RECOMENDAÇÃO ASSOCIATIVA DE TAGS CONSIDERANDO MÚLTIPLOS ATRIBUTOS TEXTUAIS

FABIANO MUNIZ BELÉM

RECOMENDAÇÃO ASSOCIATIVA DE TAGS CONSIDERANDO MÚLTIPLOS ATRIBUTOS TEXTUAIS

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 Ph. D.
Co-orientador: Marcos André Gonçalves Ph. D.

Belo Horizonte

Junho de 2011

Ficha catalográfica elaborada pela Biblioteca do ICEx - UFMG

Belém, Fabiano Muniz.

B428r Recomendação associativa de tags considerando múltiplos atributos textuais. / Fabiano Muniz Bélem – Belo Horizonte, 2011.
xvi, 79f. : il.; 29 cm.

Dissertação (mestrado) - Universidade Federal de Minas Gerais – Departamento de Ciência da Computação.

Orientadora: Jussara Marques de Almeida.
Coorientador: Marcos André Gonçalves

1. Computação. 2. Sistemas de recomendação. 3. Web 2.0. 4. Recuperação da informação. I. Orientadora. 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 considerando múltiplos atributos textuais

FABIANO MUNIZ BELEM

Dissertação defendida e aprovada pela banca examinadora constituída pelos Senhores:

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

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

PROF. PÁVEL PEREIRA CALADO
Universidade Técnica de Lisboa

PROF. ALTIGRAN SOARES DA SILVA
Departamento de Ciência da Computação - UFAM

PROF. NIVIO ZIVIANI
Departamento de Ciência da Computação - UFMG

Belo Horizonte, 06 de julho de 2011.

Resumo

Diversas aplicações da Web 2.0 permitem aos usuários a livre atribuição de palavras-chave (tags) ao conteúdo, para prover uma melhor organização e descrição do mesmo. Nessas aplicações, serviços de recomendação de tags podem auxiliar o usuário nesta tarefa, melhorando a qualidade da informação disponível e, consequentemente, a eficácia de serviços de Recuperação de Informação (RI) que exploram tags, tais como busca e classificação. Esta dissertação aborda duas tarefas relacionadas à recomendação de tags. A primeira, centrada no objeto, objetiva sugerir tags que sejam relevantes a um determinado conteúdo. A segunda tarefa é personalizada, visando recomendar tags que sejam relevantes ao mesmo tempo a um objeto alvo e a um usuário alvo. Nossas estratégias de recomendação centrada no objeto exploram conjuntamente três dimensões do problema: (i) co-ocorrência de termos com tags previamente associadas ao objeto alvo; (ii) termos extraídos de múltiplos atributos textuais; e (iii) várias métricas de relevância de tags. Nossas estratégias de recomendação de tags personalizada exploram todas as dimensões mencionadas acima e uma métrica relacionada ao histórico de atribuições de tags do usuário alvo. Em particular, nós propusemos várias novas heurísticas, que estendem estratégias estado-da-arte eficazes e eficientes incluindo novas métricas que tentam capturar a capacidade de um termo descrever o conteúdo de um objeto. Nós também exploramos duas técnicas baseadas em *learning-to-rank* (L2R), a saber, RankSVM e Programação Genética (PG). Elas geram funções que combinam múltiplas métricas para estimar a relevância de uma tag a um dado objeto ou a um par objeto-usuário. Nós avaliamos os métodos propostos em vários cenários, em três aplicações populares da Web 2.0: LastFM, YouTube e YahooVideo. Os resultados de nossas heurísticas apresentam ganhos significativos em precisão e revocação quando comparados a estratégias estado-da-arte. Alguns ganhos adicionais sobre nossas heurísticas podem ser obtidos através de nossas estratégias baseadas em L2R. Apesar dos ganhos serem modestos, vale ressaltar que essas técnicas têm a vantagem de serem bem flexíveis, podendo ser facilmente estendidas para explorar outros aspectos do problema de recomendação de tags.

Abstract

Several Web 2.0 applications allow users to assign keywords (tags) to the content, in order to provide a better organization and description of the shared content. Tag recommendation tasks may assist users in this task, improving the quality of the available information and, thus, the effectiveness of Information Retrieval (IR) services which exploit tags, such as searching and classification. This work addresses two tag recommendation tasks. The first one, content centered, aims at suggesting relevant tags to a target content. The second task is personalized, aiming at recommending tags which are simultaneously relevant to a target content and a target user. Our content centered tag recommendation strategies jointly exploit three dimensions of the problem: (i) term co-occurrence with tags pre-assigned to the target content, (ii) terms extracted from multiple textual features, and (iii) several metrics of tag relevance. Our personalized tag recommendation strategies exploit all dimensions mentioned above and a metric related to the history of tag assignments of the target user. In particular, we propose several new heuristic methods, which extend previous, highly effective and efficient state-of-the-art strategies by including new metrics that try to capture how accurately a candidate term describes the target content. We also exploit two learning-to-rank techniques, namely RankSVM and Genetic Programming, for the task of generating ranking functions that combine multiple metrics to accurately estimate the relevance of a tag to a given content or pair user-content. We evaluate all proposed methods in various scenarios for three popular Web 2.0 applications, namely, LastFM, YouTube and YahooVideo. We found that our heuristics greatly outperform the best baseline in each scenario and task. Some further, modest improvements can also be achieved, with our new learning-to-rank based strategies. They have the additional advantage of being quite flexible and easily extensible to exploit other aspects of the tag recommendation problem.

Lista de Figuras

Lista de Tabelas

Sumário

Capítulo 1

Introdução

O uso de *tags* (palavras-chave associadas a um conteúdo) popularizou-se na Web, principalmente com o surgimento de aplicações Web 2.0, que facilitam e estimulam a participação dos usuários na criação de conteúdo. Esse conteúdo, tipicamente multimídia (por exemplo, imagens e vídeos), traz desafios aos atuais métodos de Recuperação de Informação (RI) baseados no próprio conteúdo multimídia, não apenas devido a questões de escalabilidade e eficiência, mas também frente à (frequentemente pobre) qualidade do material gerado por usuários [Boll, 2007].

Diversas aplicações construídas sob os conceitos da Web 2.0 estão entre as mais populares da Web atualmente. Exemplos incluem o YouTube e o Yahoo! Video[1], dois sistemas de compartilhamento de vídeo, o Last.FM[2], uma estação de rádio *online* que possibilita a interação social em torno de tópicos relacionados a música, o Flickr[3], uma aplicação para compartilhar e organizar fotos, e o Delicious[4], um sistema utilizado para organização de *bookmarks*. 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 coleções de vídeos existentes. Nele, a quantidade de conteúdo disponibilizada em 60 dias é equivalente ao conteúdo que seria transmitido diariamente por 60 anos pelos canais de TV norte americanos NBC, CBS e ABC juntos[5]. Além disso, o acesso a vídeos no YouTube correspondem a quase 40% de todas as visualizações de vídeos na Internet[6].

[1] http://youtube.com e http://video.yahoo.com
[2] http://last.fm
[3] http://www.flickr.com
[4] http://www.del.icio.us
[5] http://www.nytimes.com/2009/09/06/magazine/06FOB-medium-t.htm
[6] http://www.comscore.com

Cada instância do conteúdo de uma aplicação da Web 2.0 é composta por um *objeto* principal, que pode estar armazenado na forma de vários tipos de mídia (texto, áudio, vídeo, imagem) e por diversas fontes de informação associadas ao objeto, denominadas aqui *atributos* do objeto. Os atributos do objeto podem ser de vários tipos. *Atributos de conteúdo* são fontes de informação que podem ser extraídas do próprio objeto, tais como o histograma de cores de uma imagem. *Atributos textuais* são blocos de texto frequentemente associados pelos usuários ao objeto, tais como tags, título e descrições. *Atributos sociais*, por sua vez, refletem o contexto social em que o objeto está inserido, ou seja, quem o criou, quais usuários o acessam e quais interações entre usuários são estabelecidas através do objeto. Podemos identificar também fontes de informação relacionadas a um usuário, aqui denominadas *atributos do perfil do usuário*. Idade, sexo e as tags que um usuário utiliza com frequência são exemplos de atributos do perfil desse usuário.

Tags, entre outros atributos textuais como título e descrição, comumente encontrados em aplicações da Web 2.0, constituem uma boa alternativa para organização, disseminação e recuperação de conteúdo. De fato, estudos recentes demonstram que tags estão entre os melhores atributos textuais para dar suporte a serviços de RI tais como classificação automática [Figueiredo et al., 2009a,b], busca [Li et al., 2008] e recomendação de conteúdo [Guy et al., 2010].

Nesse contexto, a recomendação de tags é um serviço provido por algumas aplicações com o objetivo de melhorar a qualidade da informação disponível, por meio da sugestão de termos que, idealmente, descrevam o conteúdo de forma mais precisa e completa. A pesquisa existente em recomendação de tags pode ser dividida em duas abordagens. Na primeira abordagem, a recomendação é *centrada no objeto*, objetivando sugerir termos que o descrevam bem. Em uma segunda abordagem, *personalizada*, o objetivo é sugerir termos que não só descrevam bem o objeto, mas também que sejam relevantes ao usuário alvo da recomendação. A personalização é motivada pelo fato de que usuários diferentes frequentemente optam por termos distintos ao descrever um mesmo objeto [Jäschke et al., 2007], devido a diferenças de comportamento, interesses e níveis de conhecimento.

Portanto, a principal motivação para este estudo é que recomendação de tags é um serviço de auxílio ao usuário que apresenta um grande potencial para melhorar a qualidade de tags e, consequentemente, para melhorar a eficácia dos serviços de RI que têm foco nesse tipo de atributo.

1.1 Estado-da-arte

A maioria das estratégias de recomendação de tags centradas no objeto exploram primordialmente tags previamente associadas a objetos da coleção, expandindo um conjunto inicial de tags $\mathcal{I}_o$ associado ao objeto o com termos que co-ocorrem frequentemente com os termos em $\mathcal{I}_o$ [Heymann et al., 2008; Sigurbjörnsson & van Zwol, 2008; Garg & Weber, 2008; Menezes, 2011; Menezes et al., 2010]. Outros métodos não exploram as tags previamente associadas ao objeto alvo da recomendação [Lipczak et al., 2009; Wang et al., 2009; Zhang et al., 2009]. Em vez disso, eles consideram termos extraídos de outros atributos textuais associados ao mesmo objeto. Entretanto, vários atributos textuais, inclusive tags, são criados por usuários finais, logo podem conter uma grande quantidade de *ruído* (por exemplo, palavras com erros ortográficos ou termos não relacionados ao conteúdo) [Figueiredo et al., 2009a; Koutrika et al., 2008]. Assim, é importante filtrar tais termos da lista de recomendações ou pelo menos reduzir a sua importância, favorecendo termos que sejam mais *relevantes* ao objeto alvo. Relevância aqui se refere à capacidade de um termo descrever o conteúdo de um objeto e/ou discriminá-lo de outros objetos, para dar suporte a serviços como busca e classificação, que tipicamente usam tags como fontes de dados [Figueiredo et al., 2009a; Almeida et al., 2010; Figueiredo et al., 2009b]. Com isso em mente, alguns métodos previamente propostos [Sigurbjörnsson & van Zwol, 2008; Lipczak et al., 2009; Wang et al., 2009; Belém et al., 2010a] exploram métricas de relevância, tais como *Term Frequency* (TF) [Baeza-Yates & Ribeiro-Neto, 1999], seja para filtrar candidatos irrelevantes ou dar maior importância aos relevantes.

Em suma, a maioria dos métodos anteriores aborda o problema de recomendação de tags explorando uma combinação das seguintes dimensões: (i) co-ocorrências de tags previamente associadas ao objeto, (ii) termos extraídos de múltiplos atributos textuais, ou (iii) métricas de relevância. Entretanto, que seja do nosso conhecimento, a maioria desses métodos explora no máximo duas dessas dimensões. Em comparação, propomos aqui métodos que exploram essas três dimensões simultaneamente.

Em relação à recomendação personalizada, os trabalhos existentes dão ênfase no histórico de atribuições de tags dos usuários do sistema, geralmente representado por uma matriz tridimensional de valores binários que indicam se um dado usuário atribuiu ou não uma determinada tag a um dado objeto. Os métodos filtragem colaborativa e *FolkRank* [Jäschke et al., 2007], bem como PITF - *Pairwise Interactions Tensor Factorization* [Rendle & Lars, 2010] se encaixam nessa categoria. Entretanto, os trabalhos existentes em recomendação personalizada não exploram por completo as dimensões citadas anteriormente, que podem ser importantes também no contexto de

recomendação personalizada. Poucos são os trabalhos que fazem uso de atributos textuais [Lipczak et al., 2009], ou de co-ocorrências entre tags nesse tipo de recomendação [Rae et al., 2010; Garg & Weber, 2008]. Em nossa abordagem para recomendação de tags personalizada, nós exploramos não apenas o histórico do usuário, mas também as três dimensões citadas anteriormente.

1.2 Objetivos

Os principais objetivos dessa dissertação são listados a seguir:

- Modelagem do problema de recomendação de tags. Nesse trabalho, modelamos o problema de recomendação de tags como um problema de ordenação de múltiplos termos candidatos por relevância para recomendação. Em outras palavras, objetiva-se desenvolver uma função que atribua valores aos termos candidatos, possibilitando a sua ordenação de tal forma que termos que representem recomendações mais relevantes (mais apropriadas para o conteúdo) apareçam em posições mais iniciais que termos que representem recomendações menos relevantes. Logo, a solução para este problema é a função (cujas variáveis são as métricas propostas) que melhor estime a relevância das candidatas à recomendação e, consequentemente, sugira as tags mais adequadas ou mais relevantes a um usuário ou conteúdo.

- Proposição de novas estratégias de recomendação de tags. Objetivamos propor novas técnicas que explorem todas as dimensões citadas anteriormente. Em particular, investigamos e propomos métricas de relevância e as exploramos junto a heurísticas e técnicas de *learning-to-rank*, avaliando os seus benefícios para a tarefa de recomendação centrada no objeto e personalizada.

- Avaliação experimental dos métodos. Pretendemos também comparar o desempenho das estratégias propostas entre si e com métodos de recomendação estado-da-arte em diferentes cenários e diferentes coleções de dados de aplicações populares.

1.3 Contribuições

As principais contribuições dessa dissertação são discutidas a seguir. Parte delas foram publicadas nos seguintes trabalhos: Belém et al. [2011, 2010a,b].

Proposição de novas estratégias de recomendação de tags

Para resolver o problema de recomendação de tags centrada no objeto, nós estendemos métodos tradicionais baseados em co-ocorrências de tags para incluir não apenas padrões de co-ocorrência com tags previamente associadas ao conteúdo, mas também termos extraídos de outros atributos textuais, tais como título e descrição. O conteúdo desses atributos é utilizado para extrair novos termos candidatos. Note que, com essa abordagem, é possível recomendar termos que nunca foram utilizados como tags, o que não é possível com as abordagens baseadas apenas em co-ocorrências de tags. Nós também exploramos várias métricas heurísticas para tentar capturar a *relevância* de cada candidato como recomendação ao objeto alvo. Algumas dessas métricas foram propostas neste trabalho e são responsáveis pelos maiores ganhos obtidos pelos nossos novos métodos de recomendação de tags sobre as estratégias estado-da-arte.

Em particular, nós propomos dez estratégias de recomendação de tags centrada no objeto. Oito delas são heurísticas construídas como extensões de métodos estado-da-arte que exploram co-ocorrência e algumas métricas de relevância. Nossas heurísticas estendem esses métodos explorando múltiplos atributos textuais e incluindo novas métricas de relevância que tentam capturar o quão precisamente um termo *descreve* o conteúdo de um objeto. Elas são simples e eficientes, produzindo ganhos de até 181% em precisão sobre as técnicas originais nas quais são baseadas.

Para resolver o problema de recomendação personalizada, propomos quatro estratégias, sendo duas delas heurísticas que estendem nossas melhores heurísticas de recomendação centrada no objeto, incluindo uma métrica relacionada ao histórico de atribuição de tags do usuário alvo.

Inúmeras são as possíveis heurísticas que podem ser construídas a partir da combinação das métricas de relevância em uma função de recomendação final. Encontrar a "melhor" heurística não é uma tarefa simples, devido ao amplo espaço de busca que, no nosso caso, consiste de todas as funções que possam ser construídas a partir das métricas sugeridas. Assim, nós também investigamos os benefícios do uso de técnicas de *learning-to-rank* (L2R) para "aprender" uma função que combine todas as métricas sugeridas. Em particular, foram propostas duas estratégias baseadas em L2R para recomendação de tags centrada no objeto, que foram também estendidas para incluir personalização. Uma delas explora o método tradicional RankSVM [Joachims, 2006], enquanto a outra é baseada em Programação Genética (PG) [Banzhaf et al., 1998].

PG, em particular, é um método não linear que tem sido empregado com sucesso em problemas em que há um grande espaço de busca e um objetivo a ser otimizado [Banzhaf et al., 1998]. PG foi também aplicada a outras tarefas de Recuperação de

Informação tais como classificação [Zhang et al., 2004], busca [Almeida et al., 2007] e recuperação de imagens [Torres et al., 2009]. Contudo, que seja do nosso conhecimento, somos os primeiros a utilizar PG no problema de recomendação de tags, e essa foi uma das técnicas que apresentou os melhores resultados, ao lado da estratégia baseada em RankSVM, como veremos no Capítulo 6.

Avaliação experimental das estratégias propostas

Nós avaliamos as estratégias de recomendação centrada no objeto, comparando-as com três técnicas estado-da-arte, a saber: Sum^{+}, a melhor função proposta por Sigurbjörnsson & van Zwol [2008], que explora co-ocorrências de tags previamente associadas junto a algumas estatísticas de frequência de tags na coleção; *LATRE* (*Lazy Associative Tag Recommender*) [Menezes, 2011; Menezes et al., 2010], um método bastante recente, eficaz e eficiente baseado em co-ocorrências, e o vencedor do ECML Discovery Challenge 2009 [Lipczak et al., 2009], aqui chamado de CTTR (*Co-occurrence and Text-based Tag Recommender*). CTTR explora o conteúdo de atributos textuais associados a um objeto alvo junto a uma métrica de relevância, porém não considera tags previamente associadas ao objeto alvo. Nossa avaliação foi realizada com dados reais coletados de três aplicações populares da Web 2.0: os sites de compartilhamento de vídeo YouTube e YahooVideo e a estação de rádio online LastFM.

Os resultados indicam que nossa melhor heurística, *LATRE+DP*, que estende o método *LATRE* incorporando múltiplos atributos textuais e uma métrica que tenta capturar o poder descritivo de um termo candidato, superou a melhor entre as três soluções estado-da-arte analisadas em até 40% em precisão e 32% em revocação. Nossas estratégias baseadas em L2R proveram uma melhoria adicional em relação às heurísticas propostas (de até 12% em precisão).

Nossos métodos de recomendação personalizada foram comparados com o *PITF* [Rendle & Lars, 2010] (*Pairwise Interactions Factorization*), uma estratégia estado-da-arte recente. Os resultados indicam que nossa melhor heurística, que estende *LATRE+DP* para incluir personalização através do uso do histórico do usuário, superou a alternativa estado-da-arte em até 328% em precisão e 264% em revocação. Também observamos ganhos modestos da abordagem de L2R sobre as nossas heurísticas (até 5%) no problema de recomendação personalizada. Apesar dos resultados modestos sobre nossas heurísticas, nossa abordagem de L2R provê um arcabouço flexível que pode ser facilmente estendido para incluir outras métricas de relevância ou tratar de outros aspectos do problema de recomendação.

1.4 Organização

O restante deste trabalho está organizado como segue. O Capítulo 2 discute trabalhos relacionados, enquanto o Capítulo 3 introduz os conceitos básicos, a definição do problema e a notação utilizada no restante do trabalho. O Capítulo 4 apresenta as métricas de relevância utilizadas, enquanto o Capítulo 5 discute as estratégias de recomendação de tags propostas. A avaliação experimental é apresentada do Capítulo 6. Por fim, conclusões e direções para trabalhos futuros são discutidas no Capítulo 7.

Capítulo 2

Trabalhos Relacionados

Neste capítulo, discutimos os trabalhos relacionados, dividindo-os em trabalhos que abordam o problema de recomendação de tags (Seções 2.1 e 2.2) e que analisam a qualidade de atributos textuais de objetos da Web 2.0 (Seção 2.3).

2.1 Recomendação de Tags Centrada no Objeto

A maioria das estratégias de recomendação de tags explora termos extraídos dos metadados do objeto e padrões de co-ocorrência de tags para sugerir novas tags. Em particular, algumas delas exploram padrões de co-ocorrência de tags para expandir um conjunto inicial $\mathcal{I}_o$ associado a um objeto o [Sigurbjörnsson & van Zwol, 2008; Garg & Weber, 2008; Heymann et al., 2008; Wu et al., 2009; Krestel et al., 2009; Menezes, 2011; Menezes et al., 2010]. Para isso, Heymann et al. [2008] empregam regras de associação, isto é, implicações do tipo $X \rightarrow y$, onde X é um conjunto de tags e y é uma tag candidata. Embora os autores restrinjam as regras utilizadas por um limiar de confiança, eles não provêem um *ranking* das tags recomendadas. Em contraste, Sigurbjörnsson & van Zwol [2008] propõem o método Sum^+, que explora métricas simples de co-ocorrência de tags (por exemplo, a confiança), aplicando-as sobre todas as tags no conjunto inicial para produzir uma ordenação final de tags candidatas. Algumas das métricas consideradas por ele são relacionadas à frequência de uma tag na coleção e, de certa forma, estão ligadas à "relevância" de uma tag candidata. Em comparação, nós consideramos aqui um conjunto muito mais rico de métricas, incluindo novas métricas baseadas em múltiplos atributos textuais que serão responsáveis, como veremos adiante, pelos maiores ganhos obtidos com as novas estratégias de recomendação propostas.

Devido a questões de eficiência, a maioria dessas estratégias computa co-

ocorrências entre apenas duas tags. Em outras palavras, o antecedente X contém apenas uma tag. Portanto, elas podem ignorar importantes padrões de co-ocorrência. Para tratar esse problema, Menezes [2011]; Menezes et al. [2010] propõem o LATRE (Lazy Associative Tag Recommendation), que computa regras de associação sob demanda, possibilitando a geração eficiente de regras de associação mais complexas e potencialmente melhores, e produzindo ganhos significativos sobre o método proposto por Sigurbjörnsson & van Zwol [2008].

Em uma outra direção, alguns esforços de recomendação de tags não fazem uso de tags previamente associadas a um objeto, dando ênfase em outras fontes de dados [Lipczak et al., 2009; Wang et al., 2009; Zhang et al., 2009; Lu et al., 2009; Katakis et al., 2008]. Por exemplo, Lipczak et al. [2009] propõem um método aqui denominado *adapted CTTR* (*adapted Co-ocurrence and Text based Tag Recommender*), que extrai termos de atributos textuais (por exemplo o título), expandindo-os através de regras de associação e ordenando-os de acordo com a sua taxa de utilização como tags em um conjunto de dados de treino. Este método é uma adaptação da solução vencedora do PKDD Discovery Challenge 2009 na tarefa de recomendação contextual. Um estudo similar é apresentado em [Wang et al., 2009], porém os autores utilizam a métrica *Term Frequency* $\times$ *Inverse Document Frequency* ($TF \times IDF$) [Baeza-Yates & Ribeiro-Neto, 1999], tradicionalmente empregada em tarefas de Recuperação de Informação, para extrair e ordenar por relevância os termos mais importantes do conteúdo textual do objeto. Esforços complementares [Lu et al., 2009; Zhang et al., 2009], propõem recomendar para um objeto o tags que estejam presentes em objetos com conteúdo textual similar ao de o. Um outro esforço adota técnicas de classificação multi-categoria [Katakis et al., 2008], em que cada tag representa uma possível categoria a ser assinalada a um objeto, e termos extraídos do conteúdo de atributos textuais são usados como atributos para o classificador.

Em suma, a maioria dos trabalhos anteriores trata do problema de recomendação contextual de tags explorando no máximo duas das seguintes dimensões: (i) padrões de co-ocorrência entre tags previamente associadas ao objeto alvo, (ii) múltiplos atributos textuais, ou (iii) métricas de relevância. Poucos trabalhos exploram técnicas baseadas em *learning to rank* [Cao et al., 2009; Wu et al., 2009]. Esse tipo de estratégia tenta "aprender" um modelo que permita ordenar tags a partir de um conjunto de métricas de relevância. O aprendizado ocorre com base em um conjunto de treino que contém, para cada tag de exemplo t associada a um objeto o, um rótulo indicando o grau de relevância de t em relação a o e uma lista de valores de métricas de relevância calculadas para t.

Os métodos de recomendação de tags baseados em L2R previamente propostos

são: o trabalho de Cao et al. [2009], baseado no algoritmo RankSVM, e o trabalho de Wu et al. [2009], baseado no algoritmo RankBoost. Entretanto, o primeiro não considera co-ocorrências de tags, e o segundo não considera os atributos textuais de um objeto para a tarefa de recomendação.

Por fim, existem alguns estudos complementares às estratégias baseadas em atributos textuais e co-ocorrências de tags já mencionadas. Wu et al. [2009] acrescentam informações do conteúdo de uma imagem para ordenar tags por relevância no Flickr. Siersdorfer et al. [2009] exploram conteúdo duplicado em vídeos para recomendar tags. Para isso, constrói-se um grafo ponderado cujos vértices são os vídeos e há uma aresta conectando dois vídeos se os seus conteúdos forem similares, ou seja, apresentarem uma certa quantidade de quadros iguais. Assim, recomenda-se a um objeto o tags provenientes de objetos conectados a o nesse grafo. Os autores utilizaram o novo conjunto de tags expandido em classificação automática e agrupamento, obtendo melhorias significativas nos resultados dessas tarefas em relação a não utilizar as recomendações.

Song et al. [2011, 2008] exploram técnicas de agrupamento em um grafo bipartido contendo tags, documentos e palavras como vértices e relações tag-documento e documento-palavra como arestas. O resultado do agrupamento é utilizado para identificar tópicos e quais são as tags mais representativas de cada tópico, que são definidas como aquelas que têm muitas arestas incidentes em vértices do grafo que estejam em um mesmo grupo, e poucas arestas incidentes em vértices de outros grupos. O objeto alvo primeiro é classificado em um ou mais tópicos, e as tags mais representativas de cada tópico são escolhidas como recomendações.

Os métodos Sum^{+}, $LATRE$ e *adapted* $CTTR$ são aqui utilizados como referência e detalhados na Seção 5.1.

2.2 Recomendação Personalizada de Tags

Em relação às estratégias de recomendação personalizada, os trabalhos existentes dão ênfase na utilização do histórico de atribuições de tags pelos usuários do sistema. Esse histórico, conhecido como *folksonomy* [Jäschke et al., 2007], pode ser definido como uma tupla $F = (U, T, O, \mathcal{P})$, onde U, T e O são conjuntos finitos cujos elementos são usuários, tags e objetos, respectivamente, e $\mathcal{P}$ é uma relação ternária entre esses elementos, isto é, $\mathcal{P} \subseteq U \times T \times O$. Assim, cada elemento de $\mathcal{P}$ representa a atribuição de uma tag t a um objeto o por um usuário u. A Figura 2.1 mostra um tensor (ou seja, uma "matriz tridimensional") de valores binários que representa essa relação ternária. O valor "1" indica se o usuário atribuiu a tag ao objeto e o valor "0" indica que não

houve essa atribuição.

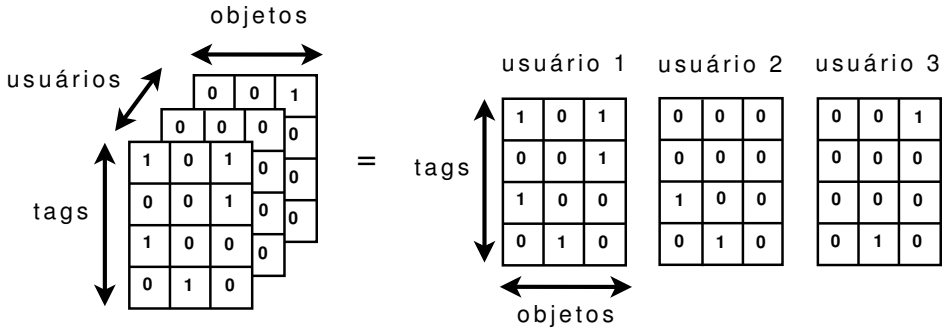


Figura 2.1. Tensor que representa uma *folksonomy* [Rendle & Lars, 2010].

Outra representação possível para a estrutura de uma *folksonomy* é um hipergrafo tripartifo $G = (V, E)$, onde $V = U \cup T \cup O$ é o conjunto de vértices, e $E = \{\{u, o, t\} | (u, o, t) \in \mathcal{P}\}$ é o conjunto de hiperarestas. Em alguns trabalhos [Jäschke et al., 2007; Xu et al., 2006], esse hipergrafo é projetado em um número menor de dimensões, de forma que a relação ternária $\mathcal{P}$ seja convertida em uma relação binária, podendo ser representada por um grafo simples, como exemplificado pela Figura 2.2, e permitindo a aplicação de estratégias de recomendação baseadas em filtragem colaborativa.

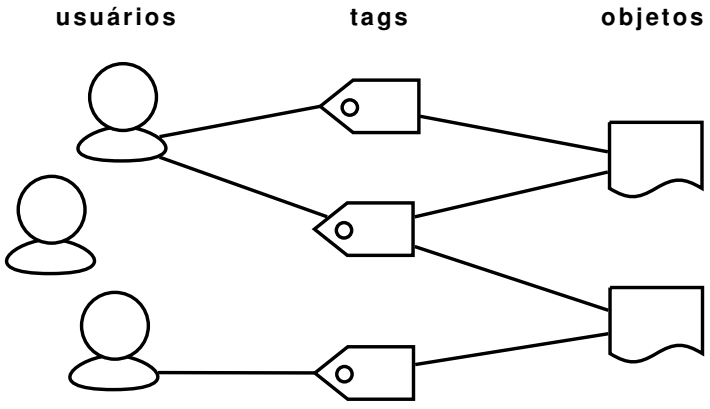


Figura 2.2. Grafo bipartido que representa relações entre usuários, tags e objetos.

A idéia básica de tais estratégias é sugerir itens com base nos interesses de usuários semelhantes [Sarwar et al., 2001]. Para aplicar filtragem colaborativa, Jäschke et al. [2007] geram como projeções de $\mathcal{P}$ duas matrizes de números binários definidas como segue:

$$\pi_{UO}\mathcal{P} \in \{0,1\}^{|U|\times|O|}, \text{ onde } (\pi_{UO}\mathcal{P})_{u,o} = \begin{cases} 1 & \text{se existe } t \in T \text{ tal que } (u,o,t) \in \mathcal{P} \\ 0 & \text{caso contrário} \end{cases}$$

$$\pi_{UT}\mathcal{P} \in \{0,1\}^{|U|\times|T|}, \text{ onde } (\pi_{UT}\mathcal{P})_{u,t} = \begin{cases} 1 & \text{se existe } o \in O \text{ tal que } (u,o,t) \in \mathcal{P} \\ 0 & \text{caso contrário} \end{cases}$$

Assim, preservando a informação de usuário, essas matrizes indicam relações de ocorrência ou de não-ocorrência de objetos com usuários e de tags com usuários, respectivamente. Também é possível definir matrizes semelhantes a estas, mas cujos valores não sejam binários. Por exemplo, o elemento $(\pi_{UT}\mathcal{P})_{u,t}$ pode ser o número de vezes que o usuário u utilizou associou a tag t a qualquer objeto, segundo Clements et al. [2010]. Cada linha dessas matrizes é um vetor que representa os interesses de um usuário. Através dessas matrizes, são computadas as similaridades entre usuários. Um exemplo clássico de medida de similaridade é o cosseno [Jäschke et al., 2007]. Sejam U_i e U_j dois vetores (i-ésima e j-ésima linhas de uma das matrizes $\pi_{UO}\mathcal{P}$ ou $\pi_{UT}\mathcal{P}$) que representam dois usuários quaisquer i e j. O cosseno entre esses dois vetores é definido como:

$$Cos(U_i, U_j) = \frac{U_i \cdot U_j}{||U_i||||U_j||}$$

A idéia dessa métrica é que quanto mais os valores de posições correspondentes de U_i e U_j forem próximos, mais próximo de 1 é o valor de $Cos(U_i, U_j)$, e mais similares são os usuários representados por U_i e U_j. Seguindo a técnica descrita em [Jäschke et al., 2007], o conjunto de recomendações fornecidas a um usuário alvo u é definido como as tags utilizadas pelos k usuários mais similares a u, de acordo a medida de similaridade selecionada.

Outra estratégia de recomendação de tags proposta por Jäschke et al. [2007], denominada *FolkRank*, é uma adaptação do conhecido *PageRank*. Ela mostrou-se superior à estratégia de filtragem colaborativa descrita pelos mesmos autores. O raciocínio por traz desta estratégia 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 um *ranking* final desses vértices, particularmente de tags.

Também baseando-se na *folksonomy* de uma aplicação, Rendle & Lars [2010] ap-

resentam o método de recomendação de tags PITF (*Pairwise Interactions Tensor Factorization*), vencedor do *PKDD Discovery Challenge* de 2009 na tarefa de recomendação baseada em grafos. Nele, o tensor que representa 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. O método $PITF$ é aqui utilizado como referência e detalhado na Seção 5.2. Seguindo uma estratégia diferente, Clements et al. [2010] exploram caminhamentos aleatórios em um grafo de tags, usuários e objetos. Um caminhamento aleatório é um processo estocástico na qual o estado inicial S_n é conhecido, e o próximo estado S_{n+1} é calculado com base em uma certa distribuição de probabilidades. No caso de recomendação personalizada de tags, o caminhamento inicia-se a partir do usuário e objeto alvos da recomendação e segue as arestas de acordo com o seu peso, entre outros fatores. Assim, as tags recomendadas são ordenadas de acordo com as probabilidades obtidas no final do caminhamento, após um determinado número de passos.

Por outro lado, Lipczak et al. [2009] exploram junto ao método *adapted CTTR* citado anteriormente na Seção 2.1 uma métrica simples relacionada ao histórico de uso de tags do usuário alvo para prover recomendações personalizadas. Em uma outra direção, Chirita et al. [2007] não explora esse histórico, baseando-se apenas no conteúdo textual de dados presentes no computador pessoal do usuário.

Outros trabalhos exploram padrões de co-ocorrência de tags também em recomendação personalizada. Por exemplo, Garg & Weber [2008] combinam informação global, ou seja, histórico de atribuições de tags agregado sobre todos os usuários da aplicação, e informação local, isto é, o histórico específico de um usuário. Os autores concluíram que o acréscimo de informação local melhora a eficácia da recomendação. Rae et al. [2010] propõem um método de recomendação denominado $PC{+}SCC{+}CC$ que recomenda tags no Flickr a partir da combinação de diferentes contextos: (1) todas as fotos no sistema; (2) as fotos de um usuário específico; (3) as fotos postadas pelos contatos de um usuário e (4) as fotos de grupos dos quais um usuário é membro. Para cada um desses contextos, um conjunto de tags candidatas é produzida, com base em probabilidades de co-ocorrências. Por fim, os conjuntos são agregados, produzindo uma ordenação de tags final.

Outros esforços exploram relações em redes sociais [Guan et al., 2009; Konstas et al., 2009; Ma et al., 2009; Rae et al., 2010]. Guan et al. [2009] provêm recomendações personalizadas de tags a partir de um grafo bipartido que conecta documentos a tags, e tags a usuários. O algoritmo proposto trata o documento e o usuário alvos da recomendação como uma consulta, e produz uma ordenação das tags através de um método iterativo de otimização que considera a relevância das tags a ambos doc-

umento e usuário. Konstas et al. [2009] exploram relacionamentos de amizade entre usuários do LastFM para melhorar métodos tradicionais de recomendação de objetos baseados apenas em conteúdo.

2.3 Qualidade de Atributos Textuais de Objetos da Web 2.0

Alguns trabalhos anteriores têm o foco no atributo textual *tags* e a sua aplicação para melhorar tarefas relacionadas a Recuperação de Informação (RI), tais como busca, recomendação, agrupamento e indexação [Li et al., 2008; Schenkel et al., 2008; Song et al., 2008; Sigurbjörnsson & van Zwol, 2008; Ramage et al., 2009; Chen et al., 2009]. No entanto, não há um consenso a respeito da qualidade de *tags* e outros conteúdos gerados por usuários. Li et al. [2008] afirmam que *tags* refletem os interesses dos usuários e são de boa qualidade para busca. Por outro lado, Marshall [2009], que fez uma análise semântica manual dos atributos textuais associados a fotos do Flickr, concluiu que *tags* são empregadas majoritariamente para uso pessoal, sendo pouco descritivas do conteúdo a que foram associadas quando comparadas a outros atributos textuais, como o título e a descrição das imagens.

Embora a inspeção manual da qualidade de tags e outros atributos textuais possa produzir boas estimativas da qualidade desses atributos, esse tipo de experimento apresenta um alto custo e um inevitável grau de subjetividade que pode afetar os resultados. Por isso, meios automáticos para avaliação da qualidade de atributos textuais, como as heurísticas *Term Frequency* (TF) e *Inverse Document Fequency* (IDF) constituem uma boa alternativa [Almeida et al., 2010].

Por isso, alguns trabalhos propõem métodos para avaliar automaticamente a qualidade de tags. van Damme et al. [2008] propuseram e avaliaram três métricas de qualidade: (1) frequência de aplicação, definida como o número de usuários que aplicaram a mesma tag a um objeto; (2) concordância, definida como o número de tags que foram selecionadas por mais de 50% dos usuários que "anotaram" o conteúdo; e (3) $TF \times IDF$, uma métrica comumente utilizada em Recuperação de Informação (RI) para atribuir pesos a termos em documentos [Baeza-Yates & Ribeiro-Neto, 1999]. Os resultados do artigo mostram que a frequência de aplicação é a métrica que melhor estima a qualidade de *tags* (também verificado por Sen et al. [2007]), seguida do $TF \times IDF$. O trabalho de van Damme et al. [2008] é um exemplo de como métricas simples e intuitivas podem ser utilizadas para avaliar a qualidade de dados textuais em documentos.

Outra publicação recente que faz uma avaliação quantitativa da qualidade de tags,

é o trabalho de Venetis et al. [2011]. Os autores apresentam métricas que capturam as propriedades estruturais de uma nuvem de tags, uma forma de visualização de tags comumente encontrada em aplicações da Web 2.0. As métricas apresentadas avaliam, sob diferentes aspectos, o quão bem uma nuvem descreve os objetos retornados por uma consulta. Por exemplo, a *cobertura* de uma nuvem de tags é definida como a fração de objetos retornados por uma consulta que foram associados a pelo menos uma das tags da nuvem em relação ao total de objetos retornados. A *coesão* mede a similaridade média entre os objetos que têm uma mesma tag associada. Outra propriedade relevante identificada pelos autores é a *popularidade* de uma tag, ou a quantidade de objetos a que essa tag foi associada. A partir das métricas propostas, Venetis et al. [2011] propuseram algoritmos para selecionar tags e desenvolveram um modelo sintético que representa as tags que um usuário médio escolheria para compor uma nuvem. Os algoritmos de seleção de tags foram avaliados de acordo com esse modelo e com dados reais do Delicious. Os autores concluíram que as métricas estudadas são geralmente pouco correlacionadas entre si, capturando diferentes aspectos de uma nuvem de tags. Os experimentos realizados indicam que os melhores algoritmos de seleção de tags, entre os estudados, são os baseados nas métricas de cobertura e popularidade.

Alguns trabalhos abordam o problema de *tag spam*, isto é, tags não relacionadas ao conteúdo associado que são criadas com o intuito de aumentar a popularidade de objetos ou simplesmente para confundir os usuários. Koutrika et al. [2007, 2008] propõem modelos para tentar explicar como bons usuários (*non-spammers*) e maus usuários (*spammers*) associam *tags* a objetos e como o *tag spam* afeta a recuperação de informação em máquinas de busca. Os autores também descrevem um método para *ranking* de documentos com base na confiabilidade dos usuários.

Alguns estudos de qualidade de atributos de objetos da Web 2.0 compreendem outros campos textuais além de tags. Esse é o caso do trabalho de Bian et al. [2009], cujo foco é em sistemas de perguntas e respostas, tais como o *Yahoo! Answers*[1], nos quais usuários podem postar perguntas e responder às perguntas de outros usuários. A identificação de conteúdo e usuários de boa qualidade é feita através de um algoritmo semi-supervisionado que calcula simultaneamente a reputação de ambos. A idéia é que a reputação dos usuários e a qualidade do conteúdo que eles produzem são frequentemente interligadas por um relacionamento de reforço mútuo. No caso de um sistema de perguntas e respostas, uma resposta tende a ter alta qualidade se o seu conteúdo é compreensível e bem formado, a pergunta correspondente é de qualidade e o usuário que respondeu tem boa reputação no sistema. Ao mesmo tempo, um usuário terá alta

[1] http://answers.yahoo.com

reputação se ele gera respostas e perguntas de alta qualidade. Finalmente, uma pergunta é de alta qualidade se é bem enunciada por um usuário com boa reputação e atrai respostas de alta qualidade.

Em [Figueiredo et al., 2009a; Almeida et al., 2010; Figueiredo et al., 2009b], nós fizemos uma ampla caracterização não apenas de *tags*, como também outros três atributos textuais (título, descrição e comentários), em quatro aplicações diferentes (YouTube, Yahoo Video, LastFM e CiteULike). A qualidade de cada atributo foi analisada sob a perspectiva de fonte de dados para serviços de RI. Para que um atributo textual tenha qualidade sob essa perspectiva, ele deve:

- ter uma *quantidade de conteúdo* suficiente para ser útil;
- fornecer uma boa descrição do conteúdo (*poder descritivo*), o que é importante para serviços que exploram a semântica dos objetos;
- possibilitar a distinção de um objeto de outros (*poder discriminativo*), para tarefas como separar objetos em categorias ou em níveis de relevância nos resultados de uma consulta.

Nesses estudos prévios, concluímos que *tags* e título são os atributos com maior poder descritivo, entretanto tags estão ausentes em 10 a 20% dos objetos, enquanto o título ocorre em praticamente todos eles. Isso motiva estudos em recomendação de tags, que podem melhorar a utilização desse atributo textual. Também verificamos uma grande diversidade de conteúdo entre os atributos textuais, ou seja, cada atributo associado a um objeto o contribui com partes diferentes do conteúdo de o. Isso motiva a proposição de estratégias de recomendação de tags que levam em consideração múltiplos atributos textuais.

Podemos citar também outras caracterizações do uso de tags, que produziram informações úteis para o projeto de sistemas de recomendação de tags [Lipczak & Milios, 2010; Rader & Wash, 2008; Li et al., 2008]. Por exemplo, Lipczak & Milios [2010] verificaram que o título e o histórico pessoal de atribuições de tags são os principais fatores que impactam as decisões do usuário para a escolha de quais tags ele utiliza. Rader & Wash [2008] mostraram que a organização pessoal impacta mais do que as influências sociais na escolha de tags feita por um usuário. Li et al. [2008] verificaram que tags são em geral consistentes com o conteúdo Web a qual são associadas, e que elas tendem a capturar bem os interesses do usuário.

Capítulo 3

Contextualização

Neste capítulo, contextualizamos o problema de recomendação de tags. Iniciamos definindo o objeto alvo da recomendação e seus atributos (Seção 3.1). Em seguida, descrevemos os elementos básicos de um sistema de recomendação de tags (Seção 3.2). Por fim, apresentamos a definição do problema abordado (Seção 3.3).

3.1 Objetos e Atributos na Web 2.0

Utilizamos o termo *objeto* para nos referirmos a uma instância de uma mídia (texto, áudio, imagem, vídeo) em uma dada aplicação da Web 2.0. Há várias fontes de informação relacionadas a um objeto, aqui denominadas *atributos* associados ao dado objeto. Em particular, os atributos podem ser classificados em: atributos de conteúdo, atributos textuais, atributos do perfil dos usuários e atributos sociais associados ao objeto.

Atributos de conteúdo referem-se a fontes de informação que podem ser extraídas do próprio objeto, tais como o histograma de cores de uma imagem. *Atributos textuais*, por sua vez, compreendem os blocos de texto associados ao objeto, que em geral apresentam tópicos ou funcionalidades bem definidos [Fernandes et al., 2007][1]. A Figura 3.1 mostra uma página do YouTube contendo alguns atributos textuais comumente encontrados em aplicações da Web 2.0, como título, *tags*, descrição e comentários de usuários. *Tags*, em particular, são palavras-chave utilizadas para descrever de forma sucinta o conteúdo de um objeto.

Atributos do perfil do usuário são fontes de informação que se referem a características e interações do usuário com a aplicação. Por fim, *atributos sociais* são aqueles

[1]Note que em alguns casos esses dois conjuntos de atributos não são disjuntos. Isso ocorre, por exemplo, quando o próprio objeto é um texto.

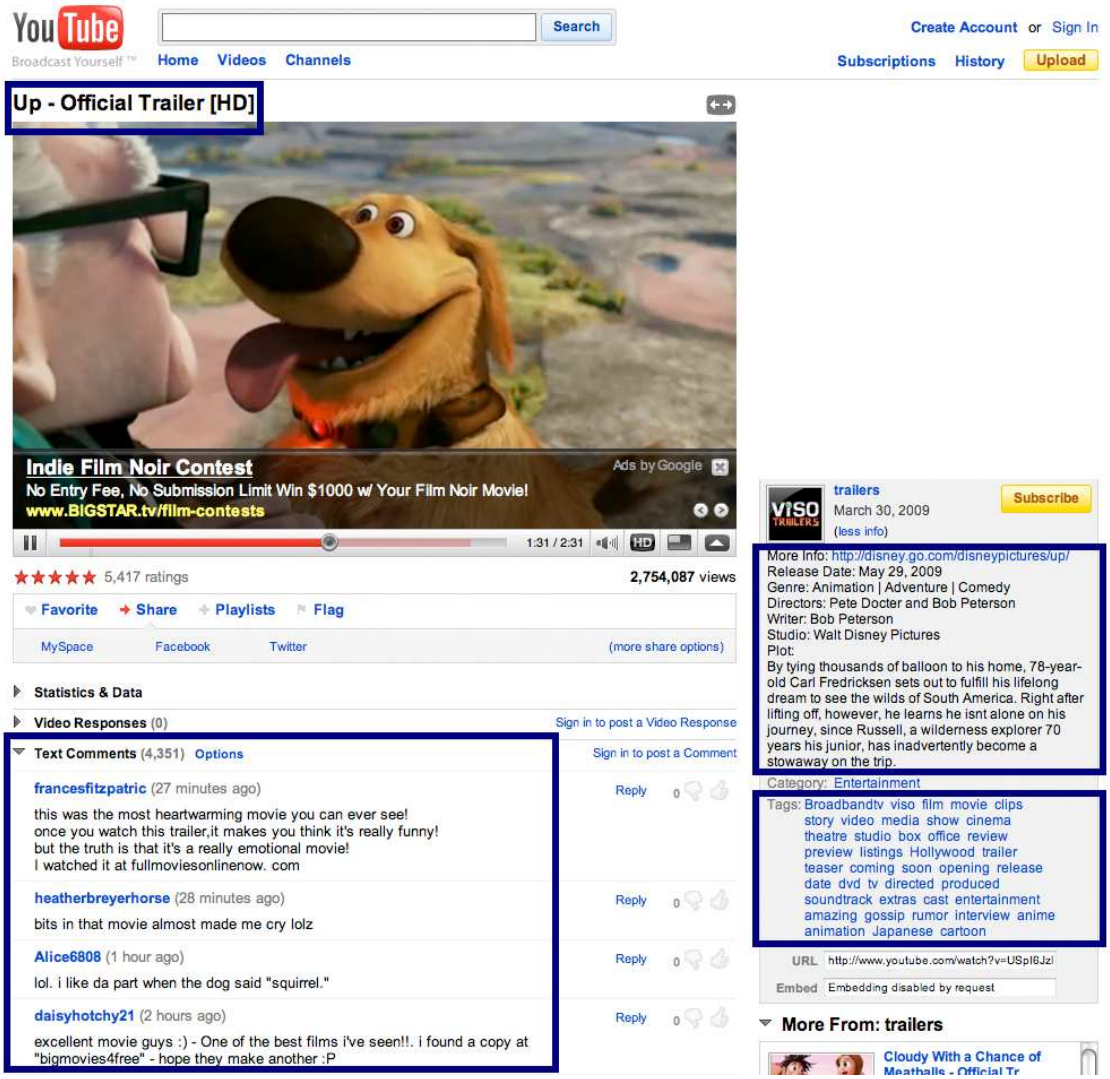


Figura 3.1. Página de um objeto do YouTube.

que refletem o contexto social em que o objeto está inserido, ou seja, quem o criou, quais usuários o acessam e quais interações entre usuários são estabelecidas através do objeto.

No presente trabalho exploramos atributos textuais e de perfil do usuário. Em particular, utilizamos os atributos textuais *título*, *descrição* e *tags*. Como atributo do perfil do usuário, exploramos o *histórico de atribuições de tags de um usuário* em particular, denominado *personomy* [Jäschke et al., 2007], que é utilizado para estimar a relevância de uma tag a esse usuário para prover recomendações personalizadas.

3.2 Sistemas de Recomendação

Nesta seção, definimos o tipo de item a ser recomendado, quem são os alvos da recomendação e quais as fontes de dados básicas de cada estratégia de recomendação adotada.

Um sistema de recomendação pode recomendar diversos tipos de itens, tais como produtos em um site de comércio eletrônico, artigos em uma biblioteca digital, usuários em uma rede social. Nesse trabalho, os itens que estamos interessados em recomendar são tags. Um sistema de recomendação de tags geralmente auxilia usuários, fornecendo a eles uma lista de termos que idealmente descrevam um dado objeto. Portanto, podemos afirmar que o alvo da recomendação de tags é um par usuário-objeto, embora ela possa ou não ser personalizada, ou seja, ela pode levar em conta apenas o conteúdo alvo da recomendação ou considerar o conteúdo e também os interesses do usuário alvo.

Nas estratégias de recomendação de tags que têm como alvo o objeto, exploramos conjuntamente os seguintes aspectos: (1) *co-ocorrências com tags previamente associadas ao objeto alvo*, (2) *múltiplos atributos textuais*, e (3) diversas *métricas de relevância*. O primeiro aspecto se baseia na hipótese de que termos que co-ocorrem frequentemente com as tags de um objeto o são bons candidatos à recomendação para o. Os múltiplos atributos textuais de um objeto, por sua vez, podem conter diversos termos relevantes para descrevê-lo. As métricas de relevância, que serão discutidas no Capítulo 4, podem ser utilizadas para filtrar ou reduzir o peso de termos que representem ruído, dando maior importância aos termos que representem recomendações mais relevantes.

Nas estratégias de recomendação personalizada de tags, exploramos não somente as três dimensões citadas acima, mas também o *histórico de atribuições de tags do usuário alvo*. A premissa básica é que as tags previamente usadas pelo usuário são uma evidência forte de seus interesses.

3.3 Definição do Problema

Seguindo a modelagem de Menezes [2011], sejam U, O e T respectivamente os conjuntos de usuários, objetos e tags presentes em uma aplicação. As estratégias de recomendação de tags aqui descritas exploram como principais fontes de dados:

1. o conjunto de atribuições de tags $\mathcal{P} \subseteq U \times O \times T$, representado por um conjunto de triplas definido como:

$$\mathcal{P} = \{\langle u, o, t\rangle | \text{ usuário } u \text{ atribuiu a tag } t \text{ ao objeto } o\}, \text{ e}$$

2. para cada objeto $o \in O$, um conjunto de atributos textuais (além de tags) $\mathcal{F}_o = \{\mathcal{F}_o^1, \mathcal{F}_o^2, ..., \mathcal{F}_o^n\}$, onde cada elemento $\mathcal{F}_o^i$ é o conjunto de termos no atributo textual i associado ao objeto o.

Sejam $\mathcal{I}_o$ o conjunto de todas as tags previamente associadas ao objeto alvo o, e $\mathcal{I}_{o,u}$ o conjunto de tags associadas por um usuário alvo u ao objeto alvo o, ou seja,

$$\mathcal{I}_o = \{t | \exists u \in U \text{tal que} \langle u, o, t\rangle \in \mathcal{P}\}$$

$$\mathcal{I}_{o,u} = \{t | \langle u, o, t\rangle \in \mathcal{P}\}$$

Assim, definimos dois tipos de estratégias de recomendação de tags:

Recomendação Centrada no Objeto Dados um conjunto de tags de entrada $\mathcal{I}_o$ e um conjunto de atributos textuais $\mathcal{F}_o$ associados ao objeto o alvo da recomendação, gerar um conjunto de tags candidatas $\mathcal{C}_o$ ($\mathcal{C}_o \cap \mathcal{I}_o = \emptyset$), ordenadas de acordo com a sua relevância ao objeto o.

Recomendação Personalizada Dados um conjunto de tags de entrada $\mathcal{I}_{o,u}$ associadas por um usuário u a um objeto o e um conjunto de atributos textuais $\mathcal{F}_o$, gerar um conjunto de tags candidatas $\mathcal{C}_{o,u}$ ($\mathcal{C}_{o,u} \cap \mathcal{I}_{o,u} = \emptyset$) ordenadas de acordo com a sua relevância a ambos usuário u e objeto o.

Note que, na recomendação centrada no objeto, as mesmas recomendações são geradas independentemente do usuário para o qual elas são fornecidas. Ela é importante quando o foco é o objeto e queremos melhorar a qualidade de suas tags, tornando-as mais completas e precisas e, consequentemente, melhorando serviços como busca, indexação e classificação que exploram tags como principal fonte de informação.

Por outro lado, a recomendação personalizada leva em conta o usuário alvo, podendo fornecer resultados distintos para diferentes usuários. Ela é interessante quando queremos não somente melhorar a qualidade de tags e dos serviços de RI nelas baseados, como também facilitar a organização pessoal do conteúdo pelo usuário alvo.

Muitas estratégias de recomendação de tags, e em particular as aqui propostas, exploram padrões de co-ocorrência entre tags associadas a um mesmo objeto em uma coleção de objetos. Para recomendação centrada no objeto, o processo de aprendizado desses padrões é definido como segue. Há um conjunto de treino $\mathcal{D} = \{\langle \mathcal{I}_d, \mathcal{F}_d\rangle\}$,

onde $\mathcal{I}_d$ ($\mathcal{I}_d \neq \emptyset$) contém todas as tags associadas ao objeto d, e $\mathcal{F}_d$ contém os termos dos outros atributos textuais associados a d. Há também um conjunto de teste $\mathcal{O}$, que é uma coleção de objetos $\{\langle \mathcal{I}_o, \mathcal{F}_o, \mathcal{Y}_o \rangle\}$, onde ambos $\mathcal{I}_o$ e $\mathcal{Y}_o$ são conjuntos de tags associadas a o. Entretanto, enquanto as tags em $\mathcal{I}_o$ são conhecidas (e fornecidas como entrada ao recomendador), as tags em $\mathcal{Y}_o$ são desconhecidas pelos métodos de recomendação e, conforme será descrito no Capítulo 6, serão usadas como *gabarito* (isto é, consideradas relevantes) na avaliação dos métodos propostos. Esta divisão das tags de cada objeto de teste nos dois subconjuntos facilita uma avaliação automática das recomendações, como é feito em diversos trabalhos [Garg & Weber, 2008; Rendle & Lars, 2010; Rae et al., 2010; Lipczak et al., 2009] e discutido na Seção 6.2. Esta avaliação simula a recomendação de novas tags ($\mathcal{Y}_o$) a um objeto que já foi anotado com $\mathcal{I}_o$. Da mesma forma, pode haver um conjunto de validação $\mathcal{V}$ usado para parametrização e "aprendizado" de funções de recomendação (vide Seção 5.4). Assim, cada objeto v em $\mathcal{V}$ também é dividido em tags de entrada $\mathcal{I}_v$ e gabarito $\mathcal{Y}_v$.

Para recomendação personalizada, definimos conjuntos de treino, teste e validação ligeiramente diferentes. Ao invés de inferir relacionamentos entre tags atribuídas a um mesmo objeto independentemente de usuários, na recomendação personalizada, estamos interessados em explorar relacionamentos entre tags atribuídas por um mesmo usuário a um mesmo objeto. Logo, o conjunto de treino para recomendação personalizada é definido como $\mathcal{D} = \{\langle \mathcal{I}_{d,u}, \mathcal{F}_d \rangle\}$, onde $\mathcal{I}_{d,u}$ contém todas as tags associadas por um usuário u a um objeto d. $\mathcal{F}_d$ é o mesmo para recomendação personalizada e centrada no objeto. Os elementos do conjunto de objetos de teste $\mathcal{O}$ são tuplas $\langle \mathcal{I}_{o,u}, \mathcal{F}_o, \mathcal{Y}_{o,u} \rangle$, onde $\mathcal{I}_{o,u}$ (tags de entrada) e $\mathcal{Y}_{o,u}$ (gabarito) são conjuntos de tags atribuídas pelo usuário u ao objeto o. Similarmente, cada elemento do conjunto de validação tem também seu conjunto de tags dividido em tags de entrada $\mathcal{I}_{v,u}$ e gabarito $\mathcal{Y}_{v,u}$.

Enfim, modelamos o problema de recomendação de tags como um problema de ordenação de múltiplos termos candidatos por relevância para recomendação. Em outras palavras, objetiva-se desenvolver uma *função de valoração* que atribua valores aos termos candidatos, possibilitando a sua ordenação de tal forma que termos que representem recomendações mais relevantes, isto é, mais apropriadas para o conteúdo (e para o usuário), apareçam em posições mais iniciais que termos que representem recomendações menos relevantes. Na avaliação da qualidade das recomendações, nosso foco é em relevância, ou seja, na quantidade de tags retornadas que são relevantes para o objeto e/ou usuário alvo. Outros aspectos como *novidade* e *diversidade* [Santos et al., 2011] serão explorados em trabalhos futuros.

Note que nossa premissa é que existem algumas tags disponíveis no objeto alvo ($\mathcal{I}_o \neq \emptyset$), como em muitos outros trabalhos [Menezes, 2011; Menezes et al., 2010;

Sigurbjörnsson & van Zwol, 2008; Garg & Weber, 2008; Heymann et al., 2008]. Outro problema relacionado é quando não há esse conjunto inicial de tags. Os métodos propostos aqui podem também ser aplicados nesse problema, mas a avaliação com $\mathcal{I}_o = \emptyset$ foi deixada como trabalho futuro.

Tabela 3.1. Definição do problema de recomendação de tags.

	Recomendação de Tags	
	Centrada no Objeto	**Personalizada**
Entrada	$\mathcal{I}_o$: conjunto de tags previamente associadas ao objeto alvo o	$\mathcal{I}_{o,u}$: conjunto de tags associado pelo usuário u ao objeto o
	$\mathcal{F}_o$: Conjunto de atributos textuais associados ao objeto o	
Saída: lista de candidatos	$\mathcal{C}_o$, $\mathcal{C}_o \cap \mathcal{I}_o = \emptyset$ ordenada por relevância a o	$\mathcal{C}_{o,u}$, $\mathcal{C}_{o,u} \cap \mathcal{I}_{o,u} = \emptyset$ ordenada por relevância a u e o

Capítulo 4

Métricas de Relevância para Recomendação de Tags

Definido o problema, são apresentadas a seguir várias métricas que são utilizadas pelos métodos propostos para estimar a relevância de um termo candidato para recomendação de tags. Algumas delas, como *Sum*, *Stability* e *Term Frequency*, já foram previamente aplicadas para a tarefa de recomendação de tags [Sigurbjörnsson & van Zwol, 2008; Menezes, 2011; Menezes et al., 2010; Heymann et al., 2008]. Outras, como *Inverse Feature Frequency* (IFF) e *Average Feature Spread* AFS, foram propostas por nós para avaliação da qualidade de atributos textuais na Web 2.0 [Figueiredo et al., 2009a], mas ainda não foram exploradas para recomendação de tags.

Cada métrica pode ser categorizada em um dos seguintes grupos:

- *Padrões de co-ocorrência de tags*: As métricas relacionadas a padrões de co-ocorrência tentam estimar a relevância de tags que co-ocorrem com tags previamente associadas ao conteúdo. Segundo Sigurbjörnsson & van Zwol [2008], elas constituem um elemento-chave para a tarefa de recomendação de tags.

- *Poder descritivo*: métricas que estimam o quanto um termo candidato está relacionado ao conteúdo do objeto alvo da recomendação. Isso é importante para serviços que exploram a semântica dos objetos, tais como a própria recomendação de tags, recomendação de conteúdo, busca, etc.

- *Poder discriminativo*: métricas que estimam a capacidade de um termo distinguir um objeto dos demais, uma característica importante para tarefas como classificar

objetos em categorias pré-definidas ou em níveis de relevância nos resultados de uma consulta.

- *Relevância para o usuário*: métricas que capturam a importância de uma tag a um dado usuário. São aqui utilizadas nos métodos de recomendação personalizada de tags.

As próximas seções introduzem as métricas de relevância exploradas.

4.1 Métricas Relacionadas a Padrões de Co-ocorrência de Tags

Em geral, estratégias de recomendação de tags baseadas em co-ocorrências exploram *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 co-ocorrências de X e y no conjunto de treino $\mathcal{D}$, e na **confiança** (θ), definida como a probabilidade condicional de que o termo candidato y seja uma tag associada a um objeto $d \in \mathcal{D}$ dado que todas as tags em X estão também associadas a d.

O número de regras extraídas do conjunto $\mathcal{D}$ pode ser muito grande. Mais ainda, algumas dessas regras podem não ser úteis para recomendação. Por exemplo, se duas palavras co-ocorrem apenas uma vez, isto oferece poucas garantias de que elas sejam relacionadas. Logo, 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 mais confiáveis. Esta seleção pode melhorar tanto a eficiência quanto a eficácia do recomendador.

Para realizar a recomendação de tags para um dado objeto o são selecionadas as regras cujos antecedentes estejam incluídos no conjunto de tags pré-existente $\mathcal{I}$. Para cada termo candidato c que aparece como consequente em qualquer uma das regras selecionadas, sua relevância como tag para o objeto o (e para o usuário u no caso de recomendação personalizada), dado o conjunto inicial de tags $\mathcal{I}$, é estimada como a soma das confianças de todas as regras contendo c, isto é:

$$Sum(c, \mathcal{I}, \ell) = \sum_{X \subseteq \mathcal{I}} \theta(X \rightarrow c), \quad (X \rightarrow c) \in \mathcal{R}, |X| \leq \ell, \tag{4.1}$$

onde $\mathcal{I}$ pode referir-se a $\mathcal{I}_o$ quando se tratar de recomendação centrada no objeto o, ou a $\mathcal{I}_{o,u}$ quando nos referirmos a recomendação personalizada para o objeto o e usuário u. Essa notação é válida aqui e no restante do texto. Por sua vez, $\mathcal{R}$ é o conjunto de regras de associação computado sobre $\mathcal{D}$, dados limiares σ_{min} e θ_{min}, e ℓ é um parâmetro que define o tamanho máximo do antecedente das regras de associação.

4.2 Métricas Relacionadas a Poder Descritivo

Aqui a relevância de uma tag pode ser estimada pela sua capacidade de descrever o conteúdo do objeto alvo. O poder descritivo de um termo candidato c pode ser estimado por várias métricas. Nós exploramos quatro métricas heurísticas. Uma delas, *Term Frequency* (TF), já foi previamente aplicada em recomendação de tags [Wang et al., 2009; Cao et al., 2009]. Porém, esses trabalhos não a utilizaram conjuntamente com co-ocorrência de tags previamente associadas ao objeto alvo da recomendação, como é feito aqui. Uma outra métrica, *Term Spread* (TS) já foi previamente proposta para avaliar a qualidade de atributos textuais [Figueiredo et al., 2009a; Almeida et al., 2010; Figueiredo et al., 2009b] e é aqui utilizada pela primeira vez na tarefa de recomendação de tags. As outras duas métricas, propostas por nós, são derivadas de TF e TS. A seguir definimos as quatro métricas.

Primeiramente, definimos o *Term Spread* de um candidato c em um objeto o, $TS(c, o)$, como o número de atributos textuais (exceto tags, no presente contexto) [1] de o que contêm c [Figueiredo et al., 2009a]:

$$TS(c,o) = \sum_{\mathcal{F}_o^i \in \mathcal{F}_o} j, \text{ onde } j = \begin{cases} 1 & \text{se } c \in \mathcal{F}_o^i \\ 0 & \text{caso contrário} \end{cases} \tag{4.2}$$

A hipótese por trás de $TS(c, o)$ é que quanto maior o número de atributos textuais do objeto o contendo o termo c, mais esse termo é relacionado ao conteúdo de o. Por exemplo, se o termo "Sting" aparece em todas os atributos textuais de um vídeo, há uma alta chance do objeto ser relacionado ao cantor. O valor máximo de TS é dado pelo número de atributos textuais selecionados, desconsiderando tags. No caso, como foram considerados título e descrição, $TS \leq 2$. TS é uma adaptação da métrica proposta por Fernandes et al. [2007], que a utilizaram para modelar a importância de blocos textuais em páginas da Web para melhoria de resultados de buscas.

[1] Tags não são incluídas no cálculo de nenhuma das métricas de poder descritivo, visto que não faz sentido utilizar as próprias tags já presentes no objeto como candidatas a recomendação, tampouco utilizar as tags do gabarito.

A *Term Frequency* de um termo candidato c em um objeto o, $TF(c, o)$, é dada por:

$$TF(c, o) = \sum_{\mathcal{F}_o^i \in \mathcal{F}_o} tf(c, \mathcal{F}_o^i), \tag{4.3}$$

onde $tf(c, \mathcal{F}_o^i)$ é o número de ocorrências de c no atributo textual i de um objeto o. Logo, TF considera todos os campos textuais de o com uma única lista de termos, contando todas as repetições de c nessa lista. Em contraposição, TS considera a estrutura de um objeto, composta de blocos (atributos) textuais, contando o número de blocos que contêm c.

Embora ambos TS e TF tentem capturar o quanto um termo descreve o conteúdo de um objeto, nenhum deles considera que diferentes atributos apresentam, em geral, diferentes capacidades descritivas. Por exemplo, o título pode descrever o conteúdo de um objeto mais precisamente que outros atributos textuais [Figueiredo et al., 2009a; Almeida et al., 2010; Figueiredo et al., 2009b]. Portanto, outras duas métricas, propostas por nós, são construídas a partir do TS e do TF, ponderando o valor de um termo pelo poder descritivo médio dos atributos textuais em que ele aparece.

O poder descritivo médio de um atributo textual $\mathcal{F}^i$ é avaliado pela heurística *Average Feature Spread* (AFS) [Figueiredo et al., 2009a]. Seja o *Feature Instance Spread* de um atributo $\mathcal{F}_o^i$ associado a um objeto o, $FIS(\mathcal{F}_o^i)$, calculado como o TS médio dos termos de $\mathcal{F}_o^i$. Define-se $AFS(\mathcal{F}^i)$ como a média de $FIS(\mathcal{F}_o^i)$ sobre todas as instâncias de $\mathcal{F}^i$ associadas a objetos no conjunto de treino $\mathcal{D}$.

Assim, é possível definir as versões ponderadas das métricas TS e TF como:

$$wTS(c, o) = \sum_{\mathcal{F}_o^i \in \mathcal{F}_o} j, \text{ onde } j = \begin{cases} AFS(\mathcal{F}^i) & \text{se } c \in \mathcal{F}_o^i \\ 0 & \text{caso contrário} \end{cases} \tag{4.4}$$

$$wTF(c, o) = \sum_{\mathcal{F}_o^i \in \mathcal{F}_o} tf(c, \mathcal{F}_o^i) \times AFS(\mathcal{F}^i) \tag{4.5}$$

Note que c pode ser extraído de diferentes atributos textuais. Isso vale para todas as métricas definidas nas seções 4.2-4.4.

4.3 Métricas Relacionadas ao Poder Discriminativo

A relevância de uma tag pode estar também relacionada ao seu poder discriminativo, isto é, à sua capacidade de distinguir um objeto dos demais. Isto é importante, por exemplo, para facilitar a separação de objetos em diferentes tópicos ou categorias, o

que é importante para classificação e recomendação de objetos, ou em diferentes níveis de relevância ou pertinência, o que é útil para busca [Almeida et al., 2010]. Uma das métricas que foram propostas como heurística para capturar o poder discriminativo de um termo candidato é o *Inverse Feature Frequency* (IFF) [Figueiredo et al., 2009a]. O IFF é uma adaptação do tradicional *Inverse Document Frequency* (IDF) que considera a frequência do termo em um atributo textual específico (no nosso caso, tags). Dado o número de elementos no conjunto de treino $|\mathcal{D}|$, o IFF de um termo candidato c é definido como:

$$IFF(c) = log\frac{|\mathcal{D}|+1}{f_c^{tag}+1} \tag{4.6}$$

onde f_c^{tag} é o número de elementos em $\mathcal{D}$ que contêm c como tag.

O valor 1 é somado ao numerador e ao denominador para tratar de novos termos que não aparecem como tag no conjunto de dados de treino. A idéia básica da métrica é que termos muito frequentes tipicamente têm baixo poder discriminativo, por exemplo a palavra "vídeo" na coleção de vídeos do YouTube.

Nota-se que esta essa métrica pode privilegiar termos que não aparecem como tag no conjunto de treino, ou seja, termos para os quais $f_c^{tag} = 0$. Entretanto, neste trabalho, essa métrica é combinada com outras em uma função, utilizando algoritmos de aprendizagem. Logo, seu peso relativo pode ser ajustado, como será descrito na Seção 5.4.

De forma similar, é possível argumentar que termos muito raros também possuem baixo poder discriminativo, pois são muito específicos e podem representar ruído, por exemplo, erros de ortografia, neologismos e palavras desconhecidas. Assim, Sigurbjörnsson & van Zwol [2008] propõem a métrica *Stability* ($Stab$), que privilegia termos com valores intermediários de frequência, reduzindo o valor tanto de termos muito frequentes como o de termos muito raros:

$$Stab(c, k_s) = \frac{k_s}{k_s + |k_s - \log(f_c^{tag})|} \tag{4.7}$$

onde o parâmetro k_s representa a "frequência ideal" de um termo e deve ser ajustado à coleção de dados. $Stab$ também é utilizada aqui para estimar a relevância de um candidato, porém, diferentemente de Sigurbjörnsson & van Zwol [2008], nós a aplicamos não apenas a tags, mas também a termos extraídos de todos os atributos textuais $\mathcal{F}_o$ associados ao objeto alvo o.

Uma terceira métrica relacionada ao poder discriminativo de um termo candidato c é a sua previsibilidade. Heymann et al. [2008] medem essa característica através da

entropia. A entropia de um termo c em relação ao atributo textual tags, $H^{tags}(c)$, é definida como:

$$H^{tags}(c) = -\sum_{(c \to i) \in \mathcal{R}} \theta(c \to i) \log \theta(c \to i) \tag{4.8}$$

onde $\theta(c \to i)$ é a confiança da regra de associação $c \to i$.

Se um termo ocorre consistentemente com certas tags, ele é mais previsível e mais discriminativo, apresentando portanto 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 por Heymann et al. [2008] apenas para avaliar recomendações, aqui ela é empregada em funções de recomendação.

4.4 Métrica Relacionada ao Usuário

Para estimar a relevância de um termo candidato para um dado usuário nós utilizamos uma métrica que contabiliza a frequência com que um usuário utiliza um dado termo como tag. A *User Frequency* (UF) referente a um termo candidato c e um usuário u, $UF(c, u)$, é definida como:

$$UF(c, u) = \frac{N_{c,u}}{N_u}, \tag{4.9}$$

onde $N_{c,u}$ é o número de vezes que o usuário u associou a tag c a algum objeto no conjunto de treino $\mathcal{D}$ e N_u é o número total de vezes que o usuário u associou qualquer tag a qualquer objeto de $\mathcal{D}$. Logo, o raciocínio por trás da métrica UF é que quanto maior a frequência com que um determinado usuário u associa uma tag candidata c, maior a probabilidade de que c seja relevante para u. Esta métrica é computada para todas as tags usadas pelo usuário alvo u em $\mathcal{D}$.

A métrica UF é similar à proposta em [Lipczak et al., 2009]. Entretanto, a métrica proposta naquele trabalho utiliza informação temporal. Como tal informação não está disponível em nossas coleções, nós adaptamos a métrica para considerar todas as atribuições de tags do usuário u no conjunto de treino $\mathcal{D}$.

4.5 Sumário

Tabela 4.1. Métricas de relevância para recomendação de tags.

Nome	Sigla	Equação
Soma	$Sum(c, o, l)$	Eq. 4.1
Term Spread	$TS(c, o)$	Eq. 4.2
Term Frequency	$TF(c, o)$	Eq. 4.3
Weighted Term Spread	$wTS(c, o)$	Eq. 4.4
Weighted Term Frequency	$wTF(c, o)$	Eq. 4.5
Inverse Feature Frequency	$IFF(c, tag)$	Eq. 4.6
Stability	$Stab(c, tag)$	Eq. 4.7
Entropy	$H(c, tag)$	Eq. 4.8
User Frequency	$UF(c, u)$	Eq. 4.9

Capítulo 5

Estratégias de Recomendação de Tags

Este capítulo apresenta as estratégias de recomendação de tags analisadas nesta dissertação. Inicialmente, nas Seções 5.1 e 5.2, são detalhados alguns dos métodos estado-da-arte de recomendação mencionados no Capítulo 2, tomados como referência para avaliação das novas soluções propostas. Estes métodos exploram no máximo duas das seguintes dimensões: co-ocorrência de termos com tags previamente associadas ao objeto alvo, múltiplos atributos textuais e métricas de relevância.

Em seguida, apresentamos as novas soluções propostas, que exploram conjuntamente as três dimensões acima, assim como incluem novas métricas de relevância. Na Seção 5.3, nós descrevemos soluções heurísticas, baseadas em extensões dos métodos existentes. Na Seção 5.4, são apresentadas estratégias que exploram algoritmos de aprendizado (ou *Learning-to-Rank* - L2R) para realizar ordenação de termos candidatos. Extensões dessas últimas estratégias para incluir personalização são discutidas na Seção 5.5.

5.1 Recomendação Centrada no Objeto: Estado-da-Arte

O primeiro método estado-da-arte utilizado como referência é denominado Sum^+ [Sigurbjörnsson & van Zwol, 2008]. Ele explora co-ocorrências de tags e métricas de relevância. Em particular, Sum^+ estende a métrica Sum (Eq. 4.1), ponderando os valores de confiança pela *Stability* dos termos no antecedente e consequente das regras de associação correspondentes, que são aqui geradas pelo algoritmo *Apriori*

[Agrawal & Srikant, 1994]. Dado um candidato c e um conjunto inicial de tags $\mathcal{I}$, Sum^+ é definido como:

$$\begin{aligned} &Sum^+(c, \mathcal{I}, k_x, k_c, k_r) = \\ &\sum_{x \in \mathcal{I}} \theta(x \rightarrow c) \times Stab(x, k_x) \times Stab(c, k_c) \times Rank(c, x, k_r) \end{aligned} \tag{5.1}$$

onde k_x, k_c e k_r são parâmetros de ajuste. O conjunto $\mathcal{I}$ pode se referir a $\mathcal{I}_o$, o conjunto de tags previamente associadas ao objeto o, quando se tratar de recomendação centrada no objeto, ou a $\mathcal{I}_{o,u}$, o conjunto de tags previamente associadas pelo usuário u ao objeto o, no caso de recomendação personalizada. Essa notação é utilizada em todo o restante deste capítulo.

$Rank(c, x, k_r)$ é igual a $k_r/(k_r + p(c, x))$, onde $p(c, x)$ é a posição de c na lista de candidatos ordenados de acordo com a confiança da regra de associação $x \rightarrow c$. De acordo com Sigurbjörnsson & van Zwol [2008], este fator é empregado para que os valores de confiança decaiam de forma mais suave.

Sum^+, como a maioria das estratégias baseadas em co-ocorrências [Sigurbjörnsson & van Zwol, 2008; Garg & Weber, 2008], restringe o tamanho da regras de associação para apenas uma tag no antecedente (isto é, ℓ=1) por questões de eficiência. Em contraste, *LATRE - Lazy Associative Tag Recommender* [Menezes, 2011; Menezes et al., 2010], nosso segundo método de referência, é capaz de gerar de forma mais eficiente regras de associação maiores, fazendo-o sob demanda. Isto difere de outros métodos (por exemplo, o Apriori) que computam todas as regras no conjunto de treino com antecedência, de forma *offline*, incluindo possivelmente um grande número de regras que podem não ser úteis para recomendação no conjunto de teste. *LATRE* atribui a cada candidato c o valor da soma das confianças das regras de associação contendo c como consequente. Ou seja, ele usa a métrica *Sum* (Eq. 4.1) com $\ell \geq 1$, logo explorando apenas padrões de co-ocorrência. Menezes *et al* apresentam duas variações do LATRE, a saber, com e sem calibração. Aqui, optou-se pela estratégia sem calibração, uma vez que ela apresenta menor complexidade. Nós conjecturamos que a calibração irá favorecer tanto o LATRE quanto nossas extensões baseadas nele, apresentadas no capítulo 5.

O terceiro método estado-da-arte considerado é denominado *Co-occurrence and Text based Tag Recommender* (*CTTR*). Ele explora termos extraídos de diferentes atributos textuais e uma métrica de relevância, mas não considera padrões de co-ocorrência com tags previamente associadas ao *objeto alvo*. *Adapted CTTR* é uma adaptação do método vencedor do *ECML Discovery Challenge 2009* [Lipczak et al., 2009]. O método original leva em consideração o histórico de uso de tags de um usuário

ao longo do tempo para prover recomendações personalizadas. Estas estatísticas não são incluídas na nossa implementação principalmente porque as aplicações analizadas não provêm o instante no tempo em que cada usuário aplicou uma tag, informação essencial para utilizar o método original.

Comparamos o *adapted CTTR* com nossas estratégias, pois ele representa um método que explora múltiplos atributos textuais. Com essa comparação, é possível observar os benefícios de utilizarmos co-ocorrências com tags previamente associadas e de aplicarmos as métricas de relevância aqui propostas para recomendação de tags.

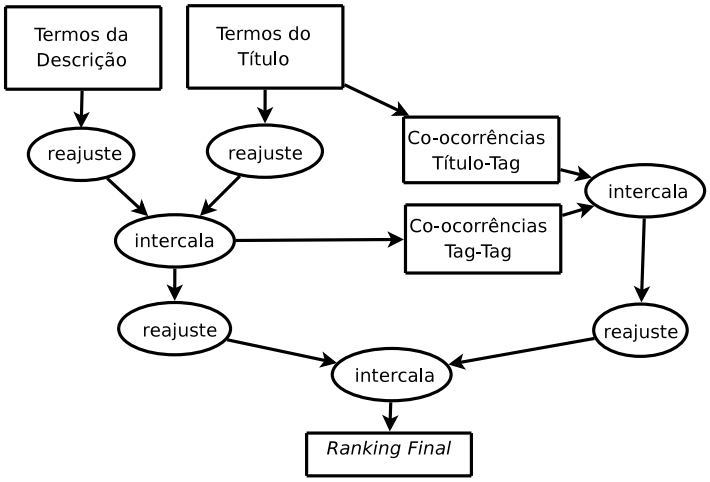


Figura 5.1. *adapted CTTR* [Lipczak et al., 2009].

O funcionamento do *adapted CTTR* é ilustrado na Figura 5.1. 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 extraído de um atributo textual $\mathcal{F}_o^i$ de o, é atribuída uma pontuação s_c^i igual à *taxa de utilização* de c como tag no conjunto de treino. Essa taxa é a fração do número de objetos do treino em que o termo c foi utilizado tanto no atributo textual $\mathcal{F}_d^i$ quanto no conjunto de tags do objeto, em relação ao número total de objetos do treino em que c foi associado ao atributo textual $\mathcal{F}_d^i$.

Em seguida, os conjuntos de 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, o que deve ser refletido na etapa de intercalação. Para isso, os autores propuseram uma estratégia de reajuste de pontuação chamada *leading precision rescorer* para ponderar as diferentes fontes de candidatos (atributos textuais). Esta estratégia associa 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. As pontuações dos

candidatos seguintes são modificadas proporcionalmente. Seja s_j a pontuação antiga do j-ésimo termo candidato na lista de recomendações. A nova pontuação s'_j do j-ésimo termo candidato é reajustada através da seguinte equação:

$$s'_j = \frac{avgP@1 \times s_j}{s_1} \tag{5.2}$$

Depois de reajustar as pontuações, elas são combinadas através de um procedimento de intercalação, definido a seguir. Seja $S_c = \{s_c^1, s_c^2, ..., s_c^n\}$ o conjunto de diferentes pontuações para um mesmo termo candidato c. A função de intercalação é definida como:

$$intercala(S_c) = 1 - \prod_{s_c^j \in S_c} (1 - s_c^j) \tag{5.3}$$

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 utilizados como antecedentes. Para isso, Lipczak et al. [2009] distinguem dois tipos de relacionamentos de co-ocorrência: (1) entre tags, que definem o conjunto de regras de associação $\mathcal{R}_{Tag,Tag}$ e (2) entre termos do título de um objeto e as tags desse mesmo objeto, que definem o conjunto de regras $\mathcal{R}_{Título,Tag}$. Em outras palavras, enquanto os antecedentes das regras em $\mathcal{R}_{Tag,Tag}$ são tags do conjunto de dados de treino, os antecedentes de $\mathcal{R}_{Título,Tag}$ são termos de títulos dos objetos nesse conjunto.

No momento da recomendação, os conjuntos de regras relacionados aos termos extraídos são combinados. Termos do título são usados como antecedentes na equação abaixo para encontrar tags candidatas relacionadas a eles:

$$S_{Título,Tag}(c, o) = 1 - \prod_{x \in \mathcal{F}_o^{título}} (1 - \theta(x \rightarrow c) \times s_x^{título}), \tag{5.4}$$

onde $(x \rightarrow c) \in \mathcal{R}_{Título,Tag}$, e $s_x^{título}$ 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 são utilizados como antecedentes na seguinte equação:

$$S_{Tag,Tag}(c, o) = 1 - \prod_{x \in \bigcup_i \mathcal{F}_o^i} (1 - \theta(x \rightarrow c) \times s_x), \tag{5.5}$$

onde $(x \rightarrow c) \in \mathcal{R}_{Tag,Tag}$, e $s_x = intercala(\{p_x^{título}, p_x^{descrição}\})$ é a pontuação resultante da intercalação título-descrição mencionada acima. Note que s_x pode ser interpretado

como uma métrica de relevância, que captura a importância de um termo nos atributos textuais de um objeto, similarmente ao TS, TF e suas variações.

No passo final, as pontuações obtidas a partir de regras de associação ($S_{Título,Tag}$ e $S_{Tag,Tag}$) e do título e descrição do objeto alvo são reajustadas e intercaladas (com as mesmas Equações 5.2 e 5.3), resultando na ordenação final dos termos por relevância.

5.2 Recomendação Personalizada: Estado-da-Arte

Em nossa avaliação dos métodos de recomendação personalizada, consideramos como referência o método estado-da-arte *Pairwise Interactions Tensor Factorization* (PITF) [Rendle & Lars, 2010], vencedor da tarefa de recomendação baseada em grafo do *PKDD Discovery Challenge 2009*. Em resumo, o PITF modela as interações entre usuários, tags e objetos como um produto de um tensor e matrizes. A partir do conjunto de atribuições de tags $\mathcal{P}$, primeiramente são inferidas restrições de preferências de tags, que estabelecem uma ordem de preferência de tags para cada par objeto-usuário. A idéia é que, para um dado par $\langle$usuário u, objeto $o\rangle$, assume-se que uma tag t_a é mais relevante que uma tag t_b se e somente se $\langle u, o, t_a\rangle \in \mathcal{P}$ e $\langle u, o, t_b\rangle \notin \mathcal{P}$.

Essas restrições são então usadas como dados de treino para um algoritmo de aprendizagem baseado no critério de otimização *Bayesian Personalized Ranking* (BPR), que realiza uma busca em um espaço de soluções através de várias iterações em direção a um gradiente.

Os parâmetros desse método são os seguintes: o número de dimensões da fatoração δ, o número de iterações t, a taxa de aprendizado do BRP λ, o número de tags recomendadas r e o número de restrições de preferência s amostradas de cada par usuário-objeto de treino.

Em trabalhos anteriores [Rendle & Lars, 2010], PITF foi avaliado apenas em conjuntos de dados densos, ou seja, nos quais usuários objetos e tags pouco populares não foram considerados. No nosso caso, todas as estratégias são avaliadas com itens de popularidades diversas.

5.3 Novas Heurísticas para Recomendação de Tags

As soluções heurísticas propostas para recomendação centrada no objeto são extensões dos métodos Sum^+ e $LATRE$ para incluir uma das quatro métricas de poder descritivo apresentadas na Seção 4.2, isto é, TF, TS, wTF ou wTS. Assim, foram propostas oito funções definidas a partir de uma combinação linear da saída do Sum^+ (ou

do $LATRE$) e de uma das quatro métricas. Seja DP (*Descriptive Power*) a métrica de poder descritivo selecionada (isto é, DP equivale a TF, TS, wTF ou wTS), c um termo candidato para o objeto alvo o e $\mathcal{I}$ o conjunto de tags inicial disponível. As heurísticas propostas apresentam as seguintes estruturas gerais:

$$Sum^{+}DP(c, \mathcal{I}, o, k_x, k_c, k_r, \alpha) = \alpha Sum^{+}(c, \mathcal{I}, k_x, k_c, k_r) + (1 - \alpha)DP(c, o) \tag{5.6}$$

$$LATRE + DP(c, \mathcal{I}, o, \ell, \alpha) = \alpha Sum(c, \mathcal{I}, \ell) + (1 - \alpha)DP(c, o) \tag{5.7}$$

O parâmetro α ($0 \leq \alpha \leq 1$) é usado como um fator de ponderação. Note que Sum^{+} e Sum são computados apenas para candidatos gerados a partir de regras de associação, enquanto DP (isto é, TS, TF, wTS ou wTF) é computado também para termos extraídos a partir de outros atributos textuais do objeto alvo o.

Nossas heurísticas para recomendação personalizada são extensões dos métodos $Sum^{+}DP$ e $LATRE{+}DP$ apresentados acima para incluir a métrica *User Frequency* (UF), que estima a frequência de uso de um termo candidato por um usuário (Seção 4.4). Logo, propomos oito novas funções de valoração compostas pela combinação linear entre a saída do $Sum^{+}DP$ (ou do $LATRE{+}DP$) e o valor de UF. Sejam DP a métrica de poder descritivo selecionada (TF, TS, wTF or wTS), e c o termo candidato a ser recomendado para o par $\langle u, o \rangle$. As heurísticas propostas têm as seguintes estruturas gerais:

$$Sum^{+}DP + UF(c, o, u, k_x, k_c, k_r, \alpha, \beta) = \beta Sum^{+}DP(c, o, \mathcal{I}_{o,u}, k_x, k_c, k_r, \alpha) + (1 - \beta)UF(c, u) \tag{5.8}$$

$$LATRE{+}DP{+}UF(c, o, u, \ell, \alpha, \beta) = \beta LATRE{+}DP(c, o, \mathcal{I}_{o,u}, \ell, \alpha) + (1 - \beta)UF(c, u) \tag{5.9}$$

O parâmetro β ($0 \leq \beta \leq 1$) é usado como um fator de ponderação. Note que os valores de Sum^{+} e $LATRE$ são computados sobre candidatos gerados a partir de regras de associação, DP é computada sobre candidatos extraídos de atributos textuais do objeto alvo o, enquanto UF é computada para termos que foram associados como tags pelo usuário u no conjunto de treino $\mathcal{D}$. Logo, um termo candidato c que foi gerado por co-ocorrências ou extraído de algum atributo textual pode não estar incluído no histórico de atribuições de tags do usuário alvo. Nesse caso, o valor de $UF(c, u)$ é 0. De forma similar, um candidato extraído do histórico de u pode não estar contido

nos atributos textuais do objeto alvo o, apresentando valor 0 para a métrica de poder descritivo DP.

5.4 Estratégias Baseadas em Aprendizado

Investigamos os benefícios da aplicação de técnicas de *learning-to-rank* (L2R) no problema de recomendação de tags. A idéia básica é usar tais técnicas para "aprender" uma boa função de valoração dos termos candidatos com base em uma lista $L_{métricas}$ de métricas de relevância. Foram consideradas duas técnicas de L2R: RankSVM e Programação Genética.

Para ambas arbordagens, $L_{métricas}$ inclui Sum, IFF, $Stab$, TS, TF, wTS, wTF, H^{tags} e Sum^{+}, definidas nas Equações 4.1-5.1. Em particular, a métrica Sum é utilizada com dois valores para o tamanho máximo ℓ para o antecedente das regras de associação, a saber, 1 e 3. Sum com $\ell = 3$ também é usado pelo $LATRE$. Também foram acrescentados a $L_{métricas}$ duas outras métricas, chamadas $Vote$ e $Vote^{+}$, propostas por Sigurbjörnsson & van Zwol [2008]. Para um dado candidato c, e um conjunto inicial de tags $\mathcal{I}$, $Vote(c,\mathcal{I})$ é o número de regras de associação cujo antecedente é um termo em $\mathcal{I}$ e cujo consequente é um termo candidato c. $Vote^{+}$ é construído a partir de $Vote$ assim como Sum^{+} é derivado de Sum, isto é, ponderando cada voto pelo valor de $Stab$ de ambos antecedente e consequente. Ou seja:

$$Vote(c,\mathcal{I}) = \sum_{x\in\mathcal{I}} j, \text{ onde } j = \begin{cases} 1, & \text{se}(x \rightarrow c) \in \mathcal{R} \\ 0, & \text{caso contrário} \end{cases} \tag{5.10}$$

$$Vote^{+}(c,\mathcal{I},k_x,k_c,k_r) = \sum_{x\in\mathcal{I}} j \times Stab(x,k_x) \times Stab(c,k_c) \times Rank(c,x,k_r),$$
$$\text{onde } j = \begin{cases} 1, & se \quad x \rightarrow c \in \mathcal{R} \\ 0, & \text{caso contrário} \end{cases} \tag{5.11}$$

Em ambas estratégias, os termos candidatos $\mathcal{C}_o$ de cada objeto o incluem todas os termos gerados pelo $LATRE$ e todos os termos extraídos dos outros atributos textuais. Para cada objeto o e cada termo candidato $c \in \mathcal{C}_o$, são computadas todas as métricas em $L_{métricas}$ nos conjuntos de teste e validação, usando o conjunto de treino $\mathcal{D}$ (por exemplo, para $Stab$, IFF), e os atributos textuais associados a o.

Cada candidato c é então 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 termo 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), baseando-se no conteúdo do gabarito $\mathcal{Y}_v$. 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 6.2). Logo, os rótulos Y_c são atribuídos somente a objetos em $\mathcal{V}$. O modelo aprendido, isto é, a função de valoração $f(M_c)$, é então aplicado a cada termo c candidato à recomendação para o objeto o do conjunto de teste. Os resultados obtidos no Capítulo 6 referem-se, pois, à aplicação das soluções no conjunto de teste.

A seguir, discutimos a aplicação de RankSVM e Programação Genética para recomendação centrada no objeto (Seções 5.4.1 e 5.4.2, respectivamente). Para ambas abordagens, $L_{métricas}$ inclui todas as métricas citadas acima. Extensões das soluções baseadas em aprendizado para recomendação personalizada são discutidas na Seção 5.5.

5.4.1 Recomendação Baseada em RankSVM

RankSVM é baseado no método de classificação estado-da-arte *Support Vector Machine* (SVM) [Joachims, 2006]. Aqui foi empregada a ferramenta SVM-rank[1] para aprender uma função $f(M_c)$=$f(W, M_c)$, onde $W = <w_1, w_2, ...w_m>$ é um vetor de pesos associados às métricas consideradas (isto é, $W \in \mathbb{R}^m$). W é computado por um método de otimização que tenta encontrar um hiperplano, definido por W, que melhor separa os candidatos rotulados como relevantes dos candidatos rotulados como irrelevantes para cada par objeto-candidato. Esses rótulos definem restrições de ordenação de candidatos (termos mais relevantes devem preceder os menos relevantes), o que é utilizado como entrada para o RankSVM. A Figura 5.2 mostra a interpretação geométrica do RankSVM. Os eixos correspondem às métricas (há apenas duas nesse exemplo simples), e cada ponto representa um candidato a recomendação para um determinado objeto.

No momento de recomendação, $f(M_c)$ é usado para ordenar todos os candidatos por relevância, de acordo com suas respectivas distâncias relativas ao hiperplano separador.

O RankSVM tem dois parâmetros principais, a saber, o tipo de função de *kernel*, que indica a estrutura da solução, e o custo j, que controla a penalidade dada a erros de classificação no processo de treino.

O RankSVM foi escolhido por ser um método bastante competitivo para a tarefa de L2R, quando considerados os resultados médios obtidos sobre todas as coleções do

[1] http://www.cs.cornell.edu/People/tj/svm_light/svm_rank.html

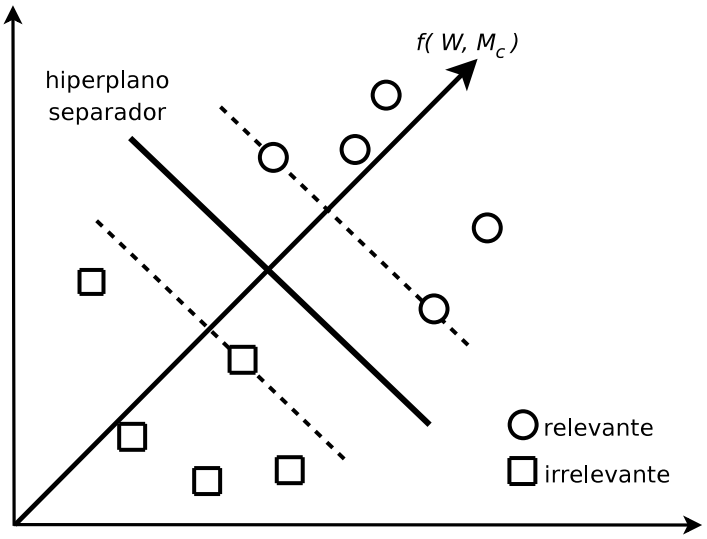


Figura 5.2. Interpretação geométrica do RankSVM.

arcabouço de L2R LETOR 3.0[2], além de estar publicamente disponível.

5.4.2 Recomendação Baseada em Programação Genética

A Programação Genética (PG) foi empregada para encontrar uma boa função de valoração $f(M_c)$ que permita ordenar termos candidatos por relevância. A idéia da PG é evoluir uma população em que cada indivíduo representa uma função de valoração. Essa evolução é inspirada nos mecanismos biológicos de seleção natural e sobrevivência dos indivíduos mais "fortes" e "adaptados". A seguir, é apresentada uma visão geral do arcabouço de PG, e como ele foi aplicado ao problema de recomendação de tags.

Visão Geral do Arcabouço de PG

A PG implementa um mecanismo de busca global no qual se evolui uma população de indivíduos (ou seja, soluções para o problema em questão). Cada indivíduo é tipicamente representado por uma árvore criada a partir de um conjunto de terminais e funções relacionados ao problema alvo. Em cada geração do processo evolucionário, os indivíduos são avaliados de acordo com uma métrica de qualidade, calculada por uma função de *Fitness*, também relacionada ao problema em questão. Assim, o valor da *Fitness* é usado como critério para selecionar os melhores indivíduos, que transmitirão suas características para gerações futuras através de operações tais como cruzamento e mutação. Neste trabalho foi implementado o método de *seleção por torneio*, que

[2]http://research.microsoft.com/en-us/um/beijing/projects/letor/

sorteia, com reposição, k indivíduos da população e escolhe o que apresentar maior valor de *Fitness*.

Enquanto o número de novos indivíduos é menor que o tamanho de população desejado n, dois indivíduos são escolhidos através do método de seleção adotado e, com probabilidade p_c, trocam "material genético" na operação de *cruzamento* para gerar um novo indivíduo. Ou seja, escolhe-se um nó de cada uma das duas árvores (pais) e trocam-se as sub-árvores abaixo desses nós. O papel do cruzamento 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*, para introduzir novos indivíduos na população, aumentando sua diversidade. Isto é útil, por exemplo, para evitar que o processo de busca evolucionário fique preso em mínimos locais. Aqui, a mutação de um indivíduo (ou seja, de uma árvore) é feita selecionando-se aleatoriamente um dos nós da árvore e substituindo-o, juntamente com sua sub-árvore correspondente, por uma nova sub-árvore gerada aleatoriamente, sem exceder a profundidade máxima permitida d.

O processo inteiro, ilustrado na Figura 5.3, é repetido até que o valor de *Fitness* dado como meta seja atingido ou o número máximo de gerações seja alcançado. 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.

A PG é um método não-linear eficaz 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 de Recuperação de Informação, tais como classificação, busca e recuperação de imagens (por exemplo, o trabalho de Yeh et al. [2007]). Entretanto, para o nosso conhecimento, este trabalho é o primeiro que a utiliza no problema de recomendação de tags.

Nós decidimos utilizar a PG neste problema por duas razões principais: (1) em comparação com o RankSVM, a PG explora um paradigma de aprendizagem diferente, que otimiza diretamente uma função alvo (por exemplo, a precisão da recomendação) e (2) tem-se demonstrado que a PG é competitiva com outras técnicas de L2R como o RankSVM e o RankBoost [Yeh et al., 2007; Freund et al., 2003]. De fato, como veremos no Capítulo 6, a PG é mais eficaz que o RankSVM em alguns cenários e os resultados de ambos são estatisticamente empatados em outros[3].

[3]Em testes iniciais, nós verificamos que o processo de treino do RankBoost foi bastante ineficiente para o tamanho de nossas coleções. Por isso, optamos por desconsiderá-lo.

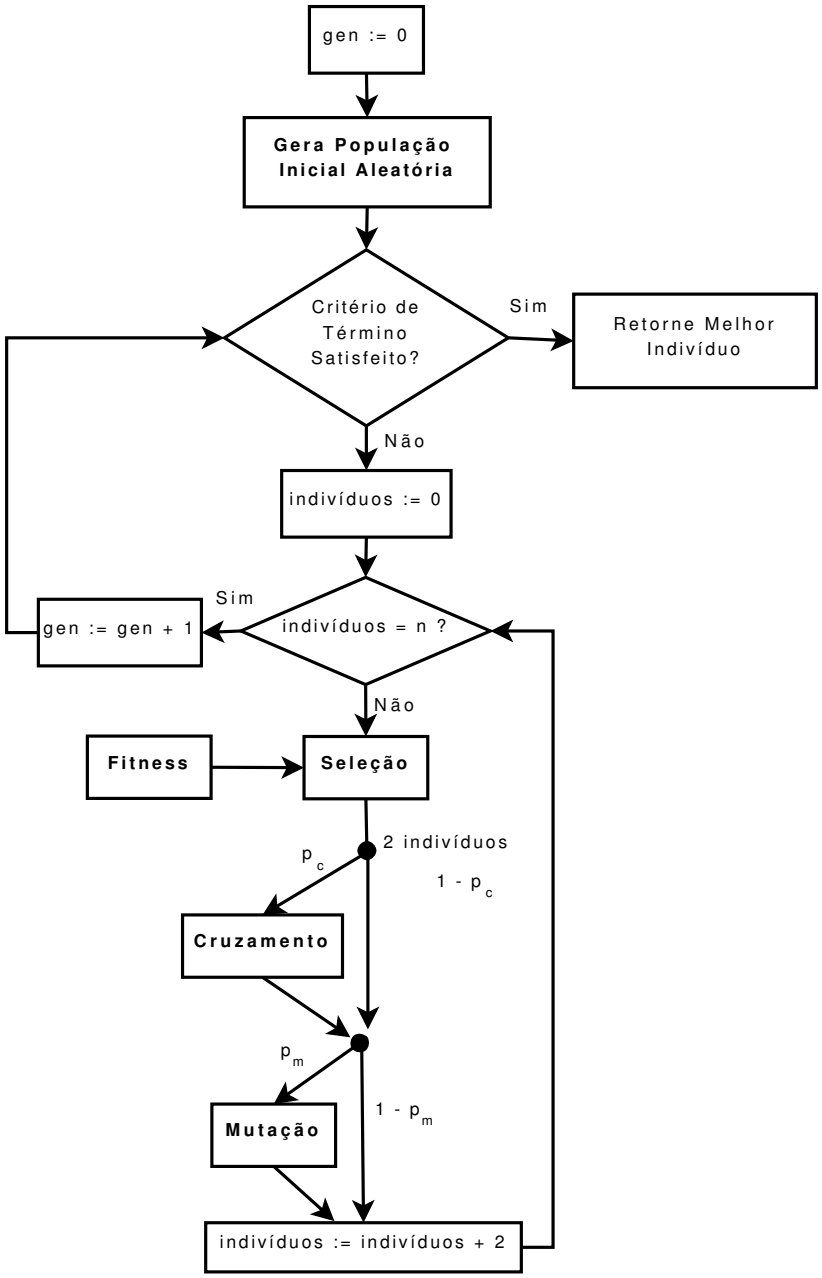


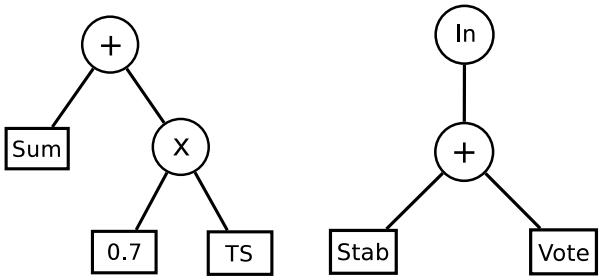
Figura 5.3. Processo Evolucionário da PG.

Programação Genética Aplicada à Recomendação de Tags

Para aplicar PG ao problema de recomendação de tags, precisamos definir a representação de um indivíduo, que nesse caso é uma árvore correspondente à função de valoração de um termo candidato, e uma função de *Fitness*. Uma árvore é composta de terminais (folhas) e não terminais (nós internos). Foram escolhidas as operações de soma ($+$), subtração ($-$), multiplicação ($\times$), divisão ($/$) e logaritmo natural (ln) como

não-terminais. Os terminais são variáveis e constantes uniformemente distribuídas entre 0 e 1. O conjunto de variáveis inclui os valores das métricas na lista $L_{métricas}$. Este conjunto pode ser obtido a partir dos valores no vetor M_c.

Para assegurar a propriedade de fechamento, foram implementadas operações de divisão e logaritmo protegidas, de forma que esses operadores retornem o valor padrão 0 quando suas entradas estiverem fora de seus respectivos domínios. A Figura 5.4 mostra exemplos de possíveis árvores e suas funções correspondentes.



(a) $Sum + 0.7 \times TS$ (b) $ln(Stab + Vote)$

Figura 5.4. Exemplos de árvores.

A *Fitness* de um indivíduo representa a qualidade das recomendações produzidas pela função de valoração correspondente. A qualidade de uma recomendação é aqui avaliada em termos da precisão nas k (com $k = 5$) primeiras posições dos termos recomendados (P@k, conforme definido a seguir), como feito por Menezes et al. [2010]; Sigurbjörnsson & van Zwol [2008][4].

Seja $\mathcal{Y}$ o conjunto de tags relevantes para um dado objeto o, ou seja, $\mathcal{Y} = \mathcal{Y}_o$ para recomendação centrada no objeto, ou para um par objeto-usuário $\langle o, u \rangle$, ou seja, $\mathcal{Y} = \mathcal{Y}_{o,u}$ para recomendação personalizada. Seja $\mathcal{C}$ a lista ordenada de recomendações produzidas pela função gerada pela PG que está sendo avaliada e $\mathcal{C}_k$ os primeiros k elementos de $\mathcal{C}$. $P@k$ é definida como:

$$P@k(\mathcal{C}, \mathcal{Y}) = \frac{|\mathcal{C}_k \cap \mathcal{Y}|}{min(k, |\mathcal{Y}|)} \tag{5.12}$$

A *Fitness* ou qualidade de uma função de valoração $f(M_c)$ é definida como a média de $P@k$ sobre todas as recomendações produzidas por $f(M_c)$ em uma amostra de objetos de tamanho s, extraída do conjunto de validação $\mathcal{V}$.

[4]Também foram testadas outras funções de *Fitness* (por exemplo, Mean Average Precision - MAP), para as quais os resultados qualitativos são os mesmos.

5.5 Extensões para Personalização

As estratégias baseadas em técnicas de *Learning-to-Rank*, descritas na Seção 5.4, podem ser facilmente estendidas para incluir novas métricas de relevância. Em particular, visando estendê-las para realizar recomendação personalizada, foi incluída a métrica UF na lista de métricas $L_{métricas}$ utilizada pelo RankSVM e pela PG. Além disso, todas as tags utilizadas pelo usuário alvo no conjunto de treino $\mathcal{D}$ foram incluídas como candidatas à recomendação. Ou seja, o conjunto de candidatos $\mathcal{C}_{o,u}$ para um dado par $\langle$objeto o, usuário $u\rangle$ inclui todos os termos gerados pelo $LATRE$, todos os termos extraídos dos outros atributos textuais e todos os termos utilizados por u em $\mathcal{D}$. Para cada candidato $c \in \mathcal{C}_{o,u}$, são computadas todas as métricas em $L_{métricas}$ usando o conjunto de treino $\mathcal{D}$, e os atributos textuais associados a o.

Portanto, os algoritmos para recomendação personalizada são os mesmos descritos na Seção anterior. Eles apenas diferem-se no que diz respeito aos dados utilizados e à inclusão da nova métrica e do conjunto adicional de termos candidatos.

5.6 Sumário

Tabela 5.1. Estratégias de recomendação de tags analisadas.

	Recomendação de Tags	
	Centrada no Objeto	**Personalizada**
Métodos de Referência (Seções 5.1-5.2)	Sum^{+} $LATRE$ $CTTR$	$PITF$
Novas Heurísticas (Seção 5.3)	$Sum^{+}DP$ $LATRE{+}DP$ $LATRE{+}DP$	$Sum^{+}DP{+}UF$ $LATRE{+}DP{+}UF$ $LATRE{+}DP{+}UF$
Aprendizado (Seções 5.4-5.5)	$RankSVM$ PG	$RankSVM$ PG

Capítulo 6

Avaliação Experimental

Neste capítulo, apresentamos primeiramente os conjuntos de dados utilizados para avaliar as estratégias de recomendação (Seção 6.1), bem como a metodologia de avaliação utilizada (Seção 6.2). Em seguida, descrevemos como foi feita a parametrização de cada estratégia (Seção 6.4) e discutimos os principais resultados obtidos (Seção 6.5).

6.1 Coleções de Dados

Os métodos de recomendação de tags foram avaliados em coleções de dados coletadas das aplicações LastFM, YouTube e YahooVideo. Cada coleção de dados contém *título*, *tags* e *descrição* dos objetos coletados. Os dados referentes às aplicações LastFM e YouTube também contêm o conjunto de atribuições de tags $\mathcal{P}$. As coletas do LastFM e do YouTube foram realizadas em Agosto de 2009, seguindo a estratégia de amostragem *bola de neve* [Goodman, 1961]. Isto é, começando a partir de um conjunto de usuários (os mais populares) selecionados como sementes, o coletor obtém recursivamente os objetos postados por esses usuários e em seguida segue as ligações sociais para outros usuários, coletando os objetos postados por eles. Nossos conjuntos de dados incluem os atributos textuais associados a 2,8 milhões de artistas do LastFM e a mais de 9 milhões de vídeos do YouTube.

Os dados do YahooVideo foram também obtidos por bola de neve, mas utilizando os vídeos mais populares como sementes e seguindo as ligações para vídeos relacionados. Eles foram obtidos em Outubro de 2008, e contém os atributos textuais de 160.228 objetos[1].

[1]Visite http://vod.dcc.ufmg.br/recc/ para informação sobre disponibilidade dos dados.

A estratégia de amostragem utilizada para o YahooVideo é ligeiramente diferente daquela adotada para YouTube e LastFM. Nessas duas aplicações, optou-se por seguir as ligações sociais entre usuários, pois tais aplicações fornecem informação sobre qual usuário associou uma dada tag a um dado objeto, o que viabiliza a avaliação de métodos de recomendação personalizada. Como o YahooVideo não disponibiliza informação sobre que usuário postou cada tag, não é possível avaliar as estratégias de recomendação personalizada propostas nesta aplicação. Com isso, optou-se por utilizar uma base de dados que já havia sido coletada anteriormente [Figueiredo et al., 2009a] conforme o método descrito acima.

Nós consideramos apenas objetos com atributos textuais em língua inglesa, e utilizamos o algoritmo de remoção de afixos de Porter[2] 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 *stop-words*, 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), visto que tais termos raramente constituem boas recomendações [Sigurbjörnsson & van Zwol, 2008].

6.2 Metodologia de Avaliação

Como discutido no Capítulo 3, foi adotada aqui uma metodologia de avaliação automática similar àquela utilizada em vários trabalhos anteriores [Garg & Weber, 2008; Rendle & Lars, 2010; Menezes et al., 2010; Guan et al., 2009]. Um subconjunto das tags associadas a cada objeto de teste é usado como *gabarito* para a recomendação, ou seja, como tags relevantes para aquele objeto. Como tal, tais tags foram desconsideradas para o cálculo das métricas de relevância. Esta metodologia foi adotada porque o processo de avaliação manual é caro e pode ser afetado pela subjetividade dos julgamentos humanos. Nota-se que os resultados obtidos de acordo com a metodologia proposta representam limites inferiores, já que algumas tags recomendadas podem de fato serem relevantes ao objeto apesar de não aparecerem no gabarito.

A metodologia proposta foi aplicada em recomendação de tags centrada no objeto da seguinte maneira: para cada tupla $\langle \mathcal{I}_o, \mathcal{F}_o, \mathcal{Y}_o \rangle$ referente a um objeto o nos conjuntos de validação ou de teste, nós selecionamos aleatoriamente metade de suas tags para serem incluídas em $\mathcal{I}_o$. A outra metade é incluída em $\mathcal{Y}_o$, o gabarito de o. Também foram utilizados título e descrição como atributos textuais em $\mathcal{F}_o$, para cada objeto o. A avaliação dos métodos de recomendação personalizada foi feita de forma similar:

[2] http://tartarus.org/ martin/PorterStemmer/

para cada tupla $\langle \mathcal{I}_{o,u}, \mathcal{F}_o, \mathcal{Y}_{o,u} \rangle$ nos conjuntos de teste ou de validação, metade de suas tags são incluídas em $\mathcal{I}_{o,u}$ (tags de entrada) e a outra metade em $\mathcal{Y}_{o,u}$ (gabarito). As métricas de avaliação serão descritas na próxima seção.

Tabela 6.1. Subconjuntos de dados. Subscritos p e c referem-se aos dados usados para recomendação centrada no objeto e personalizada, respectivamente.

Coleção	**Menor Subconjunto**		**Subconjunto Médio**		**Maior Subconjunto**	
	#tags/ objeto	**#objetos**	**#tags/ objeto**	**#objetos**	**#tags/ objeto**	**#objetos**
$LastFM_c$	2-4	63.973	5-6	93.080	7-507	78.439
$LastFM_p$	2-4	563.683	5-6	105.104	7-233	68.215
$YahooVideo$	2-6	43.353	7-11	54.143	12-52	49.372
$YouTube$	2-5	2,2 milhões	6-9	1,7 milhão	10-77	1,6 milhão

Cada método de recomendação, centrada no objeto ou personalizada, foi avaliado em dois cenários para cada coleção de dados. No primeiro cenário, cada coleção de dados foi dividida em três subconjuntos com base no número de tags disponíveis por objeto (ou par usuário-objeto), de modo que cada subconjunto contivesse aproximadamente o mesmo número de objetos (exceto para os dados do LastFM, em que houve uma maior quantidade de objetos contendo poucas tags). Os subconjuntos são mostrados na Tabela 6.1, referidos aqui como *menor*, *médio* ou *maior* subconjunto de dados, em referência aos valores de número de tags por objeto ou par usuário-objeto. Cada método foi avaliado separadamente em cada conjunto de dados, e portanto em diferentes quantidades de dados de treino. Em um segundo cenário, *misto*, cada coleção de dados foi considerada em sua totalidade, construindo uma solução para uma coleção de dados mais heterogênea. Portanto, no primeiro cenário, padrões de co-ocorrência e outras métricas de relevância são computados, soluções são "aprendidas" e parâmetros são ajustados para cada subconjunto separadamente, produzindo funções de recomendação mais especializadas. No cenário misto, uma única solução é construída para todos os objetos de uma coleção, a um custo menor.

No primeiro cenário, foram selecionados aleatoriamente 50.000 objetos de cada subconjunto (40,000 para YahooVideo). No segundo cenário, foi utilizada uma amostra que consiste da união de todas as três amostras do primeiro cenário. Em ambos os casos, cada amostra é dividida em 5 partes iguais, que são utilizadas em uma validação cruzada de 5 *folds*. Isto é, três partes são tratadas como conjunto de treino ($\mathcal{D}$), que é utilizado para extrair regras de associação e computar métricas. Uma quarta parte é utilizada como conjunto de validação ($\mathcal{V}$), que por sua vez é empregado para "aprender" as soluções (ou seja, computar a função de *Fitness* no processo evolucionário da PG e aprender o vetor W no RankSVM) e para ajustar parâmetros em todos os métodos. A

última parte é utilizada para teste.

Note que nosso processo de validação cruzada para ambas estratégias baseadas em L2R é ligeiramente diferente do tradicional: o aprendizado da função de valoração e a seleção de parâmetros são feitos no conjunto de validação $\mathcal{V}$. Isso é feito para evitar *overfitting*, que pode ocorrer se as soluções forem aprendidas no mesmo conjunto a partir do qual foram extraídas métricas e regras de associação, isto é, no conjunto de treino $\mathcal{D}$, visto que as métricas derivadas de regras de associação podem ser superestimadas[3].

Assim, 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) as coleções de dados empregadas são grandes o suficiente para um aprendizado efetivo, mesmo quando a validação cruzada é conduzida em $\mathcal{V}$ para parametrização[4], isto é, resultados em experimentos preliminares realizados sobre o conjunto de validação são basicamente idênticos aos reportados no teste com os parâmetros descobertos; e (4) os intervalos de confiança reportados em nossos experimentos (Seção 6.5) são evidências de pouca variabilidade e convergência dos métodos de aprendizado. Além disso, a disponibilidade de um amplo conjunto de dados para gerar regras de associação pode ajudar a aumentar a cobertura das regras, ou seja, mais padrões de co-ocorrência podem ser encontrados, o que gera mais candidatos e valores mais precisos para métricas. Isto beneficia todos os métodos considerados.

6.3 Métricas de Avaliação

Nesta Seção, apresentamos as métricas com as quais avaliamos a qualidade das recomendações. As mesmas foram utilizadas também pelo arcabouço de Programação Genética, que procura maximizar diretamente o valor dessas métricas, como vimos na Seção 5.4.2. Escolhemos três métricas de avaliação amplamente utilizadas na literatura [Baeza-Yates & Ribeiro-Neto, 1999].

Uma dessas métricas, $P@k$, já definida na Seção 5.4.2, com k=5, foi a principal métrica utilizada para avaliar as recomendações. Mas também medimos a revocação e a precisão média (*average precision* ou AP). Revocação é a fração do conjunto de

[3]De fato, esse problema de *overfitting* foi observado em experimentos preliminares, nos quais as funções aprendidas tiveram baixa capacidade de generalização.

[4]Para realizar a parametrização do RankSVM, foi necessário realizar uma validação cruzada dentro do próprio conjunto de validação, ou seja dividir cada conjunto de validação original em dois subconjuntos, um para treino e outro para busca de parâmetros. Isso foi necessário porque, como discutido acima, o conjunto de treino original é utilizado para gerar regras de associação e computar métricas, enquanto o conjunto de validação original é utilizado para aprendizado da função de valoração.

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 do rank [Baeza-Yates & Ribeiro-Neto, 1999]. Seja $\mathcal{Y}$ o conjunto de tags relevantes, seja em relação ao objeto o ($\mathcal{Y} = \mathcal{Y}_o$) para recomendação voltada ao objeto, seja em relação ao par objeto-usuário $\langle o, u \rangle$ ($\mathcal{Y} = \mathcal{Y}_{o,u}$) para recomendação personalizada. Seja $\mathcal{C}$ a lista ordenada de recomendações produzidas pelo método avaliado e $\mathcal{C}_k$ os primeiros k elementos de $\mathcal{C}$. A revocação e AP obtidos com a lista de recomendações $\mathcal{C}$ são definidas como:

$$Revocação(\mathcal{C}, \mathcal{Y}) = \frac{|\mathcal{C} \cap \mathcal{Y}|}{|\mathcal{Y}|} \tag{6.1}$$

$$AP(\mathcal{C}, \mathcal{Y}) = \frac{1}{|\mathcal{Y}|} \sum_{k=1}^{|\mathcal{C}|} (P@k(\mathcal{C}, \mathcal{Y}) \times rel(k)) \tag{6.2}$$

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

6.4 Parametrização

Começamos a avaliação experimental com uma série de experimentos com o conjunto de validação $\mathcal{V}$ para ajustar os valores dos parâmetros de cada método. A seguir, são sumarizados os processos de parametrização das estratégias de recomendação de tags centrada no objeto (Seção 6.4.1) e das estratégias de recomendação personalizada (Seção 6.4.2).

6.4.1 Recomendação Centrada no Objeto

Primeiramente, medimos a eficácia do método Sum^+ em função dos parâmetros k_r, k_x e k_c. Eles foram variados sequencialmente nos valores 1, 5, 10, 20 e 50. Os melhores resultados encontrados foram para $k_r = k_x = k_c = 5$, valor fixado para avaliação do método Sum^+DP, DP igual a TS, TF, wTS ou wTF.

Os parâmetros σ_{min} e θ_{min} diretamente impactam o número de regras de associação gerados e, consequentemente, o tempo de processamento do recomendador. Começamos avaliando a eficácia do método Sum^+ em função desses dois parâmetros, procurando por um bom compromisso entre tempo de processamento e precisão da recomendação. Quanto menor o valor de σ_{min} (ou de θ_{min}), maior o número de regras e portanto maior o tempo de processamento. Os resultados dos experimentos com tais

parâmetros são mostrados na Tabela 6.2 para as três aplicações e subconjuntos de objetos. Eles indicam que, na maioria dos casos, a precisão decresce com o aumento desses dois limiares, particularmente quando uma menor quantidade de tags está disponível, visto que o número de candidatos gerados já é pequeno nesse caso. Contudo, quando o número de tags disponíveis é maior, a escolha desses limiares pode ser mais agressiva sem apresentar um impacto grande na precisão. Assim, nós escolhemos σ_{min} e θ_{min} de modo que a perda de precisão com relação aos resultados em que $\sigma_{min} = \theta_{min} = 0$ fosse abaixo de 3%.

Tabela 6.2. P@5 média de Sum^+ em função de σ_{min} e θ_{min}. Em negrito, melhores resultados (melhor compromisso entre eficiência e precisão).

		LastFM				YahooVideo				YouTube			
σ_{min}	θ_{min}	Misto	Menor	Médio	Maior	Misto	Menor	Médio	Maior	Misto	Menor	Médio	Maior
1	0.00	0.414	.461	.436	.377	.493	.461	.521	.623	.250	.210	.216	.289
1	0.01	0.414	.461	.436	.377	.493	.461	.521	.623	.250	**.210**	.216	.289
1	0.10	0.413	.435	.432	.377	.491	.455	.519	.622	**.245**	.196	**.212**	.288
1	0.20	0.411	.356	.407	.377	.485	.432	.513	.619	.228	.169	.198	**.282**
1	0.30	0.384	.286	.344	.385	.471	.397	.505	.616	.206	.145	.179	.273
1	0.40	0.341	.253	.283	.390	.445	.356	.490	.610	.184	.128	.160	.259
1	0.50	0.260	.213	.243	.378	.414	.322	.470	.599	.376	.118	.146	.244
2	0.00	0.414	.454	.437	.378	.493	.452	.519	.623	.241	.181	.201	.278
2	0.01	0.414	**.454**	.437	.378	.493	**.452**	.519	.623	.241	.181	.201	.278
2	0.10	0.414	.426	**.433**	.378	.491	.447	.517	.621	.236	.164	.197	.277
2	0.20	**0.411**	.344	.406	.378	**.485**	.423	**.512**	.618	.217	.136	.178	.271
2	0.30	0.384	.275	.340	.386	.470	.386	.503	**.615**	.193	.114	.154	.260
2	0.40	0.341	.244	.278	**.391**	.445	.346	.488	.609	.170	.100	.134	.244
2	0.50	0.292	.204	.237	.378	.414	.311	.468	.598	.149	.089	.118	.226

Tabela 6.3. Valores selecionados para parâmetros dos métodos analisados, em cada subconjunto.

Coleção	Conjunto de dados	σ_{min}	θ_{min}	$\alpha(Sum^+DP)$	$\alpha(LATRE{+}DP)$
LastFM	Misto	2	0.2	0.95	0.99
	Menor	2	0.01	0.95	0.95
	Médio	2	0.1	0.95	0.95
	Maior	2	0.4	0.95	0.99
YahooVideo	Misto	2	0.2	0.8	0.9
	Menor	2	0.01	0.8	0.8
	Médio	2	0.2	0.8	0.9
	Maior	2	0.3	0.7	0.95
YouTube	Misto	1	0.1	0.8	0.9
	Menor	1	0.01	0.8	0.7
	Médio	1	0.1	0.8	0.8
	Maior	1	0.2	0.7	0.95

Resultados similares foram obtidos em experimentos com o método $LATRE$. Assim, escolhemos para o LATRE os mesmos valores de σ_{min} e θ_{min} escolhidos para

o Sum^+, para um dado subconjunto de dados e uma dada aplicação. A Tabela 6.3 mostra os valores selecionados para os parâmetros.

Em seguida, avaliamos os métodos Sum^+DP e $LATRE{+}DP$ em função do parâmetro α, que pondera cada uma das métricas da função heurística de recomendação. Os resultados do Sum^+wTS são mostrados na Tabela 6.4. Resultados similares foram obtidos para os métodos $LATRE{+}DP$ e Sum^+DP, com DP igual a TS, TF, wTF ou wTS. Note que, quando α=1, Sum^+DP se reduz a Sum^+ (e $LATRE{+}DP$ se reduz a $LATRE$), exceto pelo fato de que continuam sendo considerados os termos dos atributos textuais ($\mathcal{F}_o^i$) como candidatos, embora o valor de relevância estimado para eles seja 0, deixando-os no final da lista de recomendações. De fato, comparando os valores para α=1 com os valores em negrito da Tabela 6.2, nota-se que as diferenças são marginais na maioria dos casos. As exceções são o subconjunto misto e o subconjunto com menor número de tags por objeto do YouTube. Neles, simplesmente considerar as palavras de $\mathcal{F}_o^i$ já contribui para um aumento de até 12% na precisão.

Tabela 6.4. P@5 média de Sum^+wTS em função de α. Em negrito, melhores resultados.

	LastFM				YahooVideo				YouTube			
α	Misto	Menor	Médio	Maior	Misto	Menor	Médio	Maior	Misto	Menor	Médio	Maior
0.00	0.111	.120	.106	.080	.492	.658	.385	.429	.400	.527	.376	.293
0.50	0.242	.289	.240	.251	.621	.746	.559	.650	.464	.571	.429	.391
0.60	0.264	.303	.265	.292	.658	.761	.609	.694	.477	.575	.439	.411
0.70	0.307	.359	.322	.342	.690	.781	.644	**.721**	.494	.584	.454	**.431**
0.80	0.374	.403	.378	.378	**.707**	**.796**	**.660**	.719	**.502**	**.593**	**.461**	0.427
0.90	0.409	.450	.423	.395	.662	.775	.616	.666	.456	.587	.414	.359
0.95	**0.417**	**.470**	**.439**	**.396**	.573	.674	.555	.634	.368	.534	.316	.304
0.99	0.415	.469	.436	.394	.509	.488	.527	.620	.284	.305	.228	.289
1.00	0.413	.460	.434	.393	.496	.461	.514	.617	.271	.237	.220	.285

Para o $LATRE$, o $LATRE{+}DP$ e as estratégias baseadas em L2R, fixamos $\ell = 3$, como em Menezes et al. [2010]. O parâmetro k_s, utilizado pela métrica *Stab*, foi fixado em 5.

Para a estratégia baseada em RankSVM, foram testados dois tipos de funções de *kernel*: Linear e *Radial Basis Function* (RBF). Foi escolhido o primeiro, visto que ele levou a resultados melhores com um tempo de execução muito menor. Usando validação cruzada em $\mathcal{V}$, nós também variamos o custo j entre 10^{-3} e 10^3, concluindo que a melhor escolha foi j=100 na maioria dos casos. Também tentamos diferentes estratégias para normalizar os vetores de entrada do RankSVM, incluindo divisão pela norma-L2, z-score e o procedimento de normalização utilizado no LETOR [Liu et al., 2007]. Porém, não obtivemos melhorias com nenhuma dessas estratégias de normalização. Dessa forma, os resultados reportados aqui referem-se a dados não normalizados.

Para a estratégia baseada em Programação Genética, foram conduzidos experi-

mentos em $\mathcal{V}$ com tamanhos de população n iguais a 50, 100, e 200. Foi selecionado n=200, visto que a população mais ampla permite uma melhor cobertura do espaço de busca, levando a resultados melhores. Para este tamanho de população, o algoritmo convergiu, ou seja, os valores de *Fitness* pararam de incrementar, antes de 200 gerações, valor atribuído para g. Fixamos k=2, d=7, p_c=0.6 e p_m=0.1, valores frequentemente utilizados na literatura [Banzhaf et al., 1998].

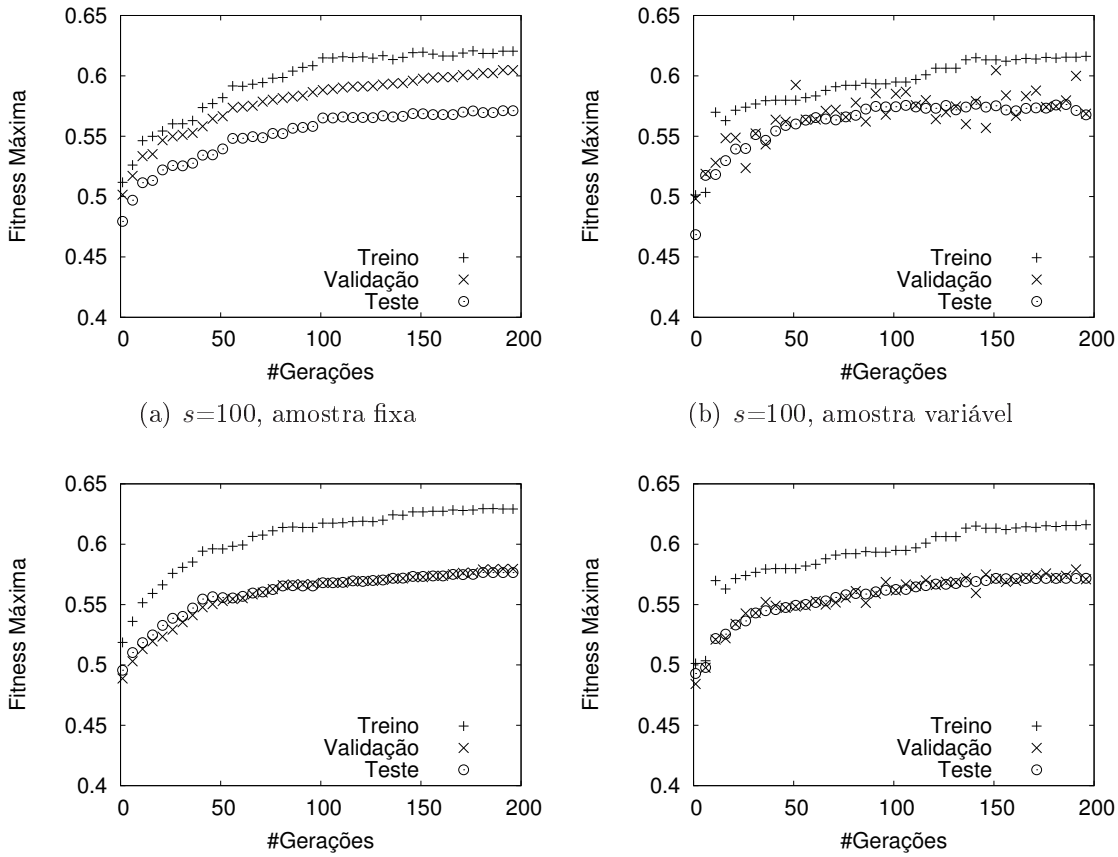
Como o custo de computar a *Fitness* durante o processo evolucionário sobre todos os objetos do conjunto de validação pode ser proibitivo, estudamos a possibilidade de computá-la apenas para uma amostra de tamanho s desses objetos. Foram feitos experimentos preliminares com conjuntos de validação com 4000 objetos cada. Consideramos s=100 e s=500. Testamos uma amostra *fixa*, selecionada no início do processo evolucionário, e também uma amostra *variável* dos objetos que muda a cada 5 gerações. Esta escolha pode ser mais adequada em relação a uma amostra fixa, visto que evita *overfitting*, ou seja, que as funções geradas pela PG sejam especializadas nos conjuntos de validação e treino considerados, não sendo suficientemente gerais para lidar com novos objetos.

A Figura 6.1 mostra, para cada uma das estratégias de amostragem citadas acima, os resultados da $P@5$ ($Fitness$) da melhor função de cada geração, durante o processo evolucionário, aplicados nos conjuntos de treino, teste e validação do YouTube, para o subconjunto com menor número de tags. Resultados similares para os demais subconjuntos e aplicações foram obtidos. Quando comparamos a eficácia da PG para ambos os tamanhos de amostra e estratégia de amostragem variável, notamos pouca diferença entre eles. Logo, escolhemos a amostragem variável com s=100, visto que esta configuração leva a um tempo de treinamento 5 vezes menor que quando s=500.

6.4.2 Recomendação Personalizada

Como nossas estratégias de recomendação personalizada de tags apresentam uma porção oriunda das estratégias de recomendação voltada ao objeto, alguns dos parâmetros utilizados, a saber, k_r, k_x, k_c, k_s, ℓ, σ_{min}, θ_{min} e α, foram fixados nos melhores valores encontrados para os métodos de recomendação centrada no objeto. Nós fixamos a métrica de poder descritivo DP=wTS nos experimentos com as heurísticas $Sum^{+}DP{+}UF$ e $LATRE{+}DP{+}UF$, visto que, como será mostrado na Seção 6.5, essa foi a métrica mais promissora quando utilizada junto aos métodos de recomendação centrada no objeto correspondentes.

Os valores do parâmetro β, usado para ponderação da métrica UF nas heurísticas $Sum^{+}DP{+}UF$ e $LATRE{+}DP{+}UF$, que levaram aos melhores resultados, são



(a) s=100, amostra fixa

(b) s=100, amostra variável

(c) s=500, amostra fixa

(d) s=500, amostra variável

Figura 6.1. Impacto de Amostragem do Conjunto de Validação (YouTube, 2-5 tags por objeto).

mostrados na Tabela 6.6. Esse parâmetro foi variado entre 0 e 1, como mostra a Tabela 6.5. Os resultados permitem comparar a contribuição da métrica UF em tarefas de recomendação personalizada. Por exemplo, considerando a estratégia $Sum^{+}DP{+}UF$, nós observamos um aumento em P@5 de até 12% no LastFM e até 7% no YouTube em relação aos resultados obtidos com β=1, ou seja, quando a métrica UF tem peso igual a 0.

As melhorias foram maiores no LastFM provavelmente devido à natureza colaborativa do uso de tags nessa aplicação, ou seja, qualquer usuário pode atribuir tags a um conteúdo no LastFM. No YouTube, ao contrário, apenas o usuário que postou o objeto pode associar tags a ele. Isso faz com que o histórico de uso de tags no YouTube seja restrito às tags utilizadas pelo usuário nos seus próprios vídeos, reduzindo os benefícios da métrica UF naquela aplicação quando comparados aos benefícios da mesma métrica no LastFM, onde o histórico de atribuições de tags de um usuário pode ser mais amplo. Este fato reflete também as melhores escolhas para β, cujos valores para o LastFM são

menores que para o YouTube, e portanto a importância dada à métrica UF no LastFM é maior que no YouTube.

Tabela 6.5. P@5 média de $LATRE{+}wTS{+}UF$ em função de β. Em negrito, melhores resultados.

	LastFM				YouTube			
β	Misto	Menor	Médio	Maior	Misto	Menor	Médio	Maior
0.0	0.440	0.483	0.397	0.381	0.244	0.215	0.205	0.234
0.1	0.519	0.625	0.492	0.454	0.406	0.528	0.401	0.386
0.2	0.556	0.651	0.541	0.488	0.430	0.544	0.424	0.401
0.3	0.579	0.668	0.574	0.508	0.451	0.560	0.445	0.416
0.4	0.591	0.681	0.593	0.519	0.470	0.576	0.463	0.429
0.5	0.596	0.690	0.604	0.522	0.486	0.589	0.478	0.439
0.6	**0.597**	0.696	**0.608**	**0.522**	0.500	0.602	0.485	0.448
0.7	0.594	**0.700**	0.604	0.520	**0.507**	**0.611**	**0.486**	0.456
0.8	0.590	0.697	0.598	0.517	0.506	0.608	0.482	**0.458**
0.9	0.584	0.688	0.592	0.513	0.500	0.603	0.477	0.455
1.0	0.563	0.668	0.587	0.503	0.488	0.594	0.471	0.448

Os parâmetos do PITF foram ajustados como segue. Nós fixamos a dimensão de fatoração δ=64, a taxa de aprendizado λ=0.05 e o tamanho das amostras s=100, visto que valores similares foram usados em Rendle & Lars [2010]. O número de tags recomendadas foi fixado em um valor relativamente alto (r=100), diferente de Rendle & Lars [2010], visto que nós também consideramos a revocação em nossa avaliação. Como o algoritmo converge antes de 50 iterações em nossos experimentos, este foi o valor escolhido para t.

Por fim, considerando as estratégias de L2R, as melhores escolhas de parâmetros foram as mesmas utilizadas para recomendação centrada no objeto.

Tabela 6.6. Valores selecionados para β em cada heurística de recomendação personalizada, em cada cenário.

Coleção	Conjunto de dados	$Sum^{+}DP{+}UF$	$LATRE{+}DP{+}UF$
LastFM	Misto	0.1	0.6
	Menor	0.2	0.7
	Médio	0.1	0.6
	Maior	0.1	0.6
YouTube	Misto	0.4	0.7
	Menor	0.5	0.7
	Médio	0.4	0.7
	Maior	0.2	0.8

6.5 Principais Resultados para Recomendação Centrada no Objeto

Nesta seção são discutidos os resultados mais relevantes de nossas estratégias de recomendação centrada no objeto, isto é, das 8 heurísticas e dos 2 métodos baseados em L2R, comparando-os com os 3 métodos estado-da-arte. A Tabela 6.7 mostra os resultados de $P@5$ para todos os métodos de recomendação centrada no objeto nos quatro subconjuntos de dados de avaliação (misto, menor, médio e maior). Resultados de revocação e AP são apresentados nas Tabelas 6.9 e 6.8 respectivamente.

Todos os resultados reportados são médias sobre 5 *folds* (conjuntos de teste). Para a estratégia baseada em PG, que é um algoritmo estocástico, cada experimento é repetido 5 vezes, totalizando 25 execuções (5 *folds*, 5 sementes geradoras de números aleatórios). As Tabelas 6.7, 6.9 e 6.8 também mostram intervalos de confiança de 95%. Note que tais intervalos indicam que, para esse nível de confiança, os resultados desviam das médias em no máximo 3%.

Para cada coleção de dados, as tabelas são divididas em 3 blocos: 3 métodos estado-da-arte, 8 novas heurísticas e 2 métodos baseados em L2R. Para cada conjunto de dados, os melhores resultados e empates estatísticos, de acordo com um teste-t com $p < 0.05$, dentro de cada bloco são mostrados em quadros sombreados. Os melhores resultados no geral são mostrados em negrito.

Iniciamos com três observações gerais. A primeira é que as melhorias obtidas com os nossos métodos sobre os métodos estado-da-arte são mais modestas no conjunto de dados do LastFM. Isso é devido a dois fatores principais: (1) há uma tendência de haver uma menor sobreposição entre os conteúdos do título, descrição e tags associados a um mesmo objeto no LastFM, conforme verificado por Figueiredo et al. [2009a], o que leva a uma maior concentração das métricas TS (e wTS) em torno de pequenos valores, tornando difícil a distinção de termos "bons" dos "ruins" usando apenas estas métricas; e (2) o número de tags por objeto tende a ser menor nos dados do LastFM. Por exemplo, 88% dos objetos do LastFM têm menos do que 10 tags, contra 48% e 73% dos objetos do YouTube e do YahooVideo, respectivamente. Esses fatores limitam os benefícios do uso das métricas de poder descritivo propostas e da exploração de padrões de co-ocorrência entre tags previamente associadas aos objetos do LastFM.

A segunda observação é que, embora os resultados obtidos para ambos os cenários, para qualquer subconjunto de dados, não são diretamente comparáveis, visto que os seus conjuntos de treino têm tamanhos diferentes[5], nós notamos que há uma tendência

[5]Note que, usando a mesma quantidade de treino para construir todas as soluções, a

geral de as soluções especializadas produzirem resultados melhores do que uma única função de *ranking*, independentemente da menor quantidade de treino disponível. Por exemplo, o valor de $P@5$ para o $LATRE{+}wTS$ no menor intervalo do YahooVideo é 12% superior ao obtido para o conjunto misto. Os valores correspondentes para revocação e AP são 3% e 9%, respectivamente. Em poucos casos, a solução única supera as especializadas (em até 8% em P@5), possivelmente devido ao maior conjunto de treino.

Uma terceira observação é que os resultados dos métodos de recomendação são melhores no YahooVideo do que nas outras duas aplicações. Isso é explicado principalmente pela maior quantidade de tags disponíveis por objeto nessa aplicação, o que aumenta o limite inferior da avaliação automática e o número de tags disponíveis como entrada, favorecendo todos os métodos de recomendação. Por sua vez, a maior quantidade de tags disponíveis no YahooVideo ocorre provavelmente porque a atribuição de tags nessa aplicação é colaborativa, ou seja, qualquer usuário pode atribuir tags a um objeto.

Quanto ao desempenho relativo dos métodos estado-da-arte, nós verificamos que, em concordância com Menezes et al. [2010], o $LATRE$ supera o Sum^{+} na maioria dos casos. De fato, notamos um ganho de até 25% em $P@5$, 33% em revocação, e 31% em AP. Entretanto, o $CTTR$ se mostra uma boa alternativa ao $LATRE$ em alguns poucos casos, particularmente no YouTube, levando a ganhos em $P@5$, revocação e AP de até 111%, 63% e 75%, respectivamente, observados no menor subconjunto do YouTube.

A seguir são discutidos os resultados das novas heurísticas (Seção 6.5.1) e das estratégias baseadas em L2R (Seção 6.5.2).

6.5.1 Novas Heurísticas

Verificamos que nossas heurísticas produzem ganhos de até 40%, 32% e 62% em P@5, revocação e AP, respectivamente, sobre o melhor método estado-da-arte em cada subconjunto de dados. Assim, introduzir uma métrica de poder descritivo pode melhorar significativamente a eficácia da recomendação, particularmente para objetos do subconjunto de dados com menor número de tags por objeto. Isso ocorre porque, quanto menor a quantidade de tags disponíveis, menores são os benefícios de se explorar

comparação poderia não ser justa também, visto que o conjunto de treino utilizado no cenário misto teria apenas um terço dos objetos de cada subconjunto além de mais heterogeneidade. Assim, nós escolhemos fixar a quantidade total de dado de treino utilizada em ambos os cenários.

Tabela 6.7. Resultados de P@5 média e intervalos de confiança de 95%: melhores resultados em cada bloco (métodos estado-da-arte, heurísticas e estratégias baseadas em L2R) em quadros sombreados. Melhores resultados gerais em negrito.

LastFM				
Estratégia	**Misto**	**Menor**	**Médio**	**Maior**
Sum^+	0.411 ± 0.001	0.454 ± 0.003	0.433 ± 0.004	0.391 ± 0.001
$LATRE$	0.405 ± 0.001	0.465 ± 0.004	0.457 ± 0.005	0.384 ± 0.004
$CTTR$	0.260 ± 0.001	0.285 ± 0.003	0.267 ± 0.002	0.217 ± 0.002
Sum^+TF	0.417 ± 0.001	0.470 ± 0.004	0.436 ± 0.004	0.395 ± 0.001
Sum^+TS	0.418 ± 0.002	0.472 ± 0.005	0.439 ± 0.004	0.396 ± 0.001
Sum^+wTF	0.417 ± 0.001	0.470 ± 0.005	0.436 ± 0.004	0.395 ± 0.001
Sum^+wTS	0.417 ± 0.002	0.470 ± 0.005	0.438 ± 0.004	0.396 ± 0.001
$LATRE$+TF	0.408 ± 0.001	0.479 ± 0.004	0.459 ± 0.005	0.385 ± 0.004
$LATRE$+TS	0.411 ± 0.001	0.487 ± 0.005	0.466 ± 0.005	0.388 ± 0.003
$LATRE$+wTF	0.408 ± 0.001	0.479 ± 0.004	0.459 ± 0.005	0.385 ± 0.004
$LATRE$+wTS	0.411 ± 0.001	0.486 ± 0.005	0.465 ± 0.005	0.388 ± 0.003
PG	**0.433 ± 0.003**	**0.500 ± 0.005**	**0.476 ± 0.002**	**0.434 ± 0.003**
$RankSVM$	0.411 ± 0.002	**0.499 ± 0.007**	0.450 ± 0.005	0.393 ± 0.003

YahooVideo				
Estratégia	**Misto**	**Menor**	**Médio**	**Maior**
Sum^+	0.484 ± 0.003	0.453 ± 0.007	0.511 ± 0.004	0.615 ± 0.004
$LATRE$	0.608 ± 0.003	0.525 ± 0.006	0.619 ± 0.004	0.731 ± 0.006
$CTTR$	0.465 ± 0.004	0.649 ± 0.005	0.407 ± 0.004	0.394 ± 0.003
Sum^+TF	0.643 ± 0.003	0.755 ± 0.002	0.597 ± 0.003	0.663 ± 0.005
Sum^+TS	0.674 ± 0.003	0.764 ± 0.003	0.631 ± 0.003	0.714 ± 0.004
Sum^+wTF	0.666 ± 0.002	0.784 ± 0.003	0.612 ± 0.003	0.673 ± 0.004
Sum^+wTS	0.707 ± 0.002	0.795 ± 0.004	0.660 ± 0.003	0.721 ± 0.004
$LATRE$+TF	0.698 ± 0.002	0.781 ± 0.002	0.673 ± 0.004	0.745 ± 0.007
$LATRE$+TS	0.716 ± 0.003	0.785 ± 0.004	0.710 ± 0.003	0.778 ± 0.007
$LATRE$+wTF	0.729 ± 0.002	0.821 ± 0.004	0.706 ± 0.003	0.754 ± 0.006
$LATRE$+wTS	0.733 ± 0.003	0.818 ± 0.005	0.729 ± 0.003	0.780 ± 0.007
PG	**0.743 ± 0.007**	**0.822 ± 0.004**	**0.734 ± 0.003**	**0.789 ± 0.006**
$RankSVM$	**0.752 ± 0.002**	**0.826 ± 0.003**	0.725 ± 0.003	**0.800 ± 0.005**

YouTube				
Estratégia	**Misto**	**Menor**	**Médio**	**Maior**
Sum^+	0.245 ± 0.002	0.211 ± 0.006	0.212 ± 0.003	0.282 ± 0.004
$LATRE$	0.285 ± 0.004	0.219 ± 0.005	0.242 ± 0.005	0.326 ± 0.006
$CTTR$	0.376 ± 0.002	0.463 ± 0.005	0.337 ± 0.004	0.269 ± 0.002
Sum^+TF	0.462 ± 0.001	0.552 ± 0.004	0.421 ± 0.006	0.403 ± 0.003
Sum^+TS	0.475 ± 0.002	0.560 ± 0.004	0.433 ± 0.005	0.419 ± 0.004
Sum^+wTF	0.490 ± 0.002	0.583 ± 0.003	0.451 ± 0.005	0.416 ± 0.004
Sum^+wTS	0.502 ± 0.003	0.593 ± 0.003	0.461 ± 0.005	0.431 ± 0.004
$LATRE$+TF	0.472 ± 0.002	0.557 ± 0.004	0.436 ± 0.004	0.425 ± 0.006
$LATRE$+TS	0.467 ± 0.003	0.561 ± 0.004	0.445 ± 0.004	0.441 ± 0.007
$LATRE$+wTF	0.503 ± 0.003	**0.596 ± 0.004**	0.464 ± 0.004	0.439 ± 0.005
$LATRE$+wTS	0.489 ± 0.003	0.593 ± 0.003	0.471 ± 0.005	0.450 ± 0.007
PG	0.507 ± 0.001	**0.595 ± 0.003**	0.477 ± 0.002	**0.462 ± 0.003**
$RankSVM$	**0.512 ± 0.002**	**0.601 ± 0.005**	**0.484 ± 0.005**	**0.468 ± 0.003**

Tabela 6.8. Resultados de AP média e intervalos de confiança de 95%: melhores resultados em cada bloco (métodos estado-da-arte, heurísticas e estratégias baseadas em L2R) em quadros sombreados. Melhores resultados gerais em negrito.

LastFM				
Estratégia	**Misto**	**Menor**	**Médio**	**Maior**
Sum^+	0.340 ± 0.001	0.377 ± 0.003	0.366 ± 0.004	0.296 ± 0.003
$LATRE$	0.344 ± 0.001	0.388 ± 0.003	0.386 ± 0.005	0.311 ± 0.005
$CTTR$	0.199 ± 0.002	0.207 ± 0.002	0.211 ± 0.001	0.175 ± 0.002
Sum^+TF	0.352 ± 0.001	0.390 ± 0.003	0.377 ± 0.004	0.312 ± 0.003
Sum^+TS	0.353 ± 0.001	0.389 ± 0.004	0.376 ± 0.004	0.312 ± 0.003
Sum^+wTF	0.352 ± 0.001	0.391 ± 0.003	0.377 ± 0.004	0.312 ± 0.003
Sum^+wTS	0.353 ± 0.001	0.389 ± 0.004	0.376 ± 0.004	0.312 ± 0.003
$LATRE$+TF	0.351 ± 0.001	0.400 ± 0.004	0.396 ± 0.005	0.319 ± 0.005
$LATRE$+TS	0.353 ± 0.002	0.406 ± 0.004	0.397 ± 0.005	0.321 ± 0.005
$LATRE$+wTF	0.351 ± 0.001	0.401 ± 0.004	0.395 ± 0.005	0.319 ± 0.005
$LATRE$+wTS	0.353 ± 0.001	0.406 ± 0.004	0.397 ± 0.005	0.321 ± 0.005
PG	**0.370 ± 0.002**	**0.421 ± 0.003**	**0.411 ± 0.002**	**0.353 ± 0.003**
$RankSVM$	0.354 ± 0.001	0.415 ± 0.004	0.390 ± 0.004	0.326 ± 0.004

YahooVideo				
Estratégia	**Misto**	**Menor**	**Médio**	**Maior**
Sum^+	0.433 ± 0.003	0.417 ± 0.006	0.503 ± 0.004	0.502 ± 0.004
$LATRE$	0.568 ± 0.003	0.496 ± 0.006	0.612 ± 0.004	0.640 ± 0.007
$CTTR$	0.417 ± 0.003	0.483 ± 0.003	0.421 ± 0.004	0.382 ± 0.003
Sum^+TF	0.613 ± 0.003	0.697 ± 0.002	0.625 ± 0.003	0.605 ± 0.005
Sum^+TS	0.635 ± 0.003	0.707 ± 0.002	0.646 ± 0.004	0.633 ± 0.004
Sum^+wTF	0.634 ± 0.003	0.731 ± 0.003	0.639 ± 0.003	0.614 ± 0.005
Sum^+wTS	0.666 ± 0.003	0.748 ± 0.003	0.672 ± 0.003	0.637 ± 0.004
$LATRE$+TF	0.683 ± 0.003	0.730 ± 0.002	0.704 ± 0.004	0.695 ± 0.007
$LATRE$+TS	0.689 ± 0.003	0.740 ± 0.003	0.728 ± 0.004	0.726 ± 0.007
$LATRE$+wTF	0.708 ± 0.003	0.768 ± 0.004	0.727 ± 0.003	0.708 ± 0.007
$LATRE$+wTS	0.705 ± 0.002	0.766 ± 0.003	**0.746 ± 0.004**	0.732 ± 0.007
PG	0.704 ± 0.006	**0.778 ± 0.003**	**0.746 ± 0.002**	0.722 ± 0.008
$RankSVM$	**0.716 ± 0.002**	**0.779 ± 0.002**	**0.747 ± 0.003**	**0.739 ± 0.006**

YouTube				
Estratégia	**Misto**	**Menor**	**Médio**	**Maior**
Sum^+	0.207 ± 0.002	0.177 ± 0.006	0.189 ± 0.003	0.226 ± 0.005
$LATRE$	0.253 ± 0.004	0.189 ± 0.006	0.225 ± 0.004	0.280 ± 0.006
$CTTR$	0.286 ± 0.002	0.331 ± 0.005	0.272 ± 0.005	0.214 ± 0.001
Sum^+TF	0.396 ± 0.001	0.447 ± 0.004	0.384 ± 0.005	0.343 ± 0.003
Sum^+TS	0.413 ± 0.001	0.464 ± 0.004	0.403 ± 0.005	0.356 ± 0.003
Sum^+wTF	0.422 ± 0.002	0.485 ± 0.004	0.410 ± 0.005	0.355 ± 0.003
Sum^+wTS	0.437 ± 0.002	0.507 ± 0.005	0.424 ± 0.006	0.365 ± 0.002
$LATRE$+TF	0.414 ± 0.002	0.456 ± 0.004	0.405 ± 0.004	0.381 ± 0.006
$LATRE$+TS	0.414 ± 0.002	0.469 ± 0.004	0.418 ± 0.004	0.395 ± 0.006
$LATRE$+wTF	0.437 ± 0.003	0.497 ± 0.003	0.426 ± 0.004	0.387 ± 0.005
$LATRE$+wTS	0.432 ± 0.002	0.510 ± 0.004	0.441 ± 0.004	0.401 ± 0.007
PG	**0.450 ± 0.001**	**0.516 ± 0.002**	**0.445 ± 0.002**	0.405 ± 0.003
$RankSVM$	0.446 ± 0.002	**0.512 ± 0.005**	**0.449 ± 0.004**	**0.411 ± 0.003**

Tabela 6.9. Resultados de revocação média e intervalos de confiança de 95%: melhores resultados em cada bloco (métodos estado-da-arte, heurísticas e estratégias baseadas em L2R) em quadros sombreados. Melhores resultados gerais em negrito.

LastFM				
Estratégia	**Misto**	**Menor**	**Médio**	**Maior**
Sum^+	0.523 ± 0.003	0.660 ± 0.003	0.547 ± 0.004	0.430 ± 0.003
$LATRE$	0.620 ± 0.003	0.662 ± 0.003	0.583 ± 0.004	0.570 ± 0.003
$CTTR$	0.762 ± 0.002	0.649 ± 0.004	0.748 ± 0.003	0.894 ± 0.002
Sum^+DP	0.647 ± 0.003	**0.758 ± 0.003**	0.659 ± 0.002	0.552 ± 0.003
$LATRE$+DP	**0.722 ± 0.001**	**0.760 ± 0.003**	**0.686 ± 0.002**	**0.665 ± 0.003**
PG	**0.722 ± 0.000**	**0.760 ± 0.001**	**0.686 ± 0.001**	**0.665 ± 0.001**
$RankSVM$	**0.722 ± 0.001**	**0.760 ± 0.003**	**0.686 ± 0.002**	**0.665 ± 0.003**
YahooVideo				
Estratégia	**Misto**	**Menor**	**Médio**	**Maior**
Sum^+	0.572 ± 0.003	0.701 ± 0.007	0.633 ± 0.003	0.634 ± 0.004
$LATRE$	0.699 ± 0.004	0.702 ± 0.008	0.711 ± 0.003	0.745 ± 0.007
$CTTR$	0.879 ± 0.002	0.882 ± 0.004	0.873 ± 0.005	0.867 ± 0.002
Sum^+DP	0.855 ± 0.002	**0.931 ± 0.003**	0.856 ± 0.004	0.829 ± 0.003
$LATRE$+DP	**0.904 ± 0.003**	**0.931 ± 0.003**	**0.888 ± 0.004**	**0.881 ± 0.005**
PG	**0.904 ± 0.001**	**0.931 ± 0.001**	**0.888 ± 0.001**	**0.881 ± 0.002**
$RankSVM$	**0.904 ± 0.003**	**0.931 ± 0.003**	**0.888 ± 0.004**	**0.881 ± 0.005**
YouTube				
Estratégia	**Misto**	**Menor**	**Médio**	**Maior**
Sum^+	0.365 ± 0.001	0.336 ± 0.005	0.317 ± 0.003	0.387 ± 0.004
$LATRE$	0.434 ± 0.006	0.336 ± 0.005	0.356 ± 0.003	0.467 ± 0.009
$CTTR$	0.559 ± 0.002	0.548 ± 0.005	0.572 ± 0.005	0.670 ± 0.004
Sum^+DP	0.677 ± 0.001	**0.725 ± 0.003**	0.645 ± 0.003	0.609 ± 0.003
$LATRE$+DP	**0.716 ± 0.004**	**0.725 ± 0.003**	**0.667 ± 0.002**	**0.661 ± 0.007**
PG	**0.716 ± 0.001**	**0.725 ± 0.001**	**0.667 ± 0.001**	**0.661 ± 0.002**
$RankSVM$	**0.716 ± 0.004**	**0.725 ± 0.003**	**0.667 ± 0.002**	**0.661 ± 0.007**

apenas co-ocorrência de tags. Nesses casos, nossas heurísticas se beneficiam mais ao explorar métricas de poder descritivo e múltiplos atributos textuais.

Comparando cada nova heurística com o método original na qual ela é baseada (Sum^+ ou $LATRE$), nós obtivemos ganhos de até 181% em P@5, 116% em revocação, e 186% em AP. Por exemplo, Sum^+wTS supera Sum^+ no menor conjunto de dados do YouTube em até 166% em P@5. Como discutido, os ganhos obtidos no LastFM são modestos, isto é, não superam 4.6% em P@5. Em comparação com o $CTTR$, nossos ganhos em P@5, revocação e AP alcançam 98%, 32% e 96%, respectivamente. Estes resultados indicam os benefícios do uso conjunto de co-ocorrências com tags previamente associadas ao objeto alvo e de métricas de poder descritivo.

Entre as heurísticas, a mais promissora é $LATRE$+wTS, que produz os melhores resultados na maioria dos casos. Para chegar a essa conclusão, fazemos duas observações. Primeiro, para qualquer métrica DP de poder descritivo (ou seja, TS, TF,

wTS ou wTF), $LATRE{+}DP$ supera $Sum^{+}DP$ na maioria dos casos. Os ganhos são de até 15% em P@5, 20% em revocação e 15% em AP, o que confirma os benefícios do uso de regras de associação mais complexas. Nos poucos casos em que $Sum^{+}DP$ é melhor que $LATRE{+}DP$, os ganhos em P@5 são abaixo de 3%. Isto ocorre especificamente no LastFM, onde as diferenças entre $Sum^{+}DP$ e $LATRE{+}DP$ são menores em geral, possivelmente devido ao menor número de tags por objeto, que restringe o número de regras mais complexas que podem ser extraídas. Similarmente, as melhorias do $LATRE{+}DP$ sobre $Sum^{+}DP$ tendem a ser maiores para objetos com maior número de tags, particularmente no YouTube e YahooVideo.

Nossa segunda observação é de que, comparando todas as quatro métricas de poder descritivo, o wTS tende a produzir os melhores resultados (ou muito próximos deles), frequentemente seguido de wTF, TS e TF. Em particular, o uso do wTS no $LATRE{+}wTS$, em comparação com a métrica tradicional TF, leva a ganhos de até 8% em P@5 e 12% em AP. Isto ocorre principalmente porque o wTS considera que os objetos são compostos por diferentes atributos textuais, que por sua vez podem apresentar diferentes capacidades descritivas [Figueiredo et al., 2009a]. O TF, por outro lado, tende a favorecer termos muito frequentes, mesmo que eles apareçam em um único atributo textual. Tais termos são frequentemente menos importantes do que os que estão espalhados por diversos atributos textuais.

6.5.2 Estratégias Baseadas em L2R

Quanto às novas estratégias baseadas em L2R, ambos os métodos produzem um ganho adicional sobre as heurísticas. Por exemplo, os ganhos em P@5 e AP obtidos com o RankSVM sobre a melhor heurística, $LATRE{+}wTS$, alcançam 4.8% e 3.2%, respectivamente. Com a estratégia baseada em PG, os ganhos correspondentes alcançam 12% e 10% (por exemplo, no maior intervalo do LastFM). Assim, as melhorias são modestas em geral, embora elas possam ser significativas em alguns casos particulares. Em termos de revocação, os resultados das duas estratégias baseadas em L2R são idênticos aos obtidos com o $LATRE{+}wTS$, visto que as três estratégias geram e recomendam o mesmo conjunto de candidatos para cada instância de teste, diferindo apenas na ordem das recomendações.

Os ganhos são modestos devido ao fato de que ambos PG e RankSVM, apesar de receberem uma lista ampla de métricas, estão explorando majoritariamente as métricas usadas por $LATRE{+}wTS$. A Figura 6.2 mostra, para cada métrica explorada, a porcentagem de funções geradas pela PG no final do processo evolucionário que contêm a dada métrica, considerando todos os subconjuntos e coleções de dados. De fato,

verificamos que a métrica mais frequentemente utilizada pela melhor função gerada no final do processo evolucionário da PG é $Sum(c, \mathcal{I}, 3)$, também usada pelo LATRE. Considerando todos os conjuntos de dados e cenários, $Sum(c, \mathcal{I}, 3)$ ocorre em 95% das funções. A segunda métrica mais utilizada é wTS, que ocorre em 74% das funções. IFF fica em terceiro lugar, ocorrendo em 49% delas. Todas as outras métricas aparecem em menos de 25% das funções. De forma similar, analisamos também o vetor de pesos W aprendido pelo RankSVM, verificando que os maiores pesos, em média, estão de fato associados com $Sum(c, \mathcal{I}, 3)$ e wTS.

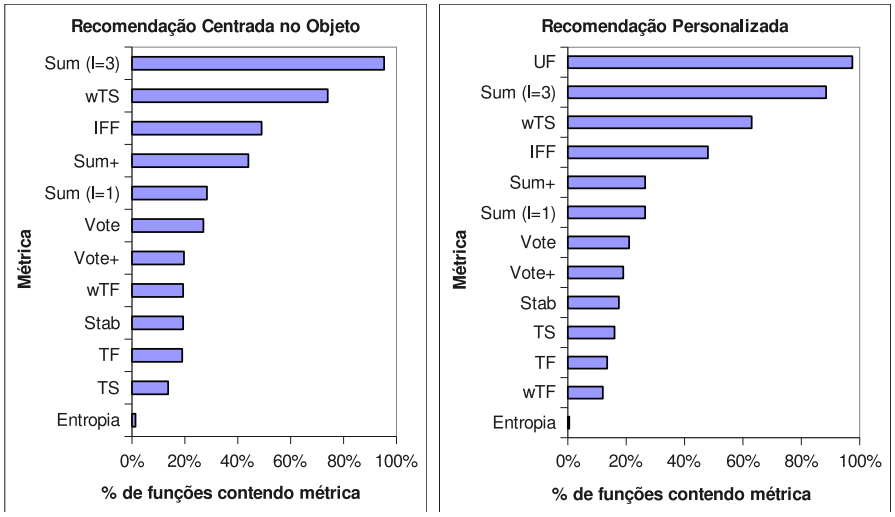


Figura 6.2. Frequência de métricas em funções geradas no final do processo evolucionário da PG.

Embora as estratégias baseadas em L2R produzam ganhos modestos sobre nossas heurísticas, nota-se que elas provêem um arcabouço flexível que pode ser facilmente estendido para incorporar novas métricas e explorar outros aspectos do problema de recomendação de tags, como por exemplo, a personalização, como já feito aqui. Também vale ser ressaltado que, após a etapa de treino, que é executada em modo *offline*, o uso das funções geradas pela PG ou RankSVM, em tempo de recomendação, não implica em um tempo de processamento extra significativo em relação à melhor heurística. Um possível custo para todas as estratégias seria a geração de regras sob demanda pelo $LATRE$. Porém, o tempo médio de processamento do $LATRE$ é adequado para recomendação em tempo real. Menezes et al. [2010] mostram que, para objetos com poucas tags, que correspondem aos nossos subconjuntos de dados com menor número de tags, o tempo necessário para recomendação é da ordem de milisegundos. Em objetos com um número grande de tags, os autores obtiveram um tempo de recomendação inferior a 1.8 segundos.

Entre as duas estratégias baseadas em L2R, o melhor método depende da base de dados. No LastFM, a PG leva a resultados melhores, com ganhos de até 10% em P@5 e 8% em AP. Já no YahooVideo e YouTube, há um empate estatístico na maioria dos conjuntos de dados com uma ligeira vantagem de um método sobre o outro em poucos casos.

Comparando ambas as soluções em termos de tempo de treino, nós verificamos também que a técnica mais eficiente depende do conjunto de dados. Por exemplo, a etapa de treino da PG requer, em média, 63.5, 50.4 e 30.6 minutos nos dados do cenário misto do LastFM, YouTube e YahooVideo, respectivamente, quando executados em um processador Intel 2.83GHz Quad-Core com 8GB de RAM. Os números correspondentes para o RankSVM são 100.8, 28.3 e (surpreendentemente) 0.81 minutos. Também nota-se que, exceto na coleção de dados do YahooVideo, o tempo de treino do RankSVM exibe uma variabilidade muito maior sobre os diferentes *folds* do que a PG. Foram computados intervalos de confiança de 95% para os tempos de treino medidos sobre um número fixo de *folds*, e observado que os tempos de treino da PG desviam de seus valores médios em até 10%. RankSVM, por sua vez, tem variação de até 48% da média com esse nível de confiança. Em outras palavras, o tempo de treino do RankSVM parece ser muito mais sensível às inerentes dificuldades dos dados de entrada.

6.6 Principais Resultados para Recomendação Personalizada

Nesta seção são discutidos os principais resultados das estratégias de recomendação personalizada propostas, incluindo 2 heurísticas e 2 soluções baseadas em L2R, comparando-as com o método estado-da-arte PITF. A Tabela 6.10 mostra os resultados de P@5 para cada estratégia e subconjunto de dados. Resultados de revocação e AP são apresentados nas Tabelas 6.12 e 6.11, respectivamente.

Como foi feito para recomendação centrada no objeto, todos os resultados reportados são médias sobre 5 *folds* (conjuntos de teste), enquanto os resultados da PG são médias sobre 25 execuções (5 *folds*, 5 sementes). As Tabelas 6.10, 6.12 e 6.11 também mostram intervalos de confiança de 95%, indicando que, com essa confiança, a maioria dos resultados desviam da média em no máximo 3%.

Novamente, para cada subconjunto de dados, a tabela é dividida em três blocos: método estado-da-arte, 2 novas heurísticas e 2 métodos de L2R, com destaque para os melhores resultados e empates estatísticos dentro de cada bloco e no geral, mostrados em quadros sombreados e letras em negrito, respectivamente.

Iniciamos a discussão a partir do método estado-da-arte. Particularmente, a eficácia do PITF no YouTube foi muito baixa provavelmente devido à natureza não colaborativa desta aplicação. Isso ocorre porque a principal desvantagem do PITF é que, conforme observado em nossos experimentos, ele não é robusto a um conjunto esparso de relacionamentos entre usuários, objetos e tags, ou seja, um conjunto de dados em que a grande maioria dessas entidades aparece raras vezes. PITF depende da disponibilidade de uma quantidade suficiente de exemplos de postagens de tags por um dado usuário u e objeto o no conjunto de treino para estimar corretamente os seus relacionamentos e prover recomendações relevantes para o par $\langle u, o \rangle$. Isso é particularmente crítico em aplicações não colaborativas, nas quais apenas um usuário pode associar tags a um objeto, como é o caso do YouTube. Por exemplo, se um objeto o de uma aplicação não-colaborativa aparece no conjunto de teste, ele estará ausente no conjunto de treino, visto que nenhum outro usuário associou tags a o. Logo, o relacionamento entre o e suas tags não pode ser inferido pelo PITF nesse caso. Como veremos a seguir, mesmo no LastFM, em que a atribuição de tags é colaborativa, os ganhos de nossas heurísticas sobre o PITF variam de 11% a 53% em P@5.

A seguir discutimos os resultados das heurísticas na Seção 6.6.1 e das estratégias baseadas em L2R na Seção 6.6.2.

6.6.1 Novas Heurísticas

Tabela 6.10. Resultados de P@5 média e intervalos de confiança de 95%: melhores resultados em cada bloco (método estado-da-arte, heurísticas e estratégias baseadas em L2R) em quadros sombreados. Melhores resultados gerais em negrito.

LastFM				
Estratégia	**Misto**	**Menor**	**Médio**	**Maior**
$PITF$	0.513 ± 0.001	.605 ± .008	.493 ± .004	.341 ± .002
$Sum^{+}wTS{+}UF$	0.574 ± .003	.669 ± .005	.574 ± .004	.490 ± .004
$LATRE{+}wTS{+}UF$	**0.597 ± 0.003**	**.700 ± .005**	.607 ± .003	.522 ± .003
PG	**0.597 ± 0.006**	**.701 ± .005**	**.616 ± .004**	**.530 ± .003**
$RankSVM$	0.579 ± 0.004	**.702 ± .005**	.593 ± .005	.503 ± .003
YouTube				
Estratégia	**Misto**	**Menor**	**Médio**	**Maior**
$PITF$	0.143 ± 0.002	.143 ± .005	.125 ± .002	.130 ± .002
$Sum^{+}wTS{+}UF$	0.525 ± 0.002	.615 ± .004	.480 ± .004	.460 ± .004
$LATRE{+}wTS{+}UF$	0.507 ± 0.002	.611 ± .004	.486 ± .004	.460 ± .007
PG	**0.529 ± 0.003**	.612 ± .003	.490 ± .003	.472 ± .003
$RankSVM$	**0.532 ± 0.004**	**.621 ± .003**	**.500 ± .005**	**.480 ± .004**

Tabela 6.11. Resultados de AP média e intervalos de confiança de 95%: melhores resultados em cada bloco (método estado-da-arte, heurísticas e estratégias baseadas em L2R) em quadros sombreados. Melhores resultados gerais em negrito.

LastFM				
Estratégia	**Misto**	**Menor**	**Médio**	**Maior**
$PITF$	0.386 ± 0.001	.415 ± .005	.364 ± .003	.312 ± .003
$Sum^{+}wTS{+}UF$	0.502 ± .003	.575 ± .004	.505 ± .003	.455 ± .004
$LATRE{+}wTS{+}UF$	**0.529 ± 0.003**	**.590 ± .005**	.541 ± .003	.493 ± .004
PG	**0.538 ± 0.005**	**.600 ± .004**	**.552 ± .003**	**.510 ± .005**
$RankSVM$	0.509 ± 0.003	**.591 ± .007**	.530 ± .004	.473 ± .003
YouTube				
Estratégia	**Misto**	**Menor**	**Médio**	**Maior**
$\|PITF$	0.110 ± 0.001	.097 ± .003	.103 ± .001	.108 ± .002
$Sum^{+}wTS{+}UF$	0.461 ± 0.002	.521 ± .005	.447 ± .005	.400 ± .003
$LATRE{+}wTS{+}UF$	0.449 ± 0.002	.509 ± .005	.458 ± .004	.413 ± .006
PG	**0.472 ± 0.002**	**.529 ± .002**	.460 ± .003	.417 ± .003
$RankSVM$	0.465 ± 0.003	**.529 ± .003**	**.468 ± .004**	**.426 ± .005**

Nós verificamos que nossas heurísticas produzem ganhos de até 331% em P@5, 264% em revocação e 436% em AP sobre o PITF. Assim, o uso de uma combinação de co-ocorrências de tags, múltiplos atributos textuais e métricas de relevância, incluindo a métrica relacionada ao histórico do usuário (UF), pode superar muito o desempenho

Tabela 6.12. Resultados de revocação média e intervalos de confiança de 95%: melhores resultados em cada bloco (método estado-da-arte, heurísticas e estratégias baseadas em L2R) em quadros sombreados. Melhores resultados gerais em negrito.

LastFM				
Estratégia	**Misto**	**Menor**	**Médio**	**Maior**
PITF	0.863 ± 0.002	.836 ± .004	.826 ± .004	.778 ± .002
$Sum^{+}wTS{+}UF$	0.885 ± .001	**.916 ± .004**	.881 ± .001	.832 ± .002
$LATRE{+}wTS{+}UF$	**0.901 ± 0.001**	**.916 ± .004**	**.892 ± .001**	**.859 ± .002**
PG	**0.901 ± 0.001**	**.916 ± .001**	**.892 ± .001**	**.859 ± .001**
RankSVM	**0.901 ± 0.001**	**.916 ± .004**	**.892 ± .001**	**.859 ± .002**

YouTube				
Estratégia	**Misto**	**Menor**	**Médio**	**Maior**
PITF	0.290 ± 0.003	.204 ± .006	.249 ± .004	.299 ± .005
$Sum^{+}wTS{+}UF$	0.728 ± 0.001	**.745 ± .003**	.680 ± .002	.655 ± .002
$LATRE{+}wTS{+}UF$	**0.749 ± 0.003**	**.745 ± .003**	**.691 ± .002**	**.684 ± .005**
PG	**0.749 ± 0.001**	**.745 ± .001**	**.691 ± .001**	**.684 ± .002**
RankSVM	**0.749 ± 0.003**	**.745 ± .003**	**.691 ± .002**	**.684 ± .005**

de métodos baseados apenas em interrelacionamentos entre usuários, objetos e tags, como é o caso do PITF.

Um fator importante que explica o sucesso de nossos métodos é que eles fornecem recomendações relevantes para um usuário mesmo que ele não apresente um histórico de atribuições de tags (*cold start problem* [Schein et al., 2002]). A extração de candidatos a partir de co-ocorrências de tags e múltiplos atributos textuais provê recomendações mais gerais para o objeto considerado, que podem ser relevantes para qualquer usuário que queira descrevê-lo. Se o usuário é mais ativo, nossos métodos podem fornecer um maior nível de personalização, se beneficiando mais da métrica UF.

Comparando as duas heurísticas propostas, $LATRE{+}wTS{+}UF$ supera $Sum^{+}wTS{+}UF$ em até 6% em P@5, 8% em AP e 3% em Recall. Em particular, as maiores melhorias ocorrem nas instâncias de teste com maior número de tags. Estes resultados são condizentes com os resultados dos métodos de recomendação centrada no objeto em que essas duas heurísticas são baseadas, que foram apresentados na Seção 6.5. No YouTube, os resultados das duas heurísticas estão empatados em geral (na única exceção, a diferença é menor que 3% em P@5). Isso ocorre porque, em geral, a diferença entre os resultados das heurísticas tende a diminuir à medida que novas fontes de candidatos são adicionadas. Por exemplo, múltiplos atributos textuais e histórico de usuário. Este fato é observado também nos experimentos com os métodos de recomendação centrada no objeto. Por exemplo, os ganhos do $LATRE$ sobre Sum^{+} são maiores do que os ganhos do $LATRE{+}DP$ sobre $Sum^{+}DP$.

6.6.2 Estratégias Baseadas em L2R

Considerando as estratégias baseadas em L2R, ambos métodos fornecem ganhos adicionais modestos sobre nossa melhor heurística, como observado para recomendação centrada no objeto. Por exemplo, os ganhos em P@5 e AP obtidos com a estratégia baseada em RankSVM sobre o $LATRE{+}wTS{+}UF$ alcançam 5% e 4%, respectivamente. Com a estratégia baseada em PG, os ganhos correspondentes chegam a 5%, por exemplo, no subconjunto misto de dados do YouTube. Nossas heurísticas e estratégias baseadas em L2R provêem o mesmo conjunto de candidatos e, consequentemente, seus resultados apresentam a mesma revocação.

Tais ganhos modestos ocorrem porque as estratégias de L2R, embora recebam uma lista contendo várias métricas, estão explorando majoritariamente as mesmas métricas utilizadas pelo $LATRE{+}wTS{+}UF$. De fato, nós verificamos que as métricas mais frequentemente utilizadas pela melhor função produzida ao final do processo evolucionário da PG são UF e $Sum(c, \mathcal{I}, 3)$, também usadas pelo $LATRE{+}wTS{+}UF$. $Sum(c, \mathcal{I}, 3)$ e as outras métricas seguem a mesma ordem de popularidade observada nos métodos de recomendação centrada no objeto correspondentes, como foi visto na Seção 6.5.2.

Entre as duas estratégias baseadas em L2R, o melhor método depende, mais uma vez, do conjunto de dados. Como ocorreu nos experimentos de recomendação centrada no objeto, a solução baseada em PG foi melhor no LastFM, com ganhos de até 6% tanto em P@5 como em AP. Já no YouTube, as duas estratégias estão estatisticamente empatadas em alguns casos, sendo o RankSVM ligeiramente melhor em outros, com ganhos de até 2% em P@5 e AP. Comparando as duas soluções em termos de tempo de treino em nossos conjuntos de dados, nós verificamos novamente que a técnica mais eficiente depende do conjunto de dados, como ocorreu com os métodos correspondentes de recomendação centrada no objeto. Por exemplo, a etapa de treino da PG requer, em média, 103 e 56 minutos considerando os dados do cenário misto do LastFM e YouTube respectivamente, quando executados em um processador Intel 2.83GHz Quad-Core com 8GB de RAM. Os números correspondentes para o RankSVM são 113 e 35 minutos. Também verificou-se que o tempo de treino do RankSVM exibe uma variabilidade muito maior sobre os diferentes *folds* do que a PG nos experimentos de recomendação personalizada.

Capítulo 7

Conclusões e Trabalhos Futuros

7.1 Conclusões

Nessa dissertação, nós propusemos e avaliamos várias estratégias de recomendação de tags centradas no objeto que exploram conjuntamente co-ocorrências de termos com tags previamente associadas ao conteúdo do objeto alvo, múltiplos atributos textuais e métricas de relevância, dimensões que não são exploradas por completo em trabalhos anteriores. Também propusemos estratégias de recomendação personalizadas, que exploram o histórico de atribuição de tags do usuário alvo, além de todas as dimensões utilizadas por nossas estratégias de recomendação centradas no objeto.

Nossas estratégias consistem em oito heurísticas e dois métodos baseados em *learning-to-rank* (L2R), que foram comparados com 4 técnicas estado-da-arte, sendo três delas centradas no objeto e uma personalizada, em várias bases de dados e cenários diferentes, compostos por diferentes quantidades de dados de treino disponíveis.

Quanto aos resultados dos métodos de recomendação centrada no objeto, verificamos que nossa melhor heurística, $LATRE{+}wTS$, fornece ganhos de até 40% em termos de precisão, 32% em revocação e 62% em AP, em relação à melhor linha de base em qualquer cenário analisado. Melhorias adicionais modestas, de até 12% em precisão, sobre nossas heurísticas, podem ser produzidas pelas estratégias baseadas em PG e RankSVM, embora a melhor dentre essas duas estratégias dependa da base de dados.

Quanto aos resultados dos métodos de recomendação personalizada, verificamos que nossa melhor heurística, $LATRE{+}wTS{+}UF$, fornece ganhos de até 328 % em termos de precisão, 264% em revocação e 424% em AP, em relação ao método estado-da-arte analisado. Uma melhoria adicional modesta, de até 5% em precisão, sobre nossas heurísticas, pode ser feita pelos métodos baseados em L2R. Concluímos também

que nossos resultados são mais robustos a dados esparsos do que métodos anteriores. Por exemplo, quando o usuário alvo tem pouca ou nenhuma participação no histórico de atribuições de tags de uma aplicação, nossos métodos são capazes de recomendar tags mais gerais relacionadas ao conteúdo e que também podem ser relevantes para tal usuário. Por fim, verificamos que o acréscimo da métrica relacionada ao histórico de usuário, UF, produz melhorias significativas em nossas estratégias de personalização. Por exemplo, os resultados de nossas heurísticas, quando um peso adequadamente ajustado para a métrica UF é utilizado, apresentam um ganho de até 12% em precisão em relação aos resultados das mesmas heurísticas com peso nulo para esta métrica.

Esses resultados ilustram os benefícios de se explorar as três dimensões discutidas anteriormente, particularmente as métricas de poder descritivo e a métrica relacionada ao histórico de usuário. Além disso, vimos que o uso de técnicas de L2R é uma solução promissora, visto que elas produzem resultados competitivos, além de serem flexíveis. Isto porque elas permitem a inclusão de novas métricas de relevância, abrindo espaço para possíveis melhorias futuras e podendo estender o escopo do problema, por exemplo, aplicando-as em outros tipos de recomendação ou mesmo em outros problemas de ordenação de itens por relevância.

7.2 Trabalhos Futuros

Recomendação de tags é um problema amplo que abre espaço para diversos trabalhos futuros, dentre eles, experimentos com usuários para avaliação manual das estratégias, um estudo mais aprofundado das estratégias aqui propostas, exploração de outras métricas e técnicas, particularmente junto às técnicas de L2R e aplicação de tais técnicas em outros problemas de recomendação e de *ranking*.

Houve apenas experimentos preliminares de avaliação manual com alguns alunos do curso de pós graduação do Departamento de Ciência da Computação da UFMG cujos resultados médios concordaram com os resultados da avaliação automática. Porém, verificamos que os julgamentos de relevância variaram muito de pessoa a pessoa, de forma que não obtivemos resultados conclusivos, logo não reportamos tais resultados neste trabalho. Assim, uma direção para trabalhos futuros seria ampliar o número de avaliações e/ou de avaliadores para tentar reduzir esses desvios. Outros aspectos como diversidade e novidade das recomendações também poderão ser avaliados.

O foco deste trabalho foi na situação em que há algumas tags disponíveis no objeto alvo, e queremos recomendar novas tags a ele, diferentes das já existentes. Outra tarefa interessante seria recomendar tags a um objeto ainda sem nenhuma. Embora

nossos métodos possam recomendar tags a um objeto que ainda não contenha tags, através do uso de outros atributos textuais, podemos futuramente dar um maior foco a estratégias que tratam desta situação. Podemos, por exemplo, recomendar a um objeto o tags existentes em objetos com conteúdo textual similar ao de o.

Como aprofundamento dos estudos aqui realizados, é interessante também investigar outras métricas. Por exemplo, no aspecto de co-ocorrência, houve foco nas métricas confiança e suporte. Há outras métricas relacionadas a regras de associação, como o *lift*, cujos benefícios para recomendação podem ser futuramente analisados. No contexto de personalização, o *UserRank* de usuários similares ao usuário alvo é um aspecto interessante a ser investigado como alternativa ou complemento à métrica de relevância ao usuário aqui estudada. Por fim, características extraídas do próprio conteúdo multimídia (vídeo, imagem, áudio) do objeto alvo também podem ser utilizadas como métricas complementares. Por exemplo, podemos utilizar termos provenientes de vídeos com conteúdo visual similar ao vídeo alvo como candidatos à recomendação. Tais métricas adicionais podem ser exploradas junto às estratégias baseadas em L2R aqui propostas, que são bastante flexíveis, abertas a extensões futuras.

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.

Almeida, H.; Gonçalves, M.; Cristo, M. & Calado, P. (2007). A combined component approach for finding collection-adapted ranking functions based on genetic programming. Em *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR, pp. 399--406, Amsterdam, The Netherlands. ACM.

Almeida, J.; Gonçalves, M.; Figueiredo, F.; Belém, F. & Pinto, H. (2010). On the quality of information for web 2.0 services. *IEEE Internet Computing*, 14(6):47--55.

Baeza-Yates, R. & Ribeiro-Neto, B. (1999). *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.; Almeida, J.; Gonçalves, M. & Pappa, G. (2010a). Exploiting co-occurrence and information quality metrics to recommend tags in web 2.0 applications. Em *Proceedings of the 19th ACM Conference on Information and Knowledge Management*, CIKM, pp. 1793--1796, Toronto, Canada.

Belém, F.; Martins, E.; Almeida, J.; Gonçalves, M. & Pappa, G. (2010b). Recomendação de tags a partir de métricas de qualidade de atributos textuais em aplicações da web 2.0. Em *Webmedia*.

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.

Bian, J.; Liu, Y.; Zhou, D.; Agichtein, E. & Zha, H. (2009). Learning to recognize reliable users and content in social media with coupled mutual reinforcement. Em *Proceedings of the 18th International Conference on World Wide Web*, WWW, pp. 51--60, New York, NY, USA. ACM.

Boll, S. (2007). MultiTube–Where Web 2.0 And Multimedia Could Meet. *IEEE MultiMedia*, 14(1).

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.

Chirita, P.; Costache, S.; Nejdl, W. & Handschuh, S. (2007). P-tag: Large scale automatic generation of personalized annotation tags for the web. Em *Proceedings of the 16th International Conference on World Wide Web*, WWW, pp. 845--854, Banff, Alberta, Canada. ACM.

Clements, M.; De Vries, A. & Reinders, M. (2010). The task-dependent effect of tags and ratings on social media access. *ACM Trans. Inf. Syst.*, 28:21:1--21:42.

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.

Figueiredo, F.; Belém, F.; Pinto, H.; Almeida, J.; Gonçalves, M.; Fernandes, D.; Moura, E. & Cristo, M. (2009a). Evidence of quality of textual features on the web 2.0. Em *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, CIKM, pp. 909--918, Hong Kong, China. ACM.

Figueiredo, F.; Belém, F.; Pinto, H.; Almeida, J.; Gonçalves, M.; Fernandes, D.; Moura, E. & Cristo, M. (2009b). Caracterizando o uso e a qualidade de atributos

textuais em aplicações da web 2.0. Em *Proceedings of the XV Brazilian Symposium on Multimedia and the Web*, WebMedia, pp. 9:1--9:8, Fortaleza, Ceará, Brazil. ACM.

Freund, Y.; Iyer, R.; 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.

Goodman, L. A. (1961). Snowball Sampling. *Annals of Mathematics and Statistics*, 32(1):148--170.

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.

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.

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.

Joachims, T. (2006). Training linear svms in linear time. Em *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD, pp. 217--226, Philadelphia, PA, USA. ACM.

Katakis, I.; Tsoumakas, G. & Vlahavas, I. (2008). Multilabel text classification for automated tag suggestion. Em *Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*, ECML PKDD, Antwerp, Belgium.

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.

Koutrika, G.; Effendi, F. A.; Gyöngyi, Z.; Heymann, P. & Garcia-Molina, H. (2007). Combating spam in tagging systems. Em *AIRWeb*, pp. 57--64.

Koutrika, G.; Effendi, F. A.; Gyöngyi, Z.; Heymann, P. & Garcia-Molina, H. (2008). Combating spam in tagging systems: An evaluation. *ACM Trans. Web*, 2(4).

Krestel, R.; Fankhauser, P. & Nejdl, W. (2009). Latent dirichlet allocation for tag recommendation. Em *Proceedings of the 3rd ACM Conference on Recommender systems*, RecSys, pp. 61--68, New York, New York, USA. ACM.

Li, X.; Guo, L. & Zhao, Y. E. (2008). Tag-based Social Interest Discovery. Em *Proceedings of the 17th International Conference on World Wide Web*, WWW, pp. 675--684, Beijing, China. 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.

Lipczak, M. & Milios, E. (2010). The impact of resource title on tags in collaborative tagging systems. Em *Proceedings of the 21st ACM Conference on Hypertext and hypermedia*, HT, pp. 179--188, Toronto, Ontario, Canada. ACM.

Liu, T. Y.; Xu, J.; Qin, T.; Xiong, W. & Li, H. (2007). Letor: Benchmark dataset for research on learning to rank for information retrieval. Em *Proceedings of the Learning to Rank Workshop in the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*.

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 jont Conference on Artifical intelligence*, pp. 2064--2069, Pasadena, California, USA. Morgan Kaufmann Publishers Inc.

Ma, H.; King, I. & Lyu, M. (2009). Learning to recommend with social trust ensemble. Em *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR, pp. 203--210, Boston, MA, USA. 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.

Menezes, G. (2011). Recomendação de tags por demanda. Em *Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Minas Gerais. Orientador: Prof. Nivio Ziviani.*

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.

Rader, E. & Wash, R. (2008). Influences on tag choices in del.icio.us. Em *Proceedings of the 2008 ACM Conference on Computer supported cooperative work*, CSCW, pp. 239--248, San Diego, CA, USA. ACM.

Rae, A.; Sigurbjörnsson, B. & van Zwol, R. (2010). Improving tag recommendation using social networks. Em *Adaptivity, Personalization and Fusion of Heterogeneous Information*, RIAO, pp. 92--99, Paris, France.

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.

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.

Santos, R. L.; Macdonald, C. & Ounis, I. (2011). Intent-aware search result diversification. Em *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR, pp. 595--604, Beijing, China. ACM.

Sarwar, B.; Karypis, G.; Konstan, J. & Riedl, J. (2001). Item-based collaborative filtering recommendation algorithms. Em *Proceedings of the 10th International Conference on World Wide Web*, WWW, pp. 285--295, Hong Kong, Hong Kong. ACM.

Schein, A.; Popescul, A.; Ungar, L. & Pennock, D. (2002). Methods and metrics for cold-start recommendations. Em *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR, pp. 253--260, New York, USA. ACM.

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.

Sen, S.; Harper, F. M.; LaPitz, A. & Riedl, J. (2007). The quest for quality tags. Em *Proceedings of the 2007 International ACM Conference on Supporting group work*, GROUP, pp. 361--370, Sanibel Island, Florida, USA. 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.; Zhang, L. & Giles, C. L. (2011). Automatic tag recommendation algorithms for social recommender systems. *ACM Trans. Web*, 5:1--31.

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.

Torres, R.; Falcão, A.; Gonçalves, M.; Papa, J.; Zhang, B.; Fan, W. & Fox, E. A. (2009). A Genetic Programming Framework For Content-Based Image Retrieval. *Pattern Recognition*, 42(2):283--292.

van Damme, C.; Hepp, M. & Coenen, T. (2008). Quality metrics for tags of broad folksonomies. Em *Proceedings of the International Conference on Semantic Systems (I-Semantics 2008)*.

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, pp. 835--844, Hong Kong, China. 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.

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.; Fu, Y.; Mao, J. & Su, D. (2006). Towards The Semantic Web: Collaborative Tag Suggestions. Em *WWW2006: Proceedings of the Collaborative Web Tagging Workshop*.

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, B.; Gonçalves, M.; Fan, W.; Chen, Y.; Fox, E. A.; Calado, P. & Cristo, M. (2004). Combining Structural And Citation-Based Evidence For Text Classification. Em *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, CIKM. ACM.

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.