i>TF</i>	0.220 ± 0.002	0.151 ± 0.002	0.444 ± 0.002	0.369 ± 0.001
<i>TS</i>	0.219 ± 0.003	0.156 ± 0.002	0.446 ± 0.003	0.370 ± 0.002
<i>wTF</i>	0.220 ± 0.003	0.156 ± 0.002	0.451 ± 0.003	0.373 ± 0.002
<i>wTS</i>	0.220 ± 0.002	0.151 ± 0.002	0.452 ± 0.003	0.373 ± 0.002
<i>Cov</i>	0.217 ± 0.003	0.139 ± 0.002	0.419 ± 0.004	0.363 ± 0.001
<i>Pred</i>	0.238 ± 0.002	0.192 ± 0.002	0.465 ± 0.003	0.379 ± 0.003
Estratégias de filtragem aplicadas ao $LATRE^+wTS$ (\mathcal{I}'_o gerado a partir de \mathcal{F}_o)				
Estratégia	Bibsonomy	Last.FM	Yahoo Video	YouTube
<i>TF</i>	0.133 ± 0.001	0.133 ± 0.003	0.431 ± 0.002	0.325 ± 0.001
<i>TS</i>	0.135 ± 0.003	0.134 ± 0.002	0.433 ± 0.002	0.325 ± 0.001
<i>wTF</i>	0.133 ± 0.003	0.134 ± 0.002	0.429 ± 0.002	0.330 ± 0.002
<i>wTS</i>	0.135 ± 0.002	0.132 ± 0.002	0.426 ± 0.002	0.331 ± 0.001
<i>Cov</i>	0.124 ± 0.002	0.105 ± 0.003	0.308 ± 0.005	0.214 ± 0.001
<i>Pred</i>	0.143 ± 0.002	0.130 ± 0.001	0.423 ± 0.002	0.338 ± 0.002
\mathcal{I}'_o gerado pelo uso de um filtro colaborativo (<i>CF</i>)				
Estratégia	Bibsonomy	Last.FM	Yahoo Video	YouTube
Sum^+wTS	0.241 ± 0.002	0.240 ± 0.002	0.486 ± 0.003	0.386 ± 0.002
$LATRE^+wTS$	0.193 ± 0.002	0.079 ± 0.001	0.389 ± 0.001	0.319 ± 0.002
\mathcal{I}'_o gerado pelo uso de um outro recomendador (<i>CTTR</i>)				
Estratégia	Bibsonomy	Last.FM	Yahoo Video	YouTube
Sum^+wTS	0.222 ± 0.002	0.211 ± 0.002	0.454 ± 0.002	0.379 ± 0.003
$LATRE^+wTS$	0.172 ± 0.002	0.064 ± 0.001	0.388 ± 0.003	0.334 ± 0.001

resultados para cada base de dados (incluindo empates estatísticos de acordo com um teste-t pareado com 95% de confiança) são mostrados em negrito.

Comparando os resultados das Tabelas 5.5 e 5.6, é possível observar que, a despeito da melhoria na eficácia do Sum^+wTS em *cold start*, qualquer uma das abordagens consideradas, quando usadas em conjunto com o Sum^+wTS apenas produziram uma melhora marginal (3-4%) sobre o *CTTR*. Mais ainda, os resultados produzidos pelo $LATRE^+wTS$ combinado com as estratégias propostas são claramente piores que os do *CTTR*, e em alguns casos, piores até que o método $LATRE^+wTS$ original sem o uso de *tags* iniciais. Em outras palavras, o conjunto \mathcal{I}'_o gerado pelas estratégias de filtragem chega a ser prejudicial para o $LATRE^+wTS$ em algumas bases. Isso ocorre pois \mathcal{I}'_o pode conter muitos termos ruins (ou irrelevantes) que fazem com que mais ruído seja considerado como candidato à recomendação. Assim é importante construir um con-

junto \mathcal{I}'_o sem esses termos ruins (irrelevantes). O Sum^+wTS é mais robusto a termos potencialmente ruins pois ele usa a métrica $Stab$ (Eq. 3.3), que penaliza termos muito comuns e também termos muito raros, diminuindo assim o impacto de uma escolha pobre de termos para \mathcal{I}'_o .

Resultados similares aos da Tabela 5.6 são obtidos quando se consideram revocação e AP , mostrando assim o potencial limitado do uso de estratégias de filtragem. Tais resultados são apresentados no Apêndice A. De todo modo, observe que CF e $Pred$ foram as estratégias que apresentaram os melhores resultados.

Uma vez que as estratégias automáticas para construir \mathcal{I}'_o não conseguiram ganhos significativos, a próxima Seção apresenta os resultados do uso do *relevance feedback* como uma alternativa para gerar um bom conjunto de entrada \mathcal{I}'_o para os métodos associativos.

5.6 Eficácia das Estratégias Assistidas pelo Usuário

Esta Seção discute os resultados das estratégias baseadas em RF propostas, comparando-as com os métodos originais sobre os quais elas foram construídas e também com o $CTTR$ e o *Plurality* (que também explora RF). São consideradas três opções para gerar o primeiro conjunto de *tags* que será apresentado para o usuário pela estratégia proposta descrita no Algoritmo 3 (linha 4): (1) aplicar o método associativo original com um conjunto \mathcal{I}'_o vazio, (2) aplicar o método associativo de recomendação original com \mathcal{I}'_o preenchido usando a estratégia de filtragem mais promissora encontrada na Seção 5.5⁴ (CF) e (3) usar o $CTTR$ para gerar o primeiro conjunto de recomendações⁵ e então aplicar o método associativo nas iterações seguintes. Para distinguir entre essas abordagens, a segunda é chamada de $CF+method$ e a terceira é apresentada como $CTTR+method$, onde *method* pode ser Sum^+wTS ou $LATRE^+wTS$. Também é considerada a estratégia híbrida que combina o uso de programação genética (PG) com RF, chamada de $PG+RF$.

⁴Note que, mesmo que as estratégias de filtragem apresentem ganhos limitados é interessante comparar o ganho obtido pelo uso conjunto delas com o *relevance feedback*

⁵O uso do $CTTR$ é motivado pela sua boa performance em *cold start*, como visto nas seções anteriores e também para mostrar a flexibilidade do arcabouço proposto.

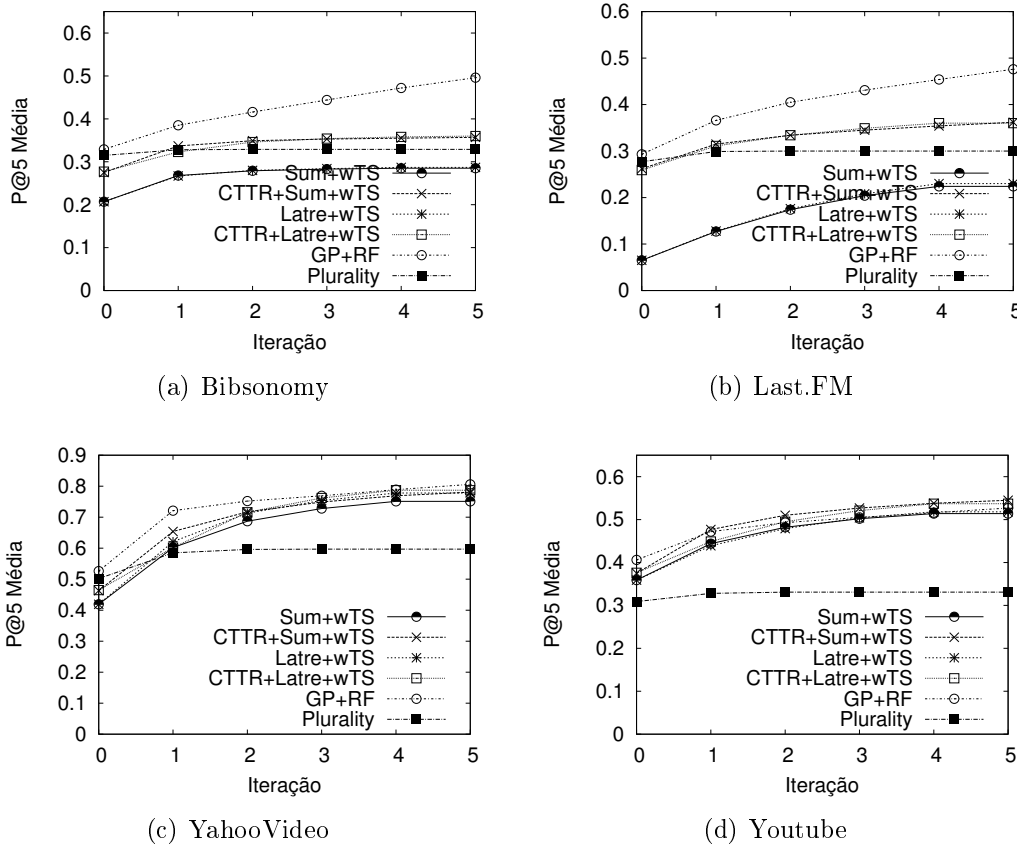


Figura 5.1. Convergência dos métodos baseados em RF.

5.6.1 Convergência dos Métodos Baseados em *Relevance Feedback*

Esta avaliação é iniciada considerando um cenário de *cold start* ($\mathcal{I}_o = \emptyset$), e discutindo a convergência do processo de RF quando aplicado ao Sum^+wTS e $LATRE^+wTS$. A Figura 5.1 mostra os valores médios de $P@5$ para as recomendações dadas por cada método em cada iteração do processo de RF. A iteração 0 corresponde às primeiras *tags* mostradas para o usuário. Para cada método associativo, são mostrados os resultados das abordagens citadas anteriormente para gerar o primeiro conjunto de recomendações, assim como o método híbrido baseado em PG. É mostrado também o resultado obtido pelo Plurality. Note que todos os métodos convergem bem rapidamente, alcançando seus melhores resultados, ou próximo a isso, com apenas 1 *feedback* fornecido pelo usuário (1 iteração). De fato, os ganhos obtidos com os *feedbacks* subsequentes, que existem apenas nas estratégias propostas e advêm do uso do *feedback* negativo, são bem pequenos. Assim pouco esforço é requerido do usuário para que a qualidade das recomendações melhore.

Observe que caso o usuário não esteja disposto a selecionar *tags* e consequentemente fornecer *feedback* não é possível melhorar as recomendações. Entretanto o esforço extra requerido é bem pequeno como mostra a Figura 5.1, uma vez que com apenas 1 *feedback* os métodos conseguem uma melhoria considerável. Observe ainda que $PG + RF$ alcança um pequeno ganho sobre os outros métodos mesmo na iteração 0 quando nenhum *feedback* foi utilizado ainda pelos métodos de recomendação.

Como mencionado na Seção 4.3 o $PG + RF$ se adapta a diferentes níveis de quantidade de informação (a partir da geração de uma função de recomendação específica para cada iteração do processo de RF) como pode ser visto na Figura 5.1 (principalmente para o Bibsonomy e Last.FM) onde o $PG + RF$ é o único método a conseguir ganhos após a primeira iteração.

5.6.2 Impacto do *Feedback* Negativo

Como mencionado na Seção 4.2 a estratégia proposta faz uso do *feedback* negativo provido pelo usuário. Essa seção mostra os benefícios de tal abordagem, comparado os ganhos obtidos pela estratégia proposta considerando tanto o uso apenas do *feedback* positivo quanto o uso de ambos os tipos de *feedback*.

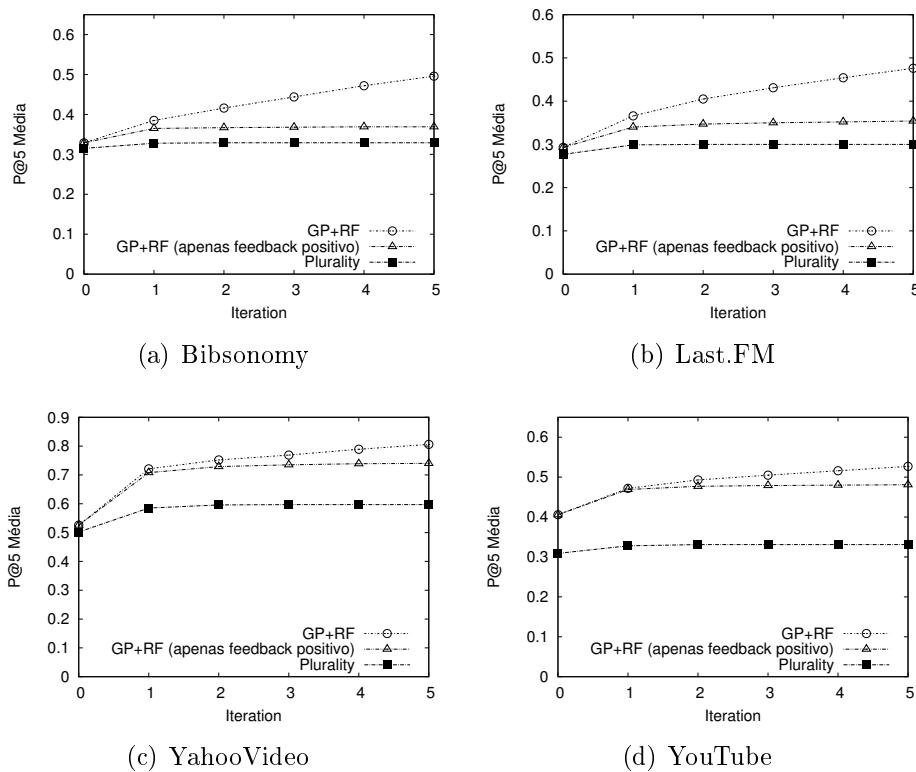


Figura 5.2. Impacto do *feedback* negativo sobre a eficácia das recomendações.

A Figura 5.2 mostra os valores médios de $P@5$ para as recomendações dadas pelo $PG + RF$ em cada iteração do processo de RF usando apenas o feedback positivo e usando ambos os tipos de feedback. Como pode ser visto a adição do *feedback* negativo ao método proposto leva a ganhos de até 34% em todas as bases de dados. Tais ganhos advêm principalmente do aumento da quantidade de informação disponível para o método de recomendação e mostram a importância de tal fonte de informação para os métodos propostos.

5.6.3 Relevance Feedback em Cenários de *Cold Start*

A Tabela 5.7 mostra a $P@5$ juntamente com intervalos de confiança de 95% para cada método após a convergência, em cada base de dados. Melhores resultados são mostrados em negrito. Comparando cada estratégia baseada em RF com o método original sobre o qual ela foi construída (linha $Sum/LATRE^+wTS$ na Tabela 5.7) os ganhos médios em $P@5$ obtidos com as novas estratégias atingem 73%, 51% e 87% para o Bibsonomy, YouTube e YahooVideo, e de 457% para o Last.FM. Os ganhos médios da melhor estratégia proposta (vide discussão abaixo) sobre o melhor método de referência em cada base de dados são 57%, 58%, 35% e 40%, para o Bibsonomy, Last.FM, YahooVideo e YouTube, respectivamente. Resultados muito similares, apresentados no Apêndice A, são obtidos quando se consideram revocação e AP.

De modo geral, $PG + RF$ é o melhor método proposto, uma vez que ele atinge os melhores resultados na maioria das bases. A exceção fica por conta do YouTube, em que o $CTTR+Sum^+wTS$ é ligeiramente (3%) melhor. No YahooVideo, a diferença de eficácia entre o $PG + RF$ e o $CF+LATRE^+wTS$ é marginal (inferior a 1%) e não compensa o custo extra do uso do $CF+LATRE^+wTS$. Nota-se que o $PG + RF$ supera o Plurality em até 59%. Estes ganhos são conseguidos principalmente por causa do uso do *feedback* negativo e pela combinação de métodos baseados em coocorrência com métricas que levam em conta a diferença de qualidade entre os vários atributos textuais dos objetos, o que, como mostrado na Tabela 5.7, supera métodos mais simples de filtragem colaborativa sobre os quais o Plurality é construído.

Por fim, deve-se destacar que mesmo que os ganhos das estratégias propostas tenham sido bem significativos ainda há muito espaço para melhorias uma vez que em três das quatro bases avaliadas a $P@5$ alcançada pelos métodos de recomendação fica em torno de 0.5.

Tabela 5.7. Estratégias de recomendação de *tags* baseadas em *Relevance feedback* aplicadas em objetos em *cold start*: Valores médios de P@5 juntamente com intervalos de confiança de 95%.

Métodos de Referência				
Estratégia	Bibsonomy	Last.FM	YahooVideo	YouTube
<i>CTTR</i>	0.275±0.002	0.260±0.001	0.465±0.004	0.376±0.002
<i>Sum⁺wTS</i>	0.207±0.002	0.065±0.001	0.419±0.004	0.359±0.002
<i>LATRE⁺wTS</i>	0.207±0.002	0.065±0.001	0.419±0.004	0.359±0.002
Plurality	0.315±0.002	0.300±0.001	0.597±0.001	0.331±0.002
Estratégias de recomendação de <i>tags</i> baseadas em <i>Relevance feedback</i>				
Estratégia	Bibsonomy	Last.FM	YahooVideo	YouTube
<i>Sum⁺wTS</i>	0.285±0.002	0.224±0.002	0.751±0.004	0.514±0.002
<i>CF+Sum⁺wTS</i>	0.343±0.002	0.386±0.002	0.795±0.003	0.547±0.003
<i>CTTR+Sum⁺wTS</i>	0.357±0.002	0.362±0.002	0.781±0.004	0.545±0.003
<i>LATRE⁺wTS</i>	0.289±0.002	0.230±0.002	0.778±0.004	0.518±0.002
<i>CF+LATRE⁺wTS</i>	0.361±0.002	0.307±0.002	0.826±0.004	0.555±0.002
<i>CTTR+LATRE⁺wTS</i>	0.360±0.002	0.360±0.002	0.787±0.004	0.537±0.002
<i>PG + RF</i>	0.496±0.003	0.476±0.004	0.806±0.007	0.527±0.004

5.6.4 Avaliação das Estratégias Baseadas em RF em Presença de Tags

Nesta Seção são avaliadas as estratégias baseadas em RF em um cenário no qual os objetos possuem *tags* previamente associadas ($\mathcal{I}_o \neq \emptyset$), selecionadas como discutido na Seção 5.2. Como mostrado na Tabela 5.8, o processo de RF melhora a precisão dos dois métodos de recomendação associativa em todas as bases de dados também neste cenário. Novamente, o uso do *CTTR* para gerar o primeiro conjunto de *tags* recomendadas, ao invés de um método de recomendação associativa, possibilita uma melhora nos resultados, exceto no YahooVideo onde o *LATRE⁺wTS* teve um desempenho semelhante ao do *CTTR+LATRE⁺wTS*.

PG + RF é novamente a melhor estratégia proposta, alcançando ganhos médios sobre o melhor método de referência (que pode ser o *Sum⁺wTS* ou o *LATRE⁺wTS* sem o uso de RF) de cerca de 38%, 43%, 17% e 18% no Bibsonomy, Last.FM, YahooVideo e YouTube, respectivamente. Note que, no YahooVideo, *CTTR+LATRE⁺wTS* (e *LATRE⁺wTS* com RF) obtêm resultados semelhantes ao *PG + RF*, e que no YouTube o *CTTR+Sum⁺wTS* obtêm um pequeno ganho (6%). Note novamente a grande superioridade do *PG + RF* tanto sobre o *CTTR* quanto sobre o Plurality.

Por fim, é interessante destacar que todos os métodos usados como referência bem como as extensões propostas são muito eficientes do ponto de vista de execução uma vez que o ponto mais custoso de todas as estratégias apresentadas é o treinamento

Tabela 5.8. Estratégias de recomendação de *tags* baseadas em *Relevance feedback* aplicadas em objetos com *tags*: Valores médios de $P@5$ juntamente com intervalos de confiança de 95%.

Métodos de Referência				
Estratégia	Bibsonomy	Last.FM	YahooVideo	YouTube
<i>CTTR</i>	0.275±0.002	0.260±0.001	0.465±0.004	0.376±0.002
<i>Sum⁺wTS</i>	0.436±0.002	0.417±0.002	0.707±0.002	0.502±0.003
<i>LATRE⁺wTS</i>	0.420±0.002	0.411±0.001	0.733±0.003	0.489±0.003
Plurality	0.315±0.002	0.315±0.001	0.595±0.001	0.332±0.002
Estratégias de recomendação de <i>tags</i> baseadas em <i>Relevance feedback</i>				
Estratégia	Bibsonomy	Last.FM	YahooVideo	YouTube
<i>Sum⁺wTS</i>	0.501±0.002	0.523±0.001	0.833±0.003	0.609±0.002
<i>LATRE⁺wTS</i>	0.541±0.002	0.525±0.002	0.872±0.003	0.620±0.002
<i>CTTR+Sum⁺wTS</i>	0.549±0.002	0.579±0.002	0.854±0.003	0.633±0.002
<i>CTTR+LATRE⁺wTS</i>	0.558±0.002	0.557±0.002	0.872±0.007	0.623±0.003
<i>PG + RF</i>	0.604±0.002	0.597±0.003	0.873±0.005	0.593±0.002

que é feito offline.

5.6.5 Sensibilidade à Falta de Cooperação e Confiabilidade do Usuário

Finalmente, foi analisada a sensibilidade do melhor método proposto, o *PG + RF*, baseado em RF à falta de cooperação e confiabilidade do usuário. Para o primeiro caso, foi considerado um cenário em que em cada iteração do processo de RF o usuário clica em $r\%$ ($0 \leq r \leq 100$) das *tags* mostradas a ele que ele considera relevantes. As Figuras 5.3(a-h) mostram os valores médios de $P@5$ para vários valores de r , tanto na presença (a, c, e, g) quanto na ausência (b, d, f, h) de *cold start*, para todas as bases de dados. Como pode ser visto, a eficácia do método diminui quando r diminui, já que, como o *feedback* negativo é explorado *tags* que não são marcadas como relevantes pelo usuário são consideradas irrelevantes e deixam de ser recomendadas em iterações futuras. Note que, em *cold start*, o *PG + RF* produz resultados, se não superiores, tão bons quanto os resultados do melhor método de referência (também mostrados nas figuras) mesmo que 50% das *tags* mostradas para o usuário não sejam selecionadas. A exceção fica por conta do YahooVideo em que é necessário selecionar pelo menos 80% das *tags*. Nas mesmas circunstâncias, em presença de *tags* iniciais, o método proposto consegue superar os métodos de referência apenas no Bibsonomy e Last.FM, necessitando de uma cooperação maior por parte do usuário de cerca de 80% no YahooVideo e YouTube para que ele seja capaz de superar os métodos de referência.

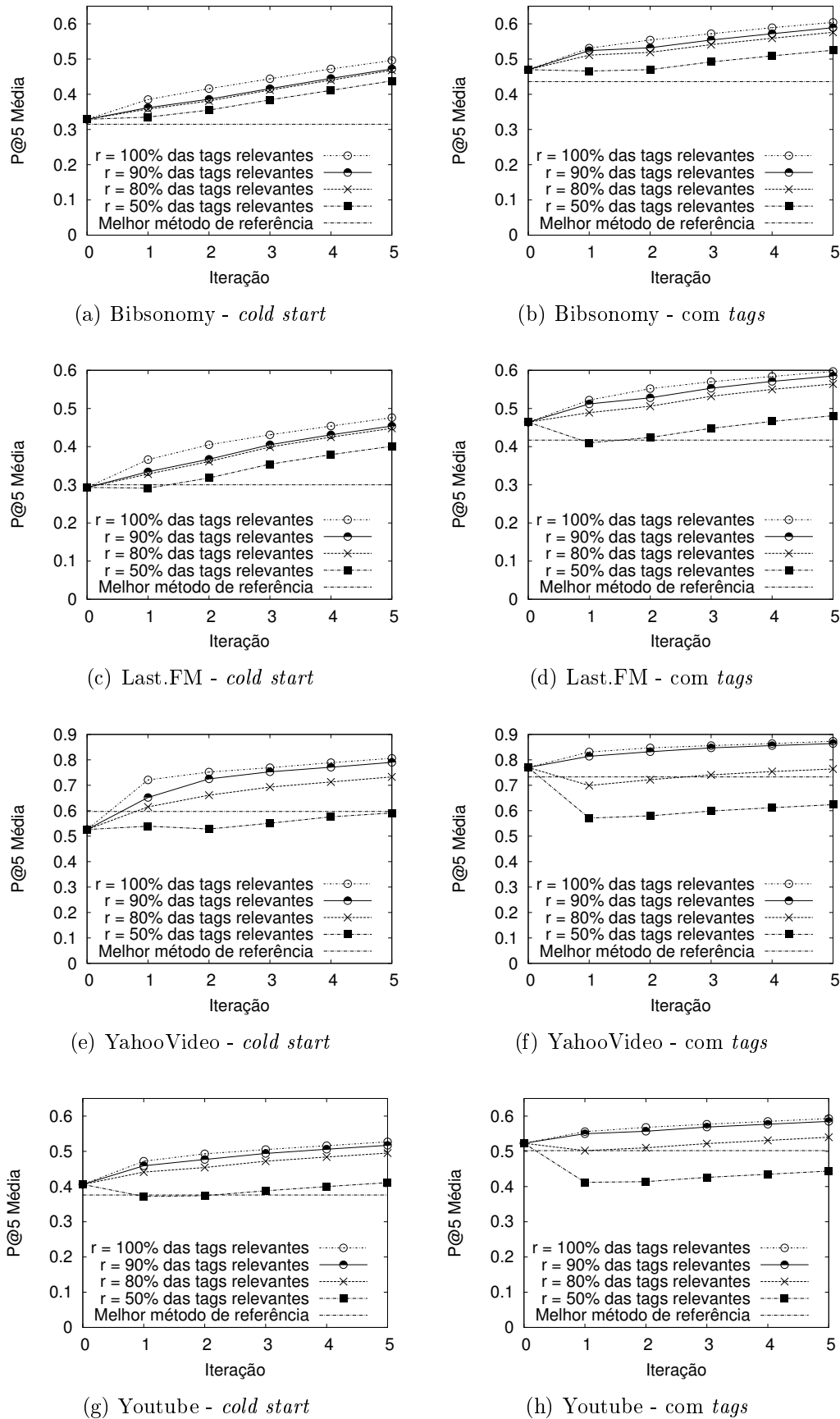


Figura 5.3. Sensibilidade dos métodos baseados em RF à falta de cooperação por parte do usuário.

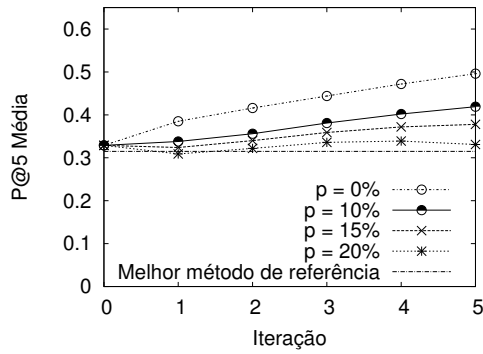
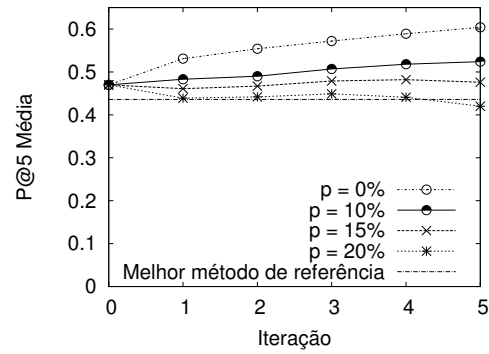
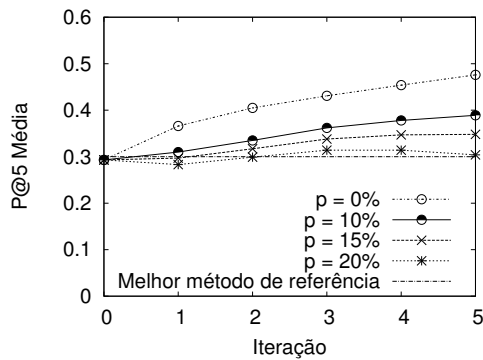
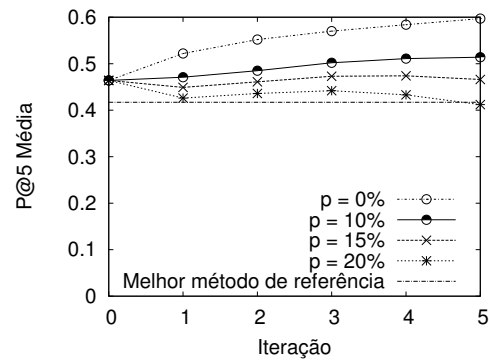
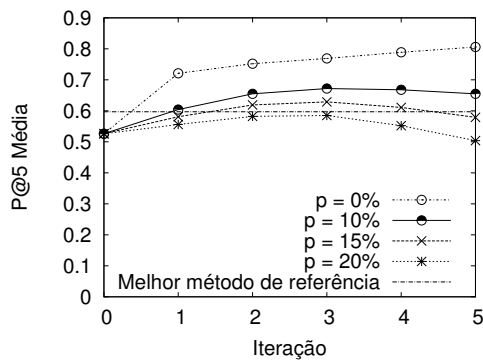
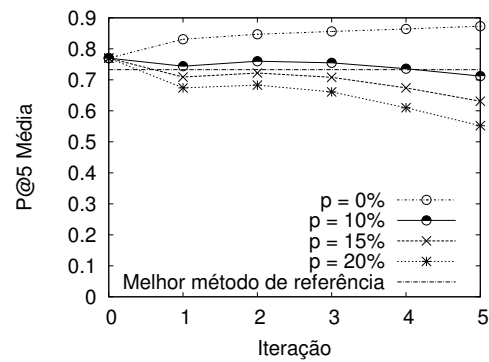
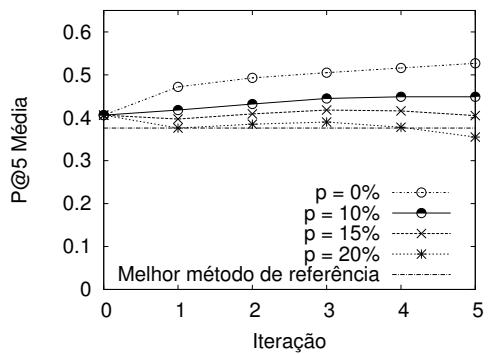
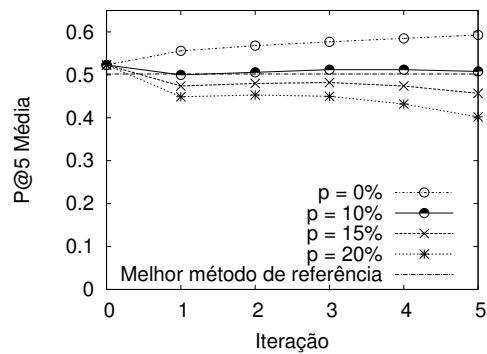
(a) Bibsonomy - *cold start*(b) Bibsonomy - *com tags*(c) Last.FM - *cold start*(d) Last.FM - *com tags*(e) YahooVideo - *cold start*(f) YahooVideo - *com tags*(g) YouTube - *cold start*(h) YouTube - *com tags*

Figura 5.4. Sensibilidade dos métodos baseados em RF à introdução de ruído.

Finalmente, foi considerado um cenário em que o usuário não só deixa de marcar *tags* relevantes mas também seleciona *tags* irrelevantes, adicionando assim ruído ao processo de RF. Isto foi simulado usando probabilidades p e q de o usuário selecionar um termo irrelevante e de deixar um termo relevante sem ser marcado, respectivamente. Por questões de simplicidade assume-se independência entre diferentes seleções de *tags*, e $q = p$. As Figuras 5.4(a-h) mostram os valores médios de $P@5$ em cada iteração, para vários valores de p , na presença (a, c, e, g) e na ausência (b, d, f, h) de *cold start* para todas as bases de dados. Note que, mesmo se a quantidade de ruído inserido no processo pelo usuário alcançar 15%, o método proposto continua superando o melhor método de referência em *cold start* (exceto para o YahooVideo onde esse limiar é de 10%) enquanto que, para objetos com *tags* iniciais, a tolerância ao ruído é um pouco menor (10%). Infelizmente, nós não sabemos de outros estudos que avaliem a quantidade de ruído acionado por um usuário real de sistemas de recomendação de *tags*, mas note que não é do interesse do usuário adicionar intencionalmente ruído. O que está sendo avaliado aqui é a tolerância do método a erros não intencionais cometidos pelo usuário. Pensando assim, 10-15% é uma quantidade razoavelmente grande de tolerância a ruído.

5.6.6 Exemplos de Recomendação

Para concluir nossa análise, ilustramos a operação de nosso melhor método - $GP + RF$ - para alguns objetos extraídos de nossas bases de dados. A Tabela 5.9 mostra exemplos de *tags* recomendadas pelo método $GP + RF$ depois de 5 iterações para três vídeos do YouTube, um artista do Last.FM e um artigo do Bibsonomy, considerando um cenário de *cold start*. Para fins de comparações, nós também mostramos as *tags* previamente associadas a cada objeto pelos usuários do YouTube / Last.FM / Bibsonomy (3ª coluna). Estas *tags* não foram usadas como entrada para nosso método, uma vez que estamos considerando um cenário de *cold start*. Assim, uma grande sobreposição entre os dois conjuntos de *tags* é uma prova de que o nosso método de fato sugere *tags* relevantes para o objeto alvo. Referimo-nos a cada objeto por seu título⁶.

O primeiro vídeo (linha 1), intitulado *Every Other Time*, é um trailer de um episódio da popular série de TV Smallville. Para esse objeto, nosso método conseguiu sugerir, como *tags*, o nome da série, bem como o nome de dois de seus principais personagens. Note que, duas dessas *tags* coincidem com as *tags* que realmente foram

⁶As URLs correspondentes são: <http://www.youtube.com/watch?v=mBh3jKIAYAU>, <http://www.youtube.com/watch?v=RABipYS7-qm>, <http://www.youtube.com/watch?v=87vc0S0G-Deg>, <http://www.last.fm/music/Radiohead>, e <http://www.bibsonomy.org/bibtex/-23b54b998c5369d6029bbca5be5881d04/flint63>. Todos esses objetos estavam disponíveis on-line para acesso público em fevereiro de 2013

Tabela 5.9. Exemplos de recomendações no YouTube.

Título e URL	Tags Recomendadas	Tags Originais Atribuídas pelo Usuário
Every Other Time	smallville, clark, lana	smallvile, lex, lana
Episode 93: The Birth of Lavos	chrono, trigger, lavos, azala	chrono, trigger, walkthroug
HotlineTV-Dark Horse Republican Governors	republican, elections, chuck, todd	hotlinetv, todd, mercurio, oregon
RadioHead	rock, radiohead, indie, cool	alternative, rock, indie, electronic,
True Knowledge: Open-Domain Question Answering Using Structured Knowledge and Inference	knowledge, processing, ai, paper	aaai, paper, ai, language, processing, information, retrieval, knowledge

assinaladas pelo usuário real para o objeto. Isso é uma evidência que nosso método seria efetivo na recomendação de tags de qualidade para esse objeto antes que nenhuma tag fosse atribuída a ele.

O segundo vídeo (linha 2), intitulado *Episode 93: The Birth of Lavos*, se refere ao jogo Chrono Trigger. Note que, tags como “lavos” e “azala” se referem a personagens desse jogo que realmente aparecem nas cenas do vídeo. Assim, elas são de fato relevantes para esse objeto, mesmo que elas não tenham sido previamente associadas ao objeto.

No terceiro vídeo (linha 3), intitulado *HotlineTV-Dark Horse Republican Governors*, os apresentadores Check Todd e John Mercurio discutem as eleições para governador dos Estados Unidos. Note que, nosso método sugere, corretamente, o assunto do vídeo (eleições), o nome de um dos apresentadores, bem como o principal partido político associado ao vídeo.

O objeto selecionado do Last.FM (linha 4) é a banda inglesa de rock *Radiohead*. Note que, nosso método consegue sugerir tags relacionadas ao estilo musical da banda (“indie”, “rock”), além do próprio nome da banda.

Nosso último exemplo é o artigo intitulado “True Knowledge: Open-Domain Question Answering Using Structured Knowledge and Inference”, que descreve uma plataforma social de perguntas e respostas chamada True Knowledge. Nosso método foi capaz de recomendar tags que representam a área do artigo (“ai”), bem como tags que sumarizam o conteúdo do artigo (“knowledge”, “processing”). Note, também, que todas as tags haviam sido previamente associadas ao objeto pelo usuário real da aplicação, atestando que, em um cenário de cold start, as recomendações seriam de fato relevantes.

5.7 Resumo dos Resultados

Neste capítulo foi avaliada uma família de métodos associativos de recomendação de *tags* (i.e., Sum^+wTS e $LATRE^+wTS$) em presença de *cold start*. Foi observado que a precisão de tais métodos ($Sum/LATRE^+wTS$) foi bastante prejudicada e que eles foram superados por métodos mais simples como o *CTTR* nesse cenário. Diferentemente do que ocorreu na ausência de *cold start* quando tanto o Sum^+wTS quanto o $LATRE^+wTS$ superaram as outras estratégias avaliadas em todas as bases, consistentemente com os resultados obtidos em [Belém et al., 2011].

Avaliou-se também o uso de estratégias de filtragem para melhorar a eficácia dos métodos associativos de recomendação na presença de *cold start*. Foi possível observar que, a despeito da melhoria na eficácia do Sum^+wTS , qualquer uma das abordagens consideradas, quando usadas em conjunto com o Sum^+wTS produziram uma melhora marginal (3-4%) sobre o *CTTR*. Mais ainda, os resultados produzidos pelo $LATRE^+wTS$ combinado com as estratégias propostas foram claramente piores que os do *CTTR*, e em alguns casos, piores até que o método $LATRE^+wTS$ original sem o uso de *tags* iniciais. Isso mostrou o potencial limitado de tais estratégias motivando a busca de novas soluções para o problema.

Em seguida, foram avaliados os resultados das estratégias baseadas em RF. De modo geral, o uso do *relevance feedback* amenizou em muito o problema do *cold start*. Dentre as estratégias propostas $PG + RF$ foi a que apresentou os melhores resultados na maioria das bases consideradas. Os ganhos médios de tal estratégia sobre o melhor método de referência em cada base de dados foram 57%, 58%, 35% e 40%, para o Bibsonomy, Last.FM, YahooVideo e YouTube, respectivamente. Verificou-se também que o processo de RF melhora a precisão dos dois métodos de recomendação associativa mesmo em cenários onde o *cold start* não ocorre. $PG + RF$ foi novamente a melhor estratégia proposta quando o objeto alvo da recomendação possuía *tags*, alcançando ganhos médios sobre o melhor método de referência de cerca de 38%, 43%, 17% e 18% no Bibsonomy, Last.FM, YahooVideo e YouTube, respectivamente.

Por fim mostrou-se que pouco esforço é necessário por parte do usuário para que as estratégias baseadas em RF sejam eficazes uma vez que elas convergem bem rapidamente (1 iteração) mesmo em cenários onde o usuário não é muito participativo, deixando de marcar até metade das *tags* relevantes, ou confiável, selecionando até 10% de *tags* irrelevantes como se fossem relevantes.

Capítulo 6

Conclusões e Trabalhos Futuros

Este capítulo apresenta um resumo das principais conclusões e contribuições (seção 6.1) desta dissertação bem como aponta para possíveis direções de trabalhos futuros (seção 6.2).

6.1 Conclusões

Esta dissertação estudou um instância do problema do *cold start* no contexto de recomendação de *tags*, particularmente foram estudados métodos associativos considerados estado-da-arte mas que até então haviam sido avaliados somente para objetos que já continham algumas *tags* iniciais. Tais estratégias foram aqui avaliadas sobre objetos sem *tags* previamente associadas, uma variação do *cold start*. Foi mostrado que *tags* previamente associadas ao objeto alvo são a fonte de informação mais importante para métodos associativos de recomendação de *tags* e também realizou-se um estudo quantificando o impacto do *cold start* sobre uma família de métodos estado-da-arte baseados em padrões de coocorrência de *tags*.

Outra contribuição dessa dissertação foi a proposição de duas estratégias gerais para aliviar tal problema. Uma delas é baseada na filtragem automática de termos de outras fontes para a construção de um conjunto \mathcal{I}_o alternativo sobre o qual padrões de coocorrência podem ser obtidos. A outra consiste em uma família de métodos de recomendação que exploram *relevance feedback* que se mostraram muito mais robustos ao problema do *cold start*.

Nosso estudo revelou que a perda em precisão dos métodos associativos estudados chega a até 84% em *cold start*. Essa perda é tão grande que nesse cenário tais métodos de recomendação são superados inclusive por métodos mais simples que não exploram padrões de coocorrência.

As estratégias propostas para gerar automaticamente o conjunto inicial de tags pela filtragem tanto de termos de objetos similares no conjunto de treino quanto de termos obtidos de outros atributos textuais do objeto (e.g., título e descrição) apresentaram ganhos limitados, motivando a busca por novas soluções para o problema.

A principal contribuição desta dissertação foi uma extensão dos métodos propostos em [Belém et al., 2011] para explorar o *relevance feedback* implícito provido pelo usuário do sistema durante o processo de recomendação. Foram avaliadas algumas variações do processo de *relevance feedback* supracitado, desde o uso de simples heurísticas até a aplicação de Programação Genética (PG) ao problema. Os resultados mostraram que a estratégia proposta reduz efetivamente o impacto do *cold start*, melhorando em até 58% a eficácia dos métodos de recomendação originais baseados em coocorrência. Foi verificado também que as estratégias propostas aumentam em até 45% a eficácia dos métodos baseados em coocorrência mesmo em cenários onde o *cold start* não ocorre. Foi verificado também que o método que combina PG e RF foi o que apresentou os melhores resultados na maioria das bases estudadas.

Por fim investigou-se a sensibilidade das estratégias propostas baseadas em *relevance feedback* à participação e confiabilidade do usuário. Em particular avaliou-se o impacto causado pela adição de ruído (i.e., termos irrelevantes) ou pelo não assinalamento de termos relevantes. Os resultados mostraram que a solução proposta permanece tão boa quanto, se não melhor, que outros métodos encontrados na literatura, mesmo que 20% do *feedback* provido pelo usuário seja composto por ruído.

6.2 Trabalhos Futuros

O estudo realizado sobre recomendação de *tags* e *cold start* abre espaço para diversos trabalhos futuros, dentre eles a extensão dos métodos propostos para lidar com recomendação personalizada, visto que usuários diferentes podem optar por termos distintos ao descrever um mesmo objeto.

Outro ponto interessante seria a realização de experimento com usuários para as estratégias propostas. Tal experimento poderia ser usado para fornecer mais subsídios por exemplo fornecendo uma caracterização mais precisa sobre a cooperação e a confiabilidade do usuário no processo de RF. Tal estudo também poderia ser usado para confirmar nossa premissa de que os resultados obtidos de acordo com a metodologia de avaliação automática utilizada representam limites inferiores da qualidade dos métodos avaliados.

A forma como o *feedback* negativo foi utilizada nesta dissertação é muito restritiva

já que uma *tag* que não foi marcada como relevante não é necessariamente irrelevante (na opinião do usuário). Assim uma direção de trabalho futuro interessante seria o estudo de outras abordagens para tratar o *feedback* negativo. Isso poderia ser feito, por exemplo atribuindo-se pesos negativos as *tags* não marcadas pelo usuário, isso faria com que tais *tags* ainda fossem consideradas pelos métodos de recomendação embora passassem a ocupar posições inferiores na ordenação dos termos candidatos. Outra forma menos restritiva de se tratar o *feedback* negativo seria criar um mecanismo de promoção das *tags* que se encontram na lista negra.

Um outro trabalho interessante seria comparar a estratégia baseada em PG proposta a métodos de aprendizado de máquina baseados em *online learning* tanto do ponto de vista de eficácia quanto de eficiência.

Por fim, como a estratégia proposta baseada em *relevance feedback* necessita do usuário, um estudo interessante seria tentar adaptá-la para que ela se torne automática, como é o caso de estratégias de pseudo *relevance feedback* de modo que ela possa ser usada em grandes coleções de objetos em *cold start* sem que a supervisão de um usuário seja necessária.

Referências Bibliográficas

- Agrawal, R. & Srikant, R. (1994). Fast algorithms for mining association rules in large databases. Em *Proceedings of the 20th International Conference on Very Large Data Bases*, VLDB, pp. 487--499. Morgan Kaufmann Publishers Inc.
- Algarni, A.; Li, Y.; Xu, Y. & Lau, R. Y. K. (2009). An effective model of using negative relevance feedback for information filtering. Em *CIKM*, pp. 1605–1608. ACM.
- Baeza-Yates, R. & Ribeiro-Neto, B. (2011). *Modern Information Retrieval*. Addison-Wesley.
- Banzhaf, W.; Nordin, P.; Keller, R. E. & Francone, F. D. (1998). *Genetic Programming – An Introduction On The Automatic Evolution Of Computer Programs And Its Applications*. Morgan Kaufmann, San Francisco, CA, USA.
- Belém, F.; Martins, E.; Pontes, T.; Almeida, J. & Gonçalves, M. (2011). Associative tag recommendation exploiting multiple textual features. Em *Proceedings of the 34th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR, pp. 1033–1042, Beijing, China.
- Bobadilla, J.; Ortega, F.; Hernando, A. & Bernal, J. (2012). A collaborative filtering approach to mitigate the new user cold start problem. *Knowledge Based Systems*, 26:225–238.
- Boll, S. (2007). MultiTube—Where Web 2.0 And Multimedia Could Meet. *IEEE MultiMedia*.
- Cao, H.; Xie, M.; Xue, L.; Liu, C.; Teng, F. & Huang, Y. (2009). Social tag prediction based on supervised ranking model. Em *Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*, ECML PKDD, pp. 35--48, Bled, Slovenia.

- Chen, L.; Wright, P. & Nejdl, W. (2009). Improving music genre classification using collaborative tagging data. Em *Proceedings of the Second ACM International Conference on Web Search and Data Mining, WSDM*, pp. 84–93, Barcelona, Espanha. ACM.
- Cristianini, N. & Shawe-Taylor, J. (2000). *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. Cambridge University Press, 1 edição.
- Fernandes, D.; Moura, E.; Ribeiro-Neto, B.; da Silva, A. & Gonçalves, M. (2007). Computing block importance for searching on web sites. Em *Proceedings of the 16th ACM Conference on Information and Knowledge Management, CIKM*, pp. 165–174, Lisboa, Portugal. ACM.
- Ferreira, C. D.; dos Santos, J. A.; da Silva Torres, R.; Gonçalves, M. A.; Rezende, R. C. & Fan, W. (2011). Relevance feedback based on genetic programming for image retrieval. *Pattern Recognition Letters*, 32(1):27–37.
- Figueiredo, F.; Pinto, H.; Beláçm, F.; Almeida, J. M.; Gonçalves, M. A.; Fernandes, D. & de Moura, E. S. (2013). Assessing the quality of textual features in social media. *Information Processing & Management*, 49(1):222–247.
- Freund, Y.; Iyer, R. D.; Schapire, R. E. & Singer, Y. (2003). An efficient boosting algorithm for combining preferences. *Journal of Machine Learning Research*, 4:933–969.
- Garg, N. & Weber, I. (2008). Personalized, interactive tag recommendation for flickr. Em *Proceedings of the 2008 ACM Conference on Recommender Systems, RecSys*, pp. 67–74, Lausanne, Switzerland. ACM.
- Giacinto, G. (2007). A nearest-neighbor approach to relevance feedback in content based image retrieval. Em *CIVR*, pp. 456–463. ACM.
- Givon, S. & Lavrenko, V. (2009). Predicting social-tags for cold start book recommendations. Em *RecSys*, pp. 333–336. ACM.
- Graham, R. & Caverlee, J. (2008). Exploring feedback models in interactive tagging. Em *WI-IAT '08: Proceedings of the 2008 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*, pp. 141–147, Washington, DC, USA. IEEE Computer Society.
- Guan, Z.; Bu, J.; Mei, Q.; Chen, C. & Wang, C. (2009). Personalized tag recommendation using graph-based ranking on multi-type interrelated objects. Em *Proceedings*

- of the 32nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR, pp. 540--547, Boston, MA, USA. ACM.
- Gunawardana, A. & Meek, C. (2009). A unified approach to building hybrid recommender systems. Em *Proceedings of the third ACM conference on Recommender systems*, RecSys '09, pp. 117--124, New York, NY, USA. ACM.
- Guy, I.; Zwerdling, N.; Ronen, I.; Carmel, D. & Uziel, E. (2010). Social media recommendation based on people and tags. Em *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR, pp. 194--201, Geneva, Switzerland. ACM.
- Heymann, P.; Ramage, D. & Garcia-Molina, H. (2008). Social tag prediction. Em *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR, pp. 531--538, Singapura, Singapura. ACM.
- Hofmann, T. (2013). Probabilistic latent semantic analysis. *CoRR*, abs/1301.6705.
- Jäschke, R.; Marinho, L.; Hotho, A.; Schmidt-Thie, L. & Stum, G. (2007). Tag recommendations in folksonomies. Em *Proceedings of the 11th European Conference on Principles and Practice of Knowledge Discovery in Databases*, ECML PKDD, pp. 506--514, Warsaw, Poland. Springer-Verlag.
- Konstas, I.; Stathopoulos, V. & Jose, J. (2009). On Social Networks And Collaborative Recommendation. Em *Proceedings of the 32nd international ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR, pp. 195--202, Boston, MA, USA. ACM.
- Koren, Y. (2008). Factorization meets the neighborhood: a multifaceted collaborative filtering model. Em *Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 426--434. ACM.
- Lam, X. N.; Vu, T.; Le, T. D. & Duong, A. D. (2008). Addressing cold-start problem in recommendation systems. Em *Proceedings of the 2nd international conference on Ubiquitous information management and communication*, ICUIMC '08, pp. 208--211, New York, NY, USA. ACM.
- Li, X.; Guo, L. & Zhao, Y. E. (2008a). Tag-based Social Interest Discovery. Em *Proceedings of the 17th International Conference on World Wide Web*, WWW, pp. 675--684, Beijing, China. ACM.

- Li, X.; Guo, L. & Zhao, Y. E. (2008b). Tag-based social interest discovery. Em *Proceedings of the 17th International World Wide Web Conference*, pp. 675–684. ACM.
- Lipczak, M.; Hu, Y.; Kollet, Y. & Milios, E. (2009). Tag sources for recommendation in collaborative tagging systems. Em *Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases, ECML PKDD*, pp. 157--172.
- Liu, N. N.; Meng, X.; Liu, C. & Yang, Q. (2011). Wisdom of the better few: cold start recommendation via representative based rating elicitation. Em *RecSys*, pp. 37–44. ACM.
- Lu, Y.-T.; Yu, S.-I.; Chang, T.-C. & Hsu, J. Y.-j. (2009). A content-based method to enhance tag recommendation. Em *Proceedings of the 21st International Joint Conference on Artificial intelligence*, pp. 2064--2069, Pasadena, California, USA. Morgan Kaufmann Publishers Inc.
- Lv, Y. & Zhai, C. (2009). Adaptive relevance feedback in information retrieval. Em *CIKM*, pp. 255–264. ACM.
- Marshall, C. (2009). No bull, no spin: a comparison of tags with other forms of user metadata. Em *Proceedings of the 9th ACM/IEEE-CS Joint Conference on Digital Libraries, JCDL*, pp. 241--250, Austin, TX, USA. ACM.
- Mei, T.; Yang, B.; Hua, X.-S. & Li, S. (2011). Contextual video recommendation by multimodal relevance and user feedback. *ACM Trans. Inf. Syst.*, 29(2):10.
- Mei, T.; Yang, L.; Hua, X.-S.; Wei, H. & Li, S. (2007). Videosense: a contextual video advertising system. Em *ACM Multimedia*, pp. 463–464. ACM.
- Menezes, G.; Almeida, J.; Belém, F.; Gonçalves, M.; Lacerda, A.; Moura, E.; Pappa, G.; Veloso, A. & Ziviani, N. (2010). Demand-driven tag recommendation. Em *Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases, ECML PKDD*, pp. 402--417, Barcelona, Spain. Springer-Verlag.
- Ness, S. R.; Theocharis, A.; Tzanetakis, G. & Martins, L. G. (2009). Improving automatic music tag annotation using stacked generalization of probabilistic svm outputs. Em *ACM Multimedia*, pp. 705–708. ACM.

O'Reilly, T. (2005). What is web 2.0.

<http://oreilly.com/pub/a/web2/archive/what-is-web-20.html>.

Poli, R. (2002). *Foundations of genetic programming*. Springer-Verlag New York, Inc., New York, NY, USA.

Preisach, C.; Marinho, L. B. & Schmidt-Thieme, L. (2010). Semi-supervised tag recommendation - using untagged resources to mitigate cold-start problems. Em *PAKDD (1)*, volume 6118 of *Lecture Notes in Computer Science*, pp. 348–357. Springer.

Ramage, D.; Heymann, P.; Manning, C. D. & Garcia-Molina, H. (2009). Clustering the tagged web. Em *Proceedings of the Second ACM International Conference on Web Search and Data Mining, WSDM*, pp. 54–63, Barcelona, Espanha. ACM.

Rege, M.; Dong, M. & Hua, J. (2008). Graph theoretical framework for simultaneously integrating visual and textual features for efficient web image clustering. Em *WWW*, pp. 317–326. ACM.

Rendle, S. & Lars, S.-T. (2010). Pairwise interaction tensor factorization for personalized tag recommendation. Em *Proceedings of the Third ACM International Conference on Web search and Data Mining, WSDM*, pp. 81–90, New York, New York, USA. ACM.

Sahebi, S. & William, W. C. (2011). Community-based recommendations: a solution to the cold start problem. Em *Workshop on Recommender Systems and the Social Web (RSWEB)*.

Said, A.; Wetzker, R.; Umbrath, W. & Hennig, L. (2009). A hybrid plsa approach for warmer cold start in folksonomy recommendation. Em *RecSys*.

Schein, A. I.; Popescul, A.; Ungar, L. H. & Pennock, D. M. (2002). Methods and metrics for cold-start recommendations. Em *SIGIR '02: Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 253–260, New York, NY, USA. ACM Press.

Schenkel, R.; Crecelius, T.; Kacimi, M.; Michel, S.; Neumann, T.; Parreira, J. & Weikum, G. (2008). Efficient Top-k Querying Over Social-Tagging Networks. Em *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR*, pp. 523–530, Singapore, Singapore. ACM.

- Siersdorfer, S.; San Pedro, J. & Sanderson, M. (2009). Automatic video tagging using content redundancy. Em *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR, pp. 395--402, Boston, MA, USA. ACM.
- Sigurbjörnsson, B. & van Zwol, R. (2008). Flickr tag recommendation based on collective knowledge. Em *Proceedings of the 17th International Conference on World Wide Web*, WWW, pp. 327--336, Beijing, China. ACM.
- Song, Y.; Zhuang, Z.; Li, H.; Zhao, Q.; Li, J.; Lee, W.-C. & Giles, C. L. (2008). Real-time automatic tag recommendation. Em *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR, pp. 515--522, Singapore, Singapore. ACM.
- Sun, D.; Luo, Z. & Zhang, F. (2011). A novel approach for collaborative filtering to alleviate the new item cold-start problem. Em *11th International Symposium on Communications and Information Technologies*, ISCIT, pp. 402--406.
- Swain, M. J. & Ballard, D. H. (1991). Color indexing. *International Journal of Computer Vision*, 7(1):11--32.
- Venetis, P.; Koutrika, G. & Garcia-Molina, H. (2011). On the selection of tags for tag clouds. Em *Proceedings of the fourth ACM international conference on Web search and data mining*, WSDM '11, pp. 835--844, New York, NY, USA. ACM.
- Wang, C.; Yilmaz, E. & Szummer, M. (2011). Relevance feedback exploiting query-specific document manifolds. Em *CIKM*, pp. 1957--1960. ACM.
- Wang, J.; Hong, L. & Davison, B. D. (2009). Tag recommendation using keywords and association rules. Em *Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*, ECML PKDD.
- Wetzker, R.; Umbrath, W. & Said, A. (2009). A hybrid approach to item recommendation in folksonomies. Em *Proceedings of the WSDM '09 Workshop on Exploiting Semantic Annotations in Information Retrieval*, ESAIR '09, pp. 25--29, New York, NY, USA. ACM.
- Wu, H.; Wang, Y. & Cheng, X. (2008). Incremental probabilistic latent semantic analysis for automatic question recommendation. Em *RecSys '08: Proceedings of the 2008 ACM conference on Recommender systems*, pp. 99--106, New York, NY, USA. ACM.

- Wu, L.; Yang, L.; Yu, N. & Hua, X.-S. (2009). Learning to tag. Em *Proceedings of the 18th International Conference on World wide web*, WWW, pp. 361--370, Madrid, Spain. ACM.
- Xu, Z. & Akella, R. (2008). Active relevance feedback for difficult queries. Em *CIKM*, pp. 459--468. ACM.
- Yang, B.; Mei, T.; Hua, X.-S.; Yang, L.; Yang, S.-Q. & Li, M. (2007). Online video recommendation based on multimodal fusion and relevance feedback. Em *CIVR*, pp. 73--80. ACM.
- Yeh, J.; Lin, J.; Ke, H. & Yang, W. (2007). Learning To Rank For Information Retrieval Using Genetic Programming. Em *SIGIR 2007 Workshop: Learning To Rank For Information Retrieval*.
- Zhang, N.; Zhang, Y. & Tang, J. (2009). A tag recommendation system based on contents. Em *Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*, ECML PKDD.
- Zhou, K.; Yang, S.-H. & Zha, H. (2011). Functional matrix factorizations for cold-start recommendation. Em *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, SIGIR '11, pp. 315--324, New York, NY, USA. ACM.

Apêndice A

Revocação e Precisão Média das Estratégias de Recomendação

Esta seção apresenta os resultados de revocação e AP omitidos na Seção 5. Como pode ser visto, de um modo geral os resultados são consistentes para as três métricas usadas nessa dissertação para avaliar os métodos de recomendação de *tags*.

Tabela A.1. Importância das diferentes fontes de informação para a eficácia dos métodos associativos Sum^+wTS e $LATRE^+wTS$. Valores médios de revocação juntamente com intervalos de confiança de 95%.

Estratégia	Bibsonomy	Last.FM	YahooVideo	YouTube
CTTR	0.342 ± 0.002	0.347 ± 0.001	0.602 ± 0.004	0.431 ± 0.002
Sum^+wTS	0.422 ± 0.002	0.389 ± 0.002	0.613 ± 0.002	0.461 ± 0.003
$LATRE^+wTS$	0.421 ± 0.002	0.383 ± 0.001	0.636 ± 0.003	0.449 ± 0.003
Sum^+	0.315 ± 0.002	0.383 ± 0.001	0.420 ± 0.003	0.225 ± 0.002
LATRE	0.337 ± 0.002	0.378 ± 0.001	0.527 ± 0.003	0.262 ± 0.004
$Sum/LATRE^+wTS$	0.203 ± 0.002	0.059 ± 0.001	0.377 ± 0.004	0.340 ± 0.002

As Tabelas A.1 e A.2 apresentam os resultados de revocação e AP correspondentes à Tabela 5.5. Consistentemente com os resultados obtidos em na Seção 5.4 tanto o Sum^+wTS quanto o $LATRE^+wTS$ superam as outras estratégias (incluindo o CTTR) em todas as bases na ausência de *cold start* e são superados pelo CTTR no cenário de *cold start*.

As Tabelas A.3 e A.4 apresentam os resultados de revocação e AP correspondentes aos resultados apresentados na Tabela 5.6 da Seção 5.5. Novamente é possível observar que, a despeito da melhoria na eficácia do Sum^+wTS em *cold start*, qualquer uma das abordagens consideradas, quando usadas em conjunto com o Sum^+wTS apenas produziram uma melhora marginal (3-4%) sobre o CTTR e que em alguns casos as

Tabela A.2. Importância das diferentes fontes de informação para a eficácia dos métodos associativos Sum^+wTS e $LATRE^+wTS$. Valores médios de AP juntamente com intervalos de confiança de 95%.

Estratégia	Bibsonomy	Last.FM	YahooVideo	YouTube
CTTR	0.183 ± 0.002	0.174 ± 0.001	0.359 ± 0.004	0.266 ± 0.002
Sum⁺wTS	0.315 ± 0.002	0.312 ± 0.002	0.562 ± 0.002	0.384 ± 0.003
LATRE⁺wTS	0.314 ± 0.002	0.307 ± 0.001	0.583 ± 0.003	0.374 ± 0.003
Sum⁺	0.236 ± 0.002	0.307 ± 0.001	0.385 ± 0.003	0.187 ± 0.002
LATRE	0.252 ± 0.002	0.302 ± 0.001	0.484 ± 0.003	0.218 ± 0.004
Sum/LATRE⁺wTS	0.107 ± 0.002	0.026 ± 0.001	0.222 ± 0.004	0.194 ± 0.002

abordagens consideradas quando usadas em conjunto com o $Latre^+wTS$ foram piores até que o método $LATRE^+wTS$ original sem o uso de *tags* iniciais. De todo modo, observe que CF e $Pred$ continuam a apresentar os melhores resultados.

Tabela A.3. Estratégias de filtragem automáticas para construir \mathcal{I}'_o : revocação média juntamente com intervalos de confiança de 95%.

Estratégias de filtragem aplicadas ao Sum^+wTS				
Estratégia	Bibsonomy	Last.FM	Yahoo Video	YouTube
<i>CTTR</i>	0.218 ± 0.002	0.296 ± 0.002	0.579 ± 0.002	0.443 ± 0.003
<i>TF</i>	0.216 ± 0.002	0.209 ± 0.002	0.563 ± 0.003	0.429 ± 0.002
<i>TS</i>	0.216 ± 0.002	0.218 ± 0.003	0.567 ± 0.003	0.430 ± 0.002
<i>wTF</i>	0.216 ± 0.002	0.219 ± 0.003	0.573 ± 0.004	0.434 ± 0.003
<i>wTS</i>	0.216 ± 0.002	0.210 ± 0.002	0.573 ± 0.004	0.434 ± 0.002
<i>Cov</i>	0.213 ± 0.002	0.187 ± 0.003	0.522 ± 0.002	0.415 ± 0.001
<i>Pred</i>	0.234 ± 0.002	0.269 ± 0.001	0.593 ± 0.004	0.443 ± 0.002
<i>CF</i>	0.237 ± 0.001	0.336 ± 0.002	0.620 ± 0.003	0.451 ± 0.002
Estratégias de filtragem aplicadas ao $Latre^+wTS$				
Estratégia	Bibsonomy	Last.FM	YahooVideo	YouTube
<i>CTTR</i>	0.169 ± 0.002	0.090 ± 0.001	0.495 ± 0.003	0.390 ± 0.001
<i>TF</i>	0.130 ± 0.003	0.180 ± 0.001	0.579 ± 0.003	0.406 ± 0.002
<i>TS</i>	0.132 ± 0.002	0.180 ± 0.003	0.577 ± 0.002	0.406 ± 0.002
<i>wTF</i>	0.130 ± 0.002	0.180 ± 0.003	0.587 ± 0.003	0.413 ± 0.002
<i>wTS</i>	0.132 ± 0.002	0.179 ± 0.002	0.591 ± 0.003	0.415 ± 0.002
<i>Cov</i>	0.121 ± 0.003	0.095 ± 0.004	0.267 ± 0.005	0.199 ± 0.001
<i>Pred</i>	0.139 ± 0.002	0.174 ± 0.001	0.585 ± 0.003	0.425 ± 0.002
<i>CF</i>	0.188 ± 0.001	0.069 ± 0.001	0.343 ± 0.002	0.300 ± 0.002

A Tabelas A.5 e A.6 mostram os valores correspondentes de revocação e AP aos da Tabela 5.7 juntamente com intervalos de confiança de 95% para cada método após a convergência, em cada base de dados. Melhores resultados são mostrados em negrito e são consistentes com os resultados apresentados na seção 5.6.3. Os ganhos médios em

Tabela A.4. Estratégias de filtragem automáticas para construir \mathcal{I}'_o : AP média juntamente com intervalos de confiança de 95%.

Estratégias de filtragem aplicadas ao Sum^+wTS				
Estratégia	Bibsonomy	Last.FM	Yahoo Video	YouTube
<i>CTTR</i>	0.119 ± 0.002	0.137 ± 0.002	0.339 ± 0.002	0.259 ± 0.003
<i>TF</i>	0.113 ± 0.002	0.097 ± 0.001	0.324 ± 0.002	0.251 ± 0.001
<i>TS</i>	0.113 ± 0.002	0.101 ± 0.002	0.326 ± 0.002	0.251 ± 0.001
<i>wTF</i>	0.113 ± 0.002	0.101 ± 0.002	0.332 ± 0.002	0.254 ± 0.001
<i>wTS</i>	0.113 ± 0.002	0.097 ± 0.001	0.333 ± 0.002	0.254 ± 0.001
<i>Cov</i>	0.111 ± 0.001	0.090 ± 0.002	0.289 ± 0.002	0.247 ± 0.001
<i>Pred</i>	0.128 ± 0.001	0.125 ± 0.001	0.347 ± 0.003	0.259 ± 0.002
<i>CF</i>	0.130 ± 0.002	0.156 ± 0.002	0.363 ± 0.003	0.264 ± 0.002
Estratégias de filtragem aplicadas ao $Latre^+wTS$				
Estratégia	Bibsonomy	Last.FM	YahooVideo	YouTube
<i>CTTR</i>	0.093 ± 0.002	0.042 ± 0.001	0.290 ± 0.003	0.228 ± 0.001
<i>TF</i>	0.065 ± 0.001	0.083 ± 0.001	0.316 ± 0.002	0.221 ± 0.001
<i>TS</i>	0.066 ± 0.001	0.083 ± 0.002	0.317 ± 0.002	0.221 ± 0.001
<i>wTF</i>	0.065 ± 0.001	0.083 ± 0.002	0.319 ± 0.002	0.224 ± 0.001
<i>wTS</i>	0.066 ± 0.001	0.083 ± 0.001	0.318 ± 0.001	0.225 ± 0.002
<i>Cov</i>	0.055 ± 0.002	0.055 ± 0.002	0.180 ± 0.003	0.136 ± 0.002
<i>Pred</i>	0.079 ± 0.001	0.079 ± 0.001	0.315 ± 0.002	0.230 ± 0.002
<i>CF</i>	0.104 ± 0.002	0.051 ± 0.001	0.290 ± 0.001	0.218 ± 0.002

revocação da melhor estratégia proposta sobre o melhor método de referência em cada base de dados são 57%, 58%, 15% e 45%, para o Bibsonomy, Last.FM, YahooVideo e YouTube, respectivamente. Ganhos similares são obtidos quando se considera AP .

Tabela A.5. Estratégias de recomendação de *tags* baseadas em *Relevance feedback* aplicadas em objetos em *cold start*: Valores médios de revocação juntamente com intervalos de confiança de 95%.

Métodos de Referência				
Estratégia	Bibsonomy	Last.FM	YahooVideo	YouTube
CTTR	0.342 ± 0.002	0.347 ± 0.001	0.602 ± 0.004	0.431 ± 0.002
Sum/LATRE ⁺ wTS	0.203 ± 0.002	0.059 ± 0.001	0.377 ± 0.004	0.340 ± 0.002
Plurality	0.392±0.002	0.400±0.001	0.773±0.001	0.379±0.002
Estratégias de recomendação de <i>tags</i> baseadas em <i>Relevance feedback</i>				
Estratégia	Bibsonomy	Last.FM	YahooVideo	YouTube
<i>Sum</i> ⁺ <i>wTS</i>	0.354±0.002	0.299±0.002	0.798±0.004	0.589±0.002
CF+ <i>Sum</i> ⁺ <i>wTS</i>	0.427±0.002	0.515±0.002	0.823±0.003	0.627±0.003
CTTR+ <i>Sum</i> ⁺ <i>wTS</i>	0.444±0.002	0.483±0.002	0.876±0.004	0.625±0.003
<i>LATRE</i> ⁺ <i>wTS</i>	0.359±0.002	0.307±0.002	0.820±0.004	0.594±0.002
CF+ <i>LATRE</i> ⁺ <i>wTS</i>	0.449±0.002	0.449±0.002	0.885±0.004	0.636±0.002
CTTR+ <i>LATRE</i> ⁺ <i>wTS</i>	0.448±0.002	0.480±0.002	0.851±0.004	0.616±0.002
PG+RF	0.617±0.003	0.635±0.004	0.884±0.007	0.604±0.004

Tabela A.6. Estratégias de recomendação de *tags* baseadas em *Relevance feedback* aplicadas em objetos em *cold start*: Valores médios de *AP* juntamente com intervalos de confiança de 95%.

Métodos de Referência				
Estratégia	Bibsonomy	Last.FM	YahooVideo	YouTube
CTTR	0.183 ± 0.002	0.174 ± 0.001	0.359 ± 0.004	0.266 ± 0.002
Sum/LATRE ⁺ wTS	0.107 ± 0.002	0.026 ± 0.001	0.222 ± 0.004	0.194 ± 0.002
Plurality	0.210±0.002	0.201±0.001	0.461±0.001	0.234±0.002
Estratégias de recomendação de <i>tags</i> baseadas em <i>Relevance feedback</i>				
Estratégia	Bibsonomy	Last.FM	YahooVideo	YouTube
<i>Sum</i> ⁺ <i>wTS</i>	0.190±0.002	0.150±0.002	0.580±0.004	0.364±0.002
CF+ <i>Sum</i> ⁺ <i>wTS</i>	0.343±0.002	0.386±0.002	0.795±0.003	0.387±0.003
CTTR+ <i>Sum</i> ⁺ <i>wTS</i>	0.238±0.002	0.242±0.002	0.603±0.004	0.386±0.003
<i>LATRE</i> ⁺ <i>wTS</i>	0.192±0.002	0.154±0.002	0.601±0.004	0.366±0.002
CF+ <i>LATRE</i> ⁺ <i>wTS</i>	0.240±0.002	0.205±0.002	0.638±0.004	0.393±0.002
CTTR+ <i>LATRE</i> ⁺ <i>wTS</i>	0.240±0.002	0.241±0.002	0.608±0.004	0.380±0.002
PG+RF	0.330±0.003	0.319±0.004	0.622±0.007	0.373±0.004