UM ARCABOUÇO PARA BUSCA E RECOMENDAÇÃO DE ARTIGOS CIENTÍFICOS

CRISTIANO NASCIMENTO

UM ARCABOUÇO PARA BUSCA E RECOMENDAÇÃO DE ARTIGOS CIENTÍFICOS

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

Orientador: Alberto Henrique Frade Laender
Coorientador: Altigran Soares da Silva

Belo Horizonte

Março de 2011

Nascimento, Cristiano
N244a Um arcabouço para busca e recomendação de artigos científicos / Cristiano Nascimento. — Belo Horizonte, 2011
xx, 74 f. : il. ; 29cm

Dissertação (mestrado) — Universidade Federal de Minas Gerais. Departamento de Ciência da Computação.
Orientador: Alberto Henrique Frade Laender
Coorientador: Altigran Soares da Silva

1. Computação - Teses. 2. Recuperação da Informação - Teses. I. Orientador. II. 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

Um arcabouço para busca e recomendação de artigos científicos

CRISTIANO ALEX OLIVEIRA DO NASCIMENTO

Dissertação defendida e aprovada pela banca examinadora constituída pelos Senhores:

PROF. ALBERTO HENRIQUE FRADE LAENDER - Orientador
Departamento de Ciência da Computação - UFMG

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

PROF. NIVIO ZIVIANI
Departamento de Ciência da Computação - UFMG

Belo Horizonte, 17 de março de 2011.

À Força do Invisível que ilumina o meu caminho.

"In essence, if we want to direct our lives, we must take control of our consistent actions. It's not what we do once in a while that shapes our lives, but what we do consistently."
(Anthony Robbins)

Resumo

A pesquisa bibliográfica é uma das tarefas mais importantes e também uma das mais trabalhosas dentro da pesquisa científica e está presente na vida de muitos pesquisadores em todo o mundo. Nas últimas décadas, essa tarefa passou a ser realizada principalmente na Web, onde milhares de novos artigos científicos são publicados e disponibilizados todos os anos. Devido à grande quantidade de artigos existentes, fazer uma análise completa dos artigos relacionados a um trabalho tornou-se impraticável. Para auxiliar os pesquisadores nessa árdua tarefa, sistemas de recomendação têm sido utilizados para filtrar e sugerir conteúdos de interesse. O problema das atuais abordagens de recomendação de artigos científicos é que elas utilizam informações privilegiadas, como redes de citações e coleções de artigos armazenadas previamente, o que impede que serviços de recomendação sejam oferecidos por outros sistemas que não tenham acesso a esse tipo de informação. Para resolver este problema, neste trabalho propomos um arcabouço para busca e recomendação de artigos científicos que independe de tais informações privilegiadas. O arcabouço proposto utiliza formulários de consulta disponíveis na Web para gerar um conjunto de artigos candidatos e recomenda os artigos mais relacionados considerando somente informações disponíveis publicamente. Para isso, propomos e avaliamos estratégias de geração e seleção de consultas a partir de informações contidas em um único artigo fornecido como entrada e avaliamos a utilização de diferentes campos do artigo científico para geração das consultas. Além disso, avaliamos a utilização de diferentes fontes de informação para encontrar os artigos e a eficácia de estratégias de ordenação baseadas em conteúdo na recomendação de artigos científicos.

Abstract

Bibliographical search is an important and a difficult task in the scientific research and it is commonly accomplished by many people around the world. There is currently an increasing number of research papers available on the Web, thus a significant portion of this task is currently performed in the digital environment. Due to this large amount of information, it has become almost impossible to cover all the relevant related work present in the literature. In order to aid researchers to find related papers, recommender systems have been applied to filter and suggest content of interest. The main problem with the current approaches is the use of privileged information, such as citation networks and previously stored full paper collections, which do not allow systems without that information to offer recommendation services. In order to solve this problem, in this work we propose a framework for finding and recommending research papers that does not use any privileged information. The framework discovers related work through search interfaces available on the Web and recommends the most relevant papers by using only publicly available information. Then, we show how to create and select queries by using information from a single paper, and evaluate for this task the use of different fields from the research paper. We also evaluate the use of different information sources and content-based strategies for paper recommendation.

Lista de Figuras

Lista de Tabelas

Sumário

Capítulo 1

Introdução

1.1 Contexto

A Web é atualmente uma das maiores, se não a maior, fonte de informação disponível no mundo. Ao longo dos anos, o conteúdo inserido na grande teia tem crescido muito mais do que a nossa capacidade de absorção. Se, por um lado, o crescimento desse conteúdo proporciona diversidade e completude de informação, por outro lado, também gera uma imensa quantidade de conteúdo de baixa qualidade, o que dificulta a recuperação de informação relevante.

O procedimento mais comum para se encontrar informação na Web é por meio do preenchimento de formulários de consulta, sejam eles de propósito geral, como nas máquinas de busca, ou específicos, como nas lojas virtuais e nas bibliotecas digitais. Nesse tipo de procedimento, um usuário especifica uma consulta inserindo informação textual em uma caixa de texto e analisa uma lista de documentos retornados, como páginas HTML, imagens, vídeos, etc, em busca daqueles que são mais importantes para o tema pesquisado. Esse tipo de procedimento tem a vantagem de ser simples e rápido, porém a formulação das consultas nem sempre é trivial, sendo necessário em muitos casos submeter várias consultas até que se consiga recuperar conteúdo relevante.

Em alguns contextos específicos, o procedimento de se formular consultas e analisar resultados pode não ser o mais apropriado para a descoberta de novos conteúdos. Em uma loja virtual, por exemplo, um usuário nem sempre faz uma visita procurando por um produto específico, podendo simplesmente navegar pelo *site* para saber as novidades, sem mesmo ter conhecimento da existência de alguns produtos que podem ser do seu interesse. Dessa forma, ele acaba não formulando nenhuma consulta e não toma conhecimento dos produtos mais adequados ao seu perfil. Em uma outra situação, um usuário poderia ter ideia do que deseja encontrar mas pode ser muito difícil formular

uma consulta que encontre os itens relevantes. Por exemplo, um determinado usuário gosta de certas bandas de *rock* e gostaria de encontrar álbuns de outras bandas que tenham estilos parecidos em um *site* especializado em música. Esse usuário poderia então formular uma consulta com a palavra 'rock' e obter uma lista com todas as bandas de *rock* disponíveis no *site*. Além de ter de analisar uma extensa lista de resultados, o usuário pode se deparar com a situação em que bandas de estilos semelhantes tenham sido rotuladas com palavras diferentes e não encontrar o que deseja. Esses são apenas alguns exemplos de casos onde a busca por meio de formulários pode não ser a melhor escolha. É nesse contexto que se aplicam os chamados sistemas de recomendação.

Um sistema de recomendação é uma ferramenta que tenta prever ou predizer um item ou um conjunto de itens que seja do interesse de um usuário [Adomavicius & Tuzhilin, 2005]. Isso pode ser feito, por exemplo, baseando-se no comportamento prévio do próprio usuário, no comportamento de outros usuários, no conteúdo de páginas ou documentos, entre outros. Esse tipo de tecnologia tem sido bem utilizado nos últimos anos em diversos cenários tais como lojas virtuais (livros, CDs), redes sociais (amigos, comunidades, músicas, bandas), bibliotecas digitais (artigos e documentos digitais), entre outros. A Figura 1.1 mostra um exemplo de sistema de recomendação de livros de uma loja virtual, a Amazon.com. Na parte superior da figura encontram-se algumas informações sobre um livro e na parte inferior são mostrados os livros com os quais ele tipicamente coocorre em um compra. A intuição é que, se o usuário está interessado no livro exibido, então ele também pode estar interessado nos demais, pois outras pessoas com o mesmo interesse compraram os livros que estão sendo sugeridos.

Neste trabalho, propomos um arcabouço para busca e recomendação de artigos científicos a fim de facilitar o trabalho de inúmeras pessoas que utilizam a Web para fazer pesquisa bibliográfica. Baseado no conteúdo de um documento fornecido por um usuário, o arcabouço recomenda uma lista de documentos relacionados que possam ser do interesse desse usuário. Para isso, o arcabouço gera consultas automaticamente, faz buscas em serviços disponíveis na Web e seleciona os documentos mais relacionados ao documento fornecido pelo usuário.

1.2 Motivação

A pesquisa bibliográfica envolve uma série de tarefas, entre elas pesquisar, recuperar, filtrar e analisar uma grande quantidade de documentos para se adquirir conhecimento, fundamentar um estudo, manter-se atualizado sobre o estado da arte em uma determinada área do conhecimento, entre outras razões. Trata-se de uma tarefa muito

Figura 1.1. Exemplo de um sistema de recomendação de livros: Amazon

importante que requer muito trabalho e, portanto, necessita de ferramentas que possam auxiliar na sua execução.

Uma estratégia muito comum utilizada pelos pesquisadores para realizar essa tarefa é seguir a lista de referências presente em um artigo que já tenham em mãos. Apesar de ser bastante efetiva em alguns casos, essa estratégia não garante uma cobertura completa sobre as principais referências da área. Por exemplo, algumas referências importantes podem não ter sido citadas pelo fato de tratarem de temas ligeiramente diferentes do tema principal. Em outros casos, os autores têm de lidar com o problema de espaço limitado para descrever seus trabalhos, que normalmente possuem um número fixo de páginas que seguem um certo modelo de documento. Para ter mais espaço para desenvolver suas ideias, algumas referências são simplesmente removidas do artigo. Um outro problema dessa estratégia é que somente os artigos publicados anteriormente ao artigo corrente podem ser referenciados. Por fim, mesmo que um pesquisador encontre um bom conjunto de referências, ele ainda precisa encontrar o

texto completo dos artigos, que nem sempre está disponível publicamente.

Atualmente, há uma grande quantidade de artigos científicos disponíveis na Web e essa quantidade aumenta ano após ano, fazendo com que uma parte significativa da tarefa de pesquisa bibliográfica seja realizada no ambiente virtual. Existem inclusive mecanismos de busca específicos para a literatura científica como o CiteSeer[1] [Giles et al., 1998] e o Google Scholar[2]. As editoras e as sociedades científicas também costumam fornecer serviços de busca para suas próprias publicações, como é o caso da Springer, ACM[3] e IEEE[4]. Nesses ambientes, os pesquisadores também formulam consultas compostas de palavras-chave para encontrar documentos de interesse, como na busca na Web em geral. Após as consultas serem realizadas, uma lista de resultados é retornada e, dessa lista, os pesquisadores identificam os documentos que atendem aos seus interesses.

Na abordagem descrita anteriormente, a responsabilidade de traduzir os interesses e objetivos da pesquisa na melhor consulta possível, com os termos mais adequados para recuperar documentos relevantes, é do próprio pesquisador. Mesmo desconsiderando o número de possíveis fontes diferentes e a sintaxe usada para formulação das consultas, essa não é uma tarefa trivial. Em muitos casos, é preciso inclusive formular diversas consultas para se recuperar os documentos relevantes. Por último, mesmo nos serviços de busca oferecidos pelas editoras e sociedades científicas, em que o conjunto de documentos a ser pesquisado é menor, a lista de resultados retornada pode conter centenas ou milhares de documentos, o que a torna impraticável de ser analisada manualmente.

Para diminuir o esforço dos pesquisadores na tarefa de pesquisa bibliográfica, vários trabalhos de recomendação de artigos científicos têm sido propostos na literatura [McNee et al., 2002; Strohman et al., 2007; Bogers & van den Bosch, 2008; Gori & Pucci, 2006]. No entanto, os trabalhos atuais apresentam algumas limitações práticas que os tornam inviáveis de serem implementados e oferecidos aos pesquisadores. Abordagens como a de McNee et al. [2002], por exemplo, assumem que o sistema de recomendação já possui uma coleção de artigos armazenada previamente para fazer a recomendação. Na prática, muitos artigos não estão disponíveis publicamente e somente algumas poucas editoras possuem coleções completas. Além disso, os sistemas de recomendação propostos utilizam apenas documentos de uma única fonte de informação, enquanto que, em uma situação real, os usuários normalmente encontram artigos relacionados espalhados em diversas fontes disponíveis na Web. Uma outra

[1] http://citeseer.ist.psu.edu/
[2] http://scholar.google.com/
[3] http://portal.acm.org/
[4] http://ieeexplore.ieee.org/

questão é que algumas abordagens, como a de Pohl et al. [2007], consideram que o sistema de recomendação possui informações históricas dos usuários, o que pode ser um problema para novos sistemas que não têm esse tipo de informação.

Algumas aplicações que possuem acesso a uma coleção de artigos já oferecem serviços de recomendação, como o Google Scholar e a ScienceDirect[5]. A Figura 1.2 mostra o serviço de recomendação da ScienceDirect em que uma lista de artigos relacionados é associada a um artigo que está sendo lido na página de um navegador Web. Do ponto de vista do usuário, talvez o maior problema dos serviços de recomendação oferecidos atualmente, além de não oferecer recomendações de fontes diferentes (embora o Google Scholar o faça), seja não oferecer opções de configuração de recomendação. Na Figura 1.2, por exemplo, a mesma recomendação é oferecida para todos os usuários. Como cada usuário (pesquisador, professor ou estudante) pode ter um objetivo diferente na busca por trabalhos relacionados, acreditamos que um serviço de recomendação deveria permitir que o usuário adaptasse as configurações de acordo com seu objetivo ou perfil.

Neste trabalho, consideramos um cenário que seja um pouco mais próximo daquele que ocorre no mundo real. Um usuário após algumas consultas, navegações ou até mesmo através de indicações de outras pessoas, obtém um documento relevante para a sua pesquisa. Então, para criar a sua base de referências bibliográficas, ele decide procurar documentos relacionados ao documento obtido anteriormente. Para isso, o usuário provavelmente irá formular novas consultas utilizando algumas palavras encontradas no corpo do documento, repetindo as consultas em diversos sistemas de busca. Por fim, o usuário analisa diversos documentos nas listas de resultados e escolhe aqueles que julga serem os mais interessantes. Como podemos notar, essa é uma tarefa que requer muito tempo e esforço por parte dos usuários e que demanda mecanismos mais práticos para se encontrar documentos relacionados.

Considerando o cenário descrito acima e as limitações das abordagens de recomendação de artigos científicos existentes, propomos nesta dissertação um arcabouço concebido especialmente para busca e recomendação de artigos científicos. O arcabouço recebe como entrada um documento de referência e retorna um conjunto de documentos relacionados a esse documento. Para isso, ele consulta várias fontes disponíveis na Web para construir uma base diversificada de documentos e utiliza apenas o conteúdo do documento de referência para fazer a recomendação. Além disso, todo o processo é automático, o que reduz a quantidade de esforço por parte dos usuários.

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

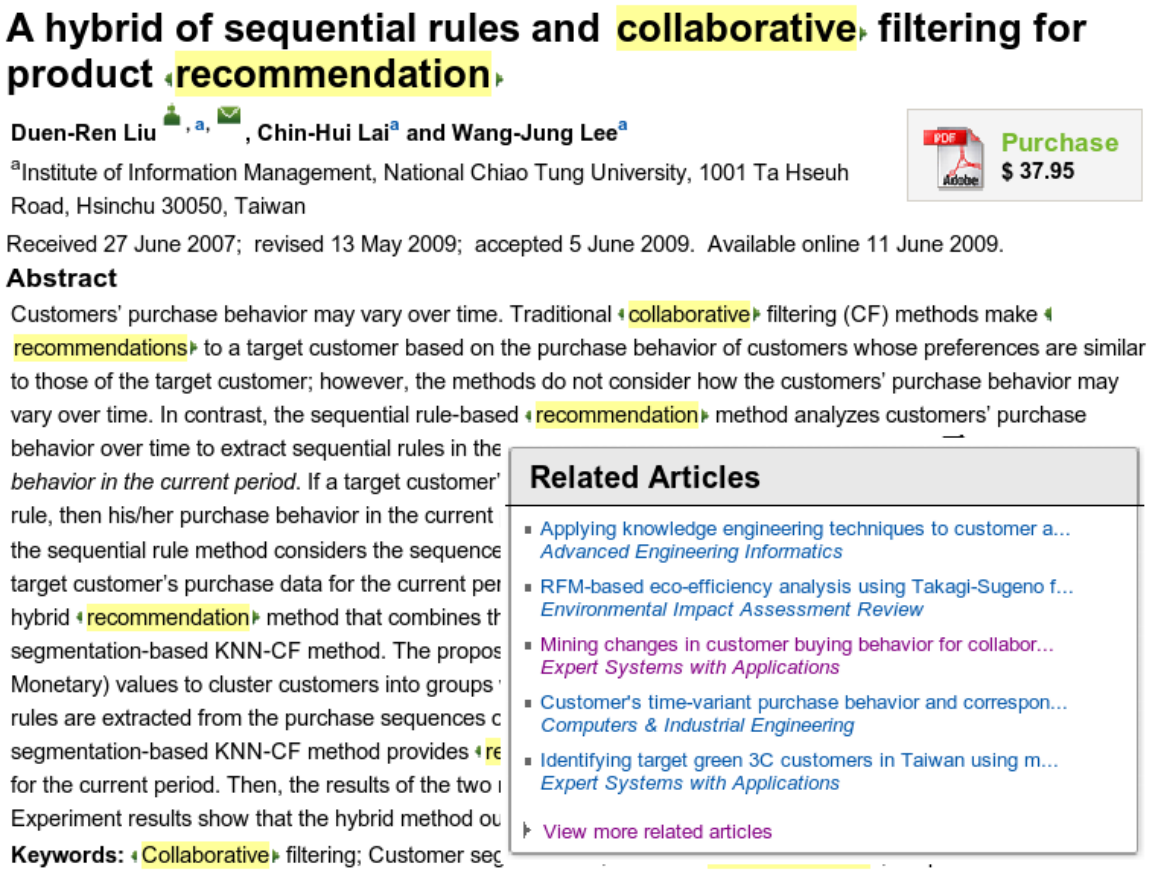


Figura 1.2. Exemplo de um serviço de recomendação de artigos científicos oferecido pela ScienceDirect

1.3 Trabalhos Relacionados

Nesta seção, apresentamos os trabalhos relacionados ao arcabouço de recomendação de artigos científicos proposto. Como o arcabouço é composto por vários módulos, existem diversos trabalhos relacionados que poderiam ser apresentados de acordo com a função específica de cada um desses módulos. Porém, nesta seção, iremos focar principalmente nos trabalhos que lidam diretamente com o problema de recomendação de artigos científicos.

É importante ressaltar que se encontram descritos na literatura vários casos particulares do problema de recomendação de artigos científicos, como é o caso do Problema de Atribuição de Artigos a Revisores [Wang et al., 2008], mas que estão fora do escopo deste trabalho e não serão aqui abordados.

Assim, organizamos a discussão dos trabalhos relacionados de acordo com a classificação do sistema de recomendação utilizado. Sistemas de recomendação geralmente são classificados de acordo com a forma que as recomendações são feitas [Adomavicius

& Tuzhilin, 2005], sendo as mais comuns as recomendações baseadas em conteúdo, as recomendações colaborativas e as recomendações híbridas. A seguir, apresentamos os principais trabalhos de recomendação de artigos científicos considerando as três categorias descritas anteriormente.

1.3.1 Recomendação Baseada em Conteúdo

Nos sistemas de recomendação baseados em conteúdo, as sugestões de itens para um usuário são feitas comparando-se o conteúdo de um item com o de outros que o usuário já tenha manifestado interesse anteriormente. Nesse tipo de abordagem normalmente cria-se um perfil para um usuário utilizando-se informações dos itens associados a ele e informações pessoais como idade, sexo e localização, caso estejam disponíveis. Depois, compara-se o perfil criado com todos os itens presentes no conjunto de possíveis recomendações e recomenda-se os itens com similaridade acima de um certo nível ou simplesmente recomenda-se os n itens mais similares.

Dada a importância das aplicações baseadas em conteúdo textual, os principais sistemas baseados em conteúdo focam nesse tipo de informação para construir o perfil do usuário e também a representação dos itens. Mais especificamente, tais sistemas utilizam as palavras ou termos (i.e., conjunto de palavras) que descrevem o conteúdo dos itens para criar as representações.

Ao criar-se as representações dos itens e o perfil do usuário é importante notar que nem todos os termos descrevem os itens da mesma forma, sendo uns mais importantes e com maior capacidade de descrição que outros. Por isso, é bastante comum a utilização de esquemas de atribuição de pesos para diferenciar os termos que melhor descrevem o conteúdo de um item. Dessa forma, é possível calcular a similaridade entre o perfil e o item de maneira mais precisa atribuindo-se maior importância ao casamento dos termos mais relevantes.

Um dos esquemas mais tradicionais utilizados na atribuição de pesos é o *term frequency/inverse document frequency* (TF-IDF) [Baeza-Yates & Ribeiro-Neto, 2011]. A frequência de um termo (TF) é o número de ocorrências desse termo em um determinado documento. Já a frequência inversa nos documentos (IDF) é a razão inversa do número de documentos que contêm o termo e o número total de documentos em uma coleção. No esquema TF-IDF, o peso de um determinado termo é calculado multiplicando-se o seu TF pelo o IDF. A intuição é que se um termo aparece várias vezes em um documento então ele provavelmente indica uma ideia central no documento, mas se ele aparecer em muitos documentos da coleção então ele não é um bom discriminante do conteúdo do documento.

McNee et al. [2006] analisam a aplicação de várias estratégias de recomendação em diferentes cenários. Uma das estratégias utilizadas foi a similaridade entre o texto completo dos artigos baseada em TF-IDF. Os serviços oferecidos pelo Google Scholar também são baseados no acesso ao texto completo do artigo, embora outras informações também sejam consideradas. Apesar de o texto completo dos artigos fornecer boa informação contextual, o uso do mesmo pode ser muito restrito, pois ele nem sempre está disponível publicamente.

Chandrasekaran et al. [2008] criam perfis para autores a partir de artigos produzidos pelos mesmos que estão armazenados no CiteSeer[6]. Para cada documento na lista de um autor, é gerada uma lista de conceitos que definem os tópicos do artigo. Depois, cria-se o perfil para o autor utilizando-se as listas de conceitos de seus artigos publicados. Para recomendar os artigos, os autores modelam os conceitos do perfil em uma árvore e utilizam distância de edição em árvores para medir a similaridade entre o perfil e os artigos candidatos [Lakkaraju et al., 2008]. Os experimentos mostraram que o uso de conceitos ao invés de palavras-chave na representação de perfis e artigos obtém melhores recomendações.

Pudhiyaveetil et al. [2009] apresentam um trabalho bastante similar, porém constroem os perfis de usuários de maneira diferente. Nesse trabalho, eles utilizam os artigos que o usuário acessou na biblioteca digital como sendo seus artigos de interesse ao invés de utilizar os artigos escritos por um autor. A recomendação é feita utilizando o peso do conceito no perfil do usuário e o peso do conceito no artigo candidato. Apesar de ser interessante, esse trabalho e o outro apresentado no parágrafo anterior limitam o serviço de recomendação à criação de taxonomias bem definidas e também ao treinamento de um classificador para gerar as listas de conceitos.

Sugiyama & Kan [2010] também criam um perfil para um determinado autor considerando não só seus artigos produzidos anteriormente mas também os artigos que ele cita e os artigos que citam seus trabalhos. Para cada artigo publicado por um autor cria-se uma representação que é uma combinação dos termos presentes no próprio artigo, nos artigos que são citados por ele e nos artigos em que ele é citado. Depois, combina-se a representação de cada artigo para gerar o perfil do autor. O mesmo processo é realizado na geração da representação dos artigos candidatos. O peso dos termos na representação dos autores é calculado usando-se somente o TF normalizado por causa do número reduzido de artigos e na representação dos candidatos utiliza-se o esquema TF-IDF tradicional. Para se fazer a recomendação, utiliza-se a similaridade do cosseno [Baeza-Yates & Ribeiro-Neto, 2011]. Os resultados mostram que o uso da

[6] http://citeseer.ist.psu.edu/

informação contextual melhora a precisão das recomendações.

Um dos problemas das abordagens baseadas em conteúdo é que elas não consideram ambiguidades presentes nas expressões em linguagem natural como sinonímia e polissemia [Pudhiyaveetil et al., 2009]. Dessa forma, elas não são capazes de encontrar itens relacionados se eles não compartilham um vocabulário comum. Um outro problema é que as recomendações podem se tornar muito específicas, isto é, os itens recomendados provavelmente serão muito similares ao perfil do usuário o que pode ser ruim pois as recomendações também deveriam trazer novidades.

Para resolver o problema de recomendações muito parecidas, Zhang et al. [2002] estendem a abordagem baseada em conteúdo levando em consideração o grau de novidade do artigo, isto é, o quanto ele difere daqueles que já foram vistos anteriormente. Para isso, utilizam teoria dos conjuntos, medida do cosseno e a medida Kullback-Leibler [Kullback & Leibler, 1951].

1.3.2 Recomendação Colaborativa

Nos sistemas de recomendação colaborativos, as sugestões são feitas considerando-se as preferências de outros usuários com perfis similares. Os perfis normalmente são construídos sem levar em consideração qualquer tipo de informação sobre o conteúdo dos itens. Ao invés disso, utiliza-se as notas que os usuários atribuem aos itens. No caso de não haver notas atribuídas explicitamente pelos usuários, o sistema de recomendação pode derivá-las de suas interações com os itens. Por exemplo, pode-se atribuir nota 1 quando um usuário compra um item e 0 caso contrário.

Depois que os perfis são criados, o sistema calcula a similaridade entre o perfil do usuário alvo e os demais usuários. Duas medidas de similaridade são comumente utilizadas para identificar usuários similares, a medida do cosseno e o coeficiente de correlação de Pearson [Resnick et al., 1994]. Através do conjunto de usuários mais similares, geram-se notas para os itens que ainda não foram associados ao usuário tentando-se predizer sua preferência. Por fim, os itens de maior nota são recomendados.

Na recomendação de artigos científicos, Bogers & van den Bosch [2008] utilizam as informações de usuários do CiteULike[7] para gerar as recomendações. O CiteULike é um serviço que permite que um usuário gerencie seus artigos registrando aqueles que são de seu interesse. Nesse ambiente, em que existem muito mais artigos do que usuários, verificou-se que os algoritmos colaborativos baseados em usuários funcionam melhor do que os baseados em itens e que ambos sofrem do problema de *cold-start* [Resnick

[7] http://www.citeulike.org/

et al., 1994]. No início, as aplicações dispõem de pouco ou nenhum dado (usuários e interações) e, portanto, a recomendação colaborativa fica bastante debilitada. Nesse caso, novos itens ainda levam algum tempo para fazer parte da lista dos usuários e, por isso, podem demorar para serem recomendados [Burke, 2007]. Já quanto aos novos usuários, esses ainda podem não ter um perfil bem estabelecido e assim não ser possível identificar quais seriam os itens de seu interesse.

Um outro trabalho que também lida com informações de usuários é apresentado por Agarwal et al. [2005]. Nesse trabalho, os autores assumem que pesquisadores da mesma área tendem a se interessar pelos mesmos artigos e utilizam essa premissa para fazer recomendações. Eles tentam melhorar o resultado de buscas por meio de recomendações baseadas em buscas feitas anteriormente pelos usuários mais similares.

Já Pohl et al. [2007], ao invés de utilizar buscas feitas por usuários, consideram os registros de *download* de artigos científicos em uma biblioteca digital. Nesse trabalho, os autores utilizam a frequência com que os artigos são baixados pelo mesmo usuário para inferir o grau de relacionamento entre eles.

Abordagens como as citadas anteriormente têm potencial para fazer recomendações de boa qualidade. No entanto, elas também possuem algumas deficiências que podem torná-las inviáveis de ser utilizadas. A primeira é o chamado *cold-start* mencionado anteriormente. Um segundo problema ocorre quando um usuário possui informação suficiente sobre os itens de seu interesse mas seu perfil não se encaixa com os perfis existentes. Neste caso, pode acontecer de se recomendar itens de pouco interesse para esses usuários. Por fim, alguns desses sistemas colaborativos ao invés de utilizar itens associados aos usuários, utilizam pontuações que os usuários atribuem aos itens. No entanto, nem sempre os usuários se dispõem a fazer essas atribuições e o sistema pode sofrer pela falta de dados.

Para solucionar o problema da falta de informação, alguns trabalhos utilizam as citações bibliográficas para derivar as interações implicitamente. As citações normalmente são modeladas como um grafo direcionado em que cada citação representa um nó e existe uma aresta direcionada a_{ij} da citação c_i para a citação c_j se c_i cita c_j. Na pesquisa científica, a citação bibliográfica pode ser vista como uma forte evidência de preferência e por isso é bastante utilizada nas recomendações.

Uma maneira de se explorar a rede de citações é analisar a ocorrência de cocitações [Small, 1973]. Nesse tipo de abordagem, observa-se a frequência com que dois artigos são citados conjuntamente em um mesmo trabalho, sendo que a alta ocorrência de cocitações indica que ambos provavelmente lidam com os mesmos assuntos em uma determinada área do conhecimento.

Em Strohman et al. [2007] as citações bibliográficas são utilizadas de duas

maneiras distintas. Primeiro, o conjunto de artigos candidatos a recomendação é expandido usando-se as referências presentes nos próprios artigos. Depois, medidas baseadas em rede são utilizadas como evidências em um modelo de recomendação. Nesse trabalho, os autores observaram que o uso de métricas baseadas na rede de citações melhora a qualidade dos resultados. Além disso, verificou-se que a medida Katz [Liben-Nowell & Kleinberg, 2003], que mede a distância entre nós em um grafo considerando o número de caminhos mínimos, tem maior impacto na recomendação de artigos do que a cocitação.

McNee et al. [2002] utilizam uma abordagem tradicional de recomendação colaborativa em que os usuários são mapeados como linhas e os itens como colunas em uma matriz de notas. No entanto, ao invés de utilizar usuários, os autores mapeiam os artigos como usuários nas linhas e as referências desses artigos como itens nas colunas. As referências que fazem parte de um artigo recebem nota 1 e as demais recebem 0. Gori & Pucci [2006] utilizam uma abordagem baseada em caminhamento aleatório sobre o grafo de citações com um algoritmo chamado *PaperRank* para gerar notas (ou pontos) para cada documento.

1.3.3 Recomendação Híbrida

Por fim, sistemas de recomendação híbridos utilizam tanto as informações de conteúdo de um item quanto as das interações de outros usuários para sugerir itens.

Torres et al. [2004], utilizam algoritmos baseados em conteúdo para encontrar artigos similares ao documento corrente, a partir da medida do cosseno, e então algoritmos colaborativos, que utilizam os k vizinhos mais próximos, para ordenar as citações recomendadas. Huang et al. [2002], constroem um grafo de duas camadas utilizando informações de usuários, livros e transações. Uma vez que o grafo tenha sido modelado, a recomendação se torna um problema de busca em grafos. Huang et al. [2004] utilizam a abordagem híbrida de maneira diferente. Enquanto em outros trabalhos as abordagens de conteúdo e colaborativa são utilizadas separadamente, nesse trabalho os autores as utilizam de forma que uma reforce a outra.

1.3.4 Outros Trabalhos Relacionados

A seguir, apresentamos outros trabalhos relacionados cuja abordagem não se enquadra na classificação utilizada anteriormente.

A ideia de utilizar documentos como entrada para interfaces de consulta não é nova. Yang et al. [2009], por exemplo, tratam do problema de referenciamento cruzado

(*cross referencing*) de conteúdo *online* utilizando essa abordagem. Dada uma notícia em um *site*, tenta-se encontrar *blogs* com conteúdo relacionado através de interfaces de consulta utilizando a notícia como fonte de geração de consultas. Adotamos a mesma estratégia de Yang et al. [2009] para gerar consultas candidatas, porém, divergimos nas estratégias de ordenação das consultas. Eles utilizam métricas baseadas em informações globais como IDF e informação mútua. Além disso, não é levado em consideração a estrutura do documento. No nosso caso, há uma diferença clara entre os segmentos de texto que compõem o artigo científico e consideramos isso ao criar as consultas.

Um segundo trabalho que trata da utilização de documentos como consultas é apresentado por Dasdan et al. [2009]. Nesse trabalho, os autores tratam do problema de *coverage testing* no qual tenta-se descobrir se um determinado conjunto de documentos contém uma quase duplicata (*near-duplicate*) de um documento dado como entrada. Nesse trabalho, subsequências de termos dentro do documento de entrada são escolhidas aleatoriamente como consultas. Consultas desse tipo aumentam as chances de se recuperar o documento de entrada por serem bastante específicas, mas não nos parece a melhor estratégia na recuperação de outros documentos relacionados. Ao invés disso, utilizamos consultas mais genéricas selecionadas de acordo com sua importância no documento.

Um outro tópico de pesquisa relacionado ao nosso trabalho é a busca de patentes. Patente é uma concessão pública que garante ao seu titular a exclusividade ao explorar comercialmente sua criação[8]. Assim como o artigo científico, o documento de patente também é formado por componentes estruturais[9]. Para registrar uma patente, é necessário que examinadores avaliem se o objeto de criação realmente representa uma invenção original. Para isso, eles fazem uma busca prévia nos arquivos de patentes existentes a fim de encontrar as invenções similares à patente avaliada. Apesar de existirem ferramentas especializadas na busca de patentes, como o Google Patents[10], a descoberta de patentes relevantes ainda é um problema que consome muito tempo e esforço.

Xue & Croft [2009] analisam várias alternativas para se gerar consultas para a busca de patentes. Assim como no nosso trabalho, eles consideram partes específicas do documento na geração de consultas. Eles fazem uma análise das combinações das seções da patente, do número de termos por consulta, número de consultas utilizadas entre outras. Também são avaliadas algumas variações de TF e IDF. Takaki et al. [2004] apresentam um trabalho bastante semelhante ao nosso. Eles dividem um segmento de texto de uma patente em sentenças, as quais denominam de subtópicos, e geram con-

[8] http://pt.wikipedia.org/wiki/Patente

[9] http://library.queensu.ca/webeng/patents/Anatomy_of_a_US_patent.pdf (Em Março/2011)

[10] http://www.google.com/patents

sultas a partir desses subtópicos. Há também uma noção de importância de subtópico que é levada em consideração na ordenação final das patentes. Essa é uma abordagem interessante mas não leva em consideração que o número de consultas geradas pode ser muito grande e não propõe uma solução para esse caso. O uso de subtópicos na geração de consultas permite a combinação de termos de forma coerente, e isso é feito neste trabalho através do uso de frases substantivas. Já o uso da importância do subtópico na ordenação dos documentos deverá ser investigada em trabalhos futuros.

1.4 Principais Contribuições

As principais contribuições desta dissertação, também abordadas em [Nascimento et al., 2011], são apresentadas a seguir:

- Proposta de um arcabouço para busca e recomendação de artigos científicos. O arcabouço proposto efetua buscas por meio de formulários disponíveis na Web para encontrar artigos relacionados a um artigo dado como entrada. Essa abordagem possibilita a independência do sistema de recomendação com relação ao armazenamento e à manutenção das fontes de informação. Além disso, o arcabouço viabiliza a utilização de diferentes fontes na recomendação de artigos científicos.

- Estratégias para geração de consultas e recomendação de artigos científicos. Propusemos estratégias para geração de consultas que não dependem de estatísticas globais e que têm potencial para serem utilizadas em artigos de diversas áreas. As estratégias utilizam um único artigo fornecido como entrada pelo usuário, o que reduz o esforço dispendido na busca por artigos relacionados. Também propusemos a utilização de estratégias de recomendação baseadas em conteúdo que se baseiam nos metadados dos artigos científicos disponíveis publicamente.

- Avaliação experimental das estratégias propostas. Avaliamos experimentalmente as estratégias propostas em um possível cenário de recomendação. Nos experimentos realizados, foram avaliadas várias combinações de estratégias para geração de consultas e recomendação de artigos científicos, bem como o impacto do uso de determinados campos do artigo científico na geração de consultas e da utilização de mais de uma fonte no processo de recomendação.

1.5 Organização da Dissertação

A seguir, apresentamos a organização do restante desta dissertação. No Capítulo 2 contextualizamos nosso sistema de recomendação, apresentamos conceitos básicos relacionados, definimos o nosso problema e apresentamos as métricas de avaliação e a carga de trabalho a serem utilizadas nos experimentos. No Capítulo 3, apresentamos a arquitetura e o funcionamento do arcabouço, discutimos algumas questões relacionadas aos processos de extração dos componentes estruturais de um artigo científico, extração dos resultados das fontes de informação e submissão de consultas, sumarizamos os desafios existentes para a automatização completa do arcabouço proposto e apresentamos um protótipo do mesmo desenvolvido como uma interface Web. No Capítulo 4, apresentamos um possível cenário de recomendação e as estratégias propostas para geração e ordenação de consultas e também para ordenação dos artigos científicos a serem recomendados. No Capítulo 5, descrevemos o projeto dos experimentos e apresentamos e analisamos os resultados obtidos. Por fim, no Capítulo 6, concluímos a dissertação e apresentamos algumas propostas de trabalhos futuros.

Capítulo 2

Conceitos Básicos

2.1 Sistemas de Recomendação

Ao se tratar de sistemas de recomendação é importante que três aspectos estejam bem definidos: para quem, o quê e como recomendar. Iniciamos a discussão identificando quem são os alvos da recomendação, depois os tipos de item a serem recomendados e, por fim, a estratégia adotada para recomendar.

Um sistema de recomendação normalmente é construído para atender necessidades de pessoas. Portanto, podemos dizer que o alvo da recomendação é uma pessoa, embora isso possa variar de acordo com o contexto. Neste trabalho de recomendação de artigos, o alvo de cada recomendação normalmente será um estudante, um professor ou um pesquisador em geral que está envolvido com a pesquisa científica.

Apesar de identificarmos que as recomendações serão feitas para pessoas, um sistema de recomendação não entende o que é uma pessoa. Na verdade, o tipo de informação que ele necessita é algo que represente o interesse do alvo da recomendação, isto é, o interesse do estudante, do professor, etc. Na Web, por exemplo, o conteúdo das páginas navegadas ou consultas geradas por um usuário poderiam ser utilizadas para representar seus interesses. Em uma loja virtual, os produtos que um usuário adquire são evidências claras de suas preferências. Existem ambientes em que as pessoas podem indicar explicitamente sua opinião sobre um determinado item, como dar nota para um determinado livro.

Quando se trata de recomendação de artigos científicos, podemos inferir os interesses de uma pessoa através do texto dos artigos escritos ou lidos por ela, das referências bibliográficas utilizadas, dos artigos baixados de uma biblioteca digital, de uma determinada seção de um artigo, dos autores, entre outros. Em nosso sistema de recomendação, utilizaremos um único artigo fornecido pelo usuário para inferir seus

interesses.

Dependendo do contexto, um sistema de recomendação pode sugerir um ou vários tipos de item. Como exemplos de tipos de item, podemos citar páginas da Web em geral, notícias, músicas, vídeos, *posts* de *blogs*, livros, artigos da Wikipedia, pessoas (amigos, artistas, autores, etc), jogos eletrônicos, eletrodomésticos, entre tantos outros. Em uma loja virtual, por exemplo, sugerir apenas livros na seção de livros parece ser a estratégia mais interessante. No entanto, imagine que um usuário esteja navegando na página de descrição do livro "A cabeça de Steve Jobs", cujo assunto é Steve Jobs, cofundador e principal executivo da Apple, uma empresa que produz equipamentos de alta tecnologia. Nesse contexto, poderia ser interessante sugerir, além de livros, produtos fornecidos pela empresa como computadores ou telefones celulares. No nosso caso, limitaremos o arcabouço a recomendar apenas artigos científicos, embora ele possa ser ajustado para outros tipos de item.

Existem dois tipos principais de estratégia que podem ser utilizados em um sistema de recomendação. A primeira é a recomendação baseada em conteúdo, que seleciona itens considerando a correlação entre o conteúdo dos itens e as preferências dos usuários. A segunda, é a recomendação baseada em filtro colaborativo, que seleciona itens considerando a correlação entre usuários que têm preferências similares. Existe ainda um terceiro tipo de estratégia que envolve a combinação das outras duas, que é a recomendação híbrida.

Em tarefas de recomendação de artigos científicos, as estratégias baseadas em conteúdo utilizam apenas o texto dos artigos. Já as estratégias baseadas em colaboração têm duas formas principais de atuar: a primeira é criar um perfil para o usuário, onde ele manifeste suas preferências (por exemplo, lendo uma página, marcando um artigo como lido, ou sendo o autor do artigo) e a segunda forma é inferir a preferência do usuário através da rede de citações, isto é, se um artigo a_i aparece na lista de referências de outro artigo a_j isso indica que a_j tem uma preferência por a_i. Optamos por não utilizar a recomendação colaborativa em nosso trabalho devido a restrições e privilégios de acesso à informação utilizada. Acreditamos que esse tipo de informação (perfil de usuários, rede de citações) é valiosa mas, no entanto, é impraticável construir um sistema de recomendação sem que essa informação esteja disponível publicamente ou que sejamos os editores que publicam e detêm os direitos sobre os artigos.

2.2 Artigo Científico

Um artigo científico é um trabalho acadêmico que apresenta resultados de pesquisa realizada de acordo com alguma metodologia científica e tenha sido publicado em periódicos ou anais de conferências[1]. Para ser publicado, um artigo deve passar por um processo de revisão em que pesquisadores da mesma área o avaliam e decidem sobre a sua publicação. Os revisores analisam sua qualidade e relevância, os métodos utilizados, os resultados obtidos, a adequação do tema estudado ao veículo de publicação considerado, a originalidade, entre outros. O processo de publicação pode demorar vários meses com algumas avaliações, edições e ressubmissões, no caso de periódicos, ou algumas semanas, no caso de conferências, em que o artigo é avaliado somente uma vez.

Para se elaborar um artigo científico é essencial que sejam analisadas as melhores informações sobre o assunto abordado. O que ocorre na prática, embora não seja uma regra obrigatória, é que um artigo científico normalmente cita outro artigo científico como fonte de informação. Isso ocorre por causa da qualidade e confiabilidade associada ao artigo devido ao difícil processo de publicação. Os pesquisadores tentam dar maior credibilidade aos seus trabalhos referenciando artigos já publicados e priorizando aqueles publicados em veículos onde os critérios de avaliação sejam elevados.

Em geral, um artigo científico não é um documento muito longo. Na área de ciência da computação, por exemplo, artigos publicados em conferência normalmente não ultrapassam 15 páginas e artigos publicados em periódicos em geral têm menos de 40 páginas. No entanto, é uma tarefa árdua ler por completo dezenas ou às vezes centenas de artigos para então selecionar os mais adequados.

Uma característica comum de qualquer artigo científico é que ele possui uma certa estrutura. Essa estrutura nem sempre é a mesma mas uma grande parte dos artigos compartilham uma estrutura comum. Acreditamos que os componentes que fazem parte dessa estrutura podem ajudar no processo de recomendação, por isso resolvemos destacá-los. Na próxima seção, descrevemos esses componentes, os quais denominaremos de campos de um artigo científico.

2.3 Campos de um Artigo Científico

Um artigo científico normalmente possui uma estrutura bem definida. Chamaremos as partes que compõem essa estrutura simplesmente de campos, e utilizaremos essa denominação no restante desta dissertação. Os campos mais comuns encontrados em um

[1] http://pt.wikipedia.org/wiki/Artigo_científico

A Source Independent Framework for Research Paper Recommendation

Cristiano Nascimento
Department of Computer Science
Universidade Federal de Minas Gerais
Belo Horizonte, Brazil
crist@dcc.ufmg.br

Alberto H. F. Laender
Department of Computer Science
Universidade Federal de Minas Gerais
Belo Horizonte, Brazil
laender@dcc.ufmg.br

Altigran S. da Silva
Department of Computer Science
Universidade Federal do Amazonas
Manaus, Brazil
alti@dcc.ufam.edu.br

Marcos André Goncalves
Department of Computer Science
Universidade Federal de Minas Gerais
Belo Horizonte, Brazil
mgoncalv@dcc.ufmg.br

ABSTRACT

As the number of research papers available on the Web has increased enourmously over the years, paper recommendation systems have been proposed to help researchers on automatically finding works of interest. The main problem with the current approaches is that they assume that recommending algorithms are provided with a rich set of evidence (e.g., document collections, citations, profiles) which is normally available to only a few publishers, meaning that in most cases the recommendations are restricted to the content covered by these publishers. In this paper we propose a source independent framework for research paper recommendation. The framework requires as input only a single research paper and generates several potential queries by using terms in that paper, which are then submitted to existing Web information sources (e.g., digital libraries) that hold research papers. Once a set of candidate papers for recommendation is generated, the framework applies content-based recommending algorithms to rank the candidates in order to recommend the ones most similar to the input paper, which is done by using only publicly available metadata (i.e., title and abstract). We evaluate our proposed framework by performing an extensive experimenation in which we analyzed several strategies for query generation and several ranking strategies for paper recommendation.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*information filtering, retrieval models, search process*; H.3.7 [**Information Storage and Retrieval**]: Digital Libraries

General Terms

Algorithms, Experimentation

Keywords

Content-Based Filtering, Recommender Systems, Information Seeking, Digital Libraries

1. INTRODUCTION

Bibliographical search is an essential task performed daily by thousands of researchers around the world. This task consists of finding and analysing many scientific documents in order to define a set of works that help to develop a specific research. This set of documents might contain, among others, the (formal) definition of a given research problem that one is trying to solve, a technique useful to be applied in a specific facet of the problem, information to help to understand the context of the research, the history of development of the field, a comparative study among existing techniques, and so on.

A typical strategy used by many researchers in this task is to follow a list of references in documents they already possess. Despite potentially effective in some cases, this is a quite limited approach since there is no guarantee of full coverage and important references may not be cited, besides the fact that only papers published before the current paper can be traced. Finally, even if the researcher is able to find a good set of references she may not be able to actually access those documents if they are not publicly available.

Today there is an increasing amount of research papers available on the Web, thus a significant portion of the bibliographical search task is currently performed in the digital environment. In this environment, researchers make common use of tools such textual search services which allow them to look for interesting documents according to their research interests, usually expressed as queries (set of keywords). After that, they are presented with a result list that contains documents that may be related to their search. One disadvantage of this approach is that it is responsibility of

Figura 2.1. Exemplo de artigo científico em formato PDF: primeira página

artigo científico são: título, lista de autores, resumo, corpo e referências bibliográficas. O corpo de um artigo é dividido ainda em seções e subseções, as quais podem também ser consideradas como campos. Outros campos, além desses básicos, podem ser adicionados a um artigo de acordo com a área de conhecimento e o veículo de publicação.

A Figura 2.1 mostra a primeira página de um artigo científico, onde aparecem alguns campos importantes. No topo da página encontra-se o título do artigo. Logo abaixo do título está a lista de autores. Na coluna à esquerda encontra-se o resumo (*Abstract*) e, logo abaixo, são mostradas as categorias e descritores. Na coluna à direita temos os termos gerais e palavras-chave. Mais abaixo inicia-se o corpo do artigo com uma seção de introdução. Nessa figura podemos observar exemplos de campos que não estão presentes na maioria dos artigos: categorias (*categories*) e descritores (*subject*

descriptors). Esses campos foram inseridos no artigo devido à exigência do veículo de publicação que obriga os autores a utilizar esse formato para se enquadrar em um dos critérios de avaliação.

No exemplo da Figura 2.1, consideramos que os termos em negrito delimitam e determinam os componentes do artigo. No entanto, o conceito de campo pode ser mais amplo. Podemos dizer que os campos representam partes do artigo com uma semântica específica. Por exemplo, o referencial teórico é uma parte importante de qualquer artigo científico, mas nem sempre se encontra em uma seção separada, dividindo espaço com outras partes.

Os artigos científicos atualmente se encontram amplamente publicados em formato eletrônico. Na figura mostrada anteriormente, temos o exemplo de um artigo científico em formato PDF. Como esse formato é um dos mais utilizados, neste trabalho definimos que esse será o formato do artigo fornecido como entrada. Chamamos a atenção ao formato porque isso afeta o modo com que os campos de um artigo são extraídos.

Um outro formato bem comum em que o artigo também pode aparecer é o de uma página Web formatada por meio da linguagem HTML. A Figura 2.2 mostra a página Web de um artigo científico disponível na biblioteca digital da ACM. Ali encontram-se alguns de seus campos na forma de metadados: o título e o resumo. Entretanto, nem todos os seus campos estão disponíveis publicamente. Observe que existe uma URL para o texto completo porém o documento só é disponibilizado para compra. Como faremos buscas em formulários de consulta em bibliotecas digitais, as informações sobre os artigos que iremos obter como resposta estarão formatadas em HTML. Por isso, esse será o outro tipo de formato que consideraremos neste trabalho.

Em trabalhos anteriores, algumas abordagens [Chandrasekaran et al., 2008; Pudhiyaveetil et al., 2009] utilizam informação privilegiada, como registros de perfil de usuários em que somente os proprietários da fonte de informação têm acesso, para fazer a recomendação. Isso dificulta a pesquisa e o desenvolvimento de novas aplicações de recomendação em curto espaço de tempo devido à falta de dados. Por isso, neste trabalho, assumimos que as únicas informações disponíveis para se fazer a recomendação de artigos são o artigo fornecido como entrada pelo usuário e alguns metadados disponíveis publicamente na fonte de informação.

Figura 2.2. Exemplo de artigo científico e alguns de seus campos na forma de metadados armazenados em uma biblioteca digital

2.4 Formulários de Consulta

Formulários Web são formulários feitos para que os usuários forneçam informações a uma aplicação Web. Existem formulários para diversos propósitos, tais como autenticação, publicação de mensagens e consulta. O interesse neste trabalho são os formulários de consulta: formulários que são utilizados para consultar coleções de documentos ou bancos de dados e recuperar itens sobre um determinado assunto.

A Figura 2.3 mostra um exemplo de formulário de consulta de uma biblioteca

digital de artigos científicos. Esse formulário está dividido em duas partes: na parte de cima é possível consultar artigos e na parte de baixo consultar livros. Na parte para consulta a artigos é possível selecionar artigos ou imagens, fazer um consulta genérica na caixa de texto *all fields* ou ainda, por autores, pelo título do veículo de publicação, etc. Na parte para de consulta a livros, é possível escolher em qual campo queremos fazer a consulta, selecionar o assunto de interesse, determinar o período de publicação, etc.

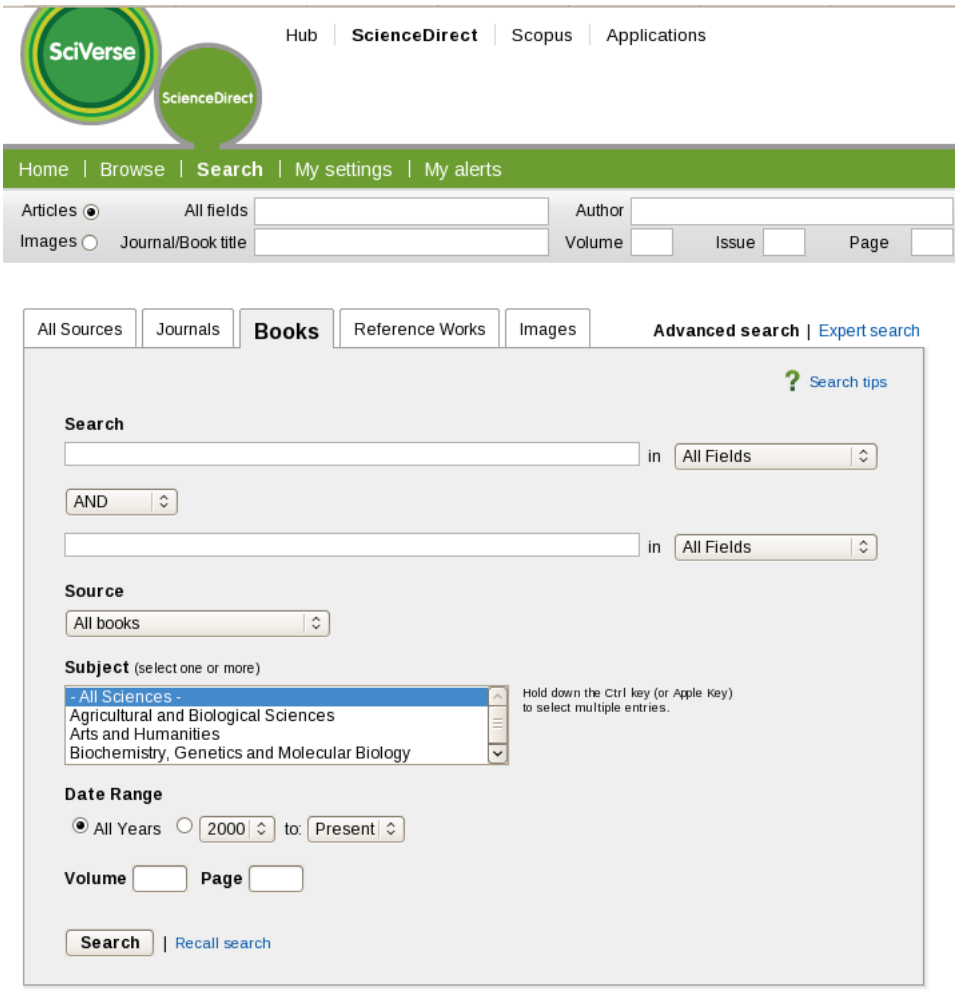


Figura 2.3. Exemplo de formulário de consulta: ScienceDirect

Como vimos, o formulário apresentado na Figura 2.3 apresenta opções de consulta bem refinadas. Na prática, é possível observar que os usuários não têm costume de utilizar tais recursos, preferindo consultas em formulários genéricos (com apenas um campo de consulta) ou utilizando apenas campos genéricos (como *all fields* na Figura 2.3). Formulários com muitas opções restringem o espaço de busca e parecem ser mais úteis quando sabemos exatamente o que estamos procurando. Para consultas

em que se espera reunir o maior número de informações possíveis, como é o caso deste trabalho, o formulário genérico parece ser a melhor opção.

Um outro problema com formulários específicos é que eles são difíceis de serem preenchidos automaticamente. Os formulários Web foram criados para que pessoas os preenchessem a partir da compreensão do rótulo associado a cada campo. Esse tipo de compreensão por parte dos sistemas computacionais representa um grande desafio pois não há um padrão, podendo cada formulário possuir um rótulo diferente para o mesmo tipo de campo. Mesmo que soubéssemos como preencher os campos corretamente, nesse instante não seria interessante aumentar a complexidade do arcabouço proposto criando estratégias de preenchimento de formulários [Toda et al., 2010].

A etapa de busca de artigos científicos em serviços disponíveis na Web por meio do preenchimento de formulários de consulta é uma das principais diferenças do nosso trabalho para as outras abordagens existentes na literatura [McNee et al., 2002; Gori & Pucci, 2006]. Nas abordagens anteriores, os autores utilizam documentos pertencentes a uma única fonte. No entanto, acreditamos que seja mais interessante utilizar documentos de fontes diferentes pois é o que normalmente os usuários fazem na prática. Além disso, essas abordagens já dispõem do conjunto completo de documentos de uma dada fonte. No nosso caso, não possuímos documento algum, sendo necessário buscá-los por meio de interfaces de consulta em diversos serviços disponíveis na Web. As buscas são feitas sempre que um novo documento é fornecido ao arcabouço. Dessa forma, não precisamos nos preocupar com a questão de armazenamento e atualização dos documentos, como é o caso do CiteSeer [Giles et al., 1998]. Além disso, o arcabouço utilizará somente os metadados encontrados nas páginas de resultado, pois nem todos os serviços disponíveis permitem acesso gratuito ao conteúdo completo dos documentos.

2.5 Definição do Problema

Um dos pontos fundamentais na recomendação de artigos científicos é descobrir qual o objetivo dos usuários ao procurar por artigos relacionados. Neste trabalho, embora possamos imaginar que todos os usuários estejam procurando por artigos científicos relacionados, cada um o faz de acordo com um objetivo específico.

Imagine que um artigo X é dado como entrada para o arcabouço. Uma lista de artigos relacionados Y é apresentada a dois usuários, um professor e o outro um aluno. Os dois provavelmente terão visões opostas sobre a lista. Enquanto o professor pode achar a lista pouco atrativa por apresentar artigos que já conhece, o aluno

pode ficar muito satisfeito por ser apresentado a artigos "clássicos" da área. Isto ocorre porque ambos têm níveis de conhecimento diferentes e, consequentemente, objetivos diferentes. Assim sendo, observamos que o objetivo dos usuários é fundamental para definir o critério com que os artigos deverão ser recomendados.

A seguir, listamos alguns dos possíveis focos na recomendação de artigos científicos: *aspecto temporal*, como no exemplo acima, os usuários podem estar interessados em artigos mais recentes ou mais antigos; *especificidade*, o artigo pode ser mais geral ou mais específico; *técnica ou problema abordado*, os artigos relacionados podem ter o foco na técnica utilizada no artigo de referência ou no problema abordado por ele; *alvo da recomendação*, pode-se recomendar artigos relacionados para um artigo específico [Strohman et al., 2007], para um conjunto de artigos Chandrasekaran et al. [2008], para uma parte do texto de um artigo [He et al., 2010], entre outros.

Um sistema de recomendação que faça a mesma recomendação para todos os usuários talvez não seja o ideal. McNee et al. [2006] chamam a atenção exatamente para o fato de que diferentes interesses podem exigir diferentes abordagens de recomendação. A maneira com que o arcabouço é concebido permite que algoritmos de terceiros sejam incorporados a cada módulo, inclusive o de recomendação. Dessa forma, um usuário poderia baixar pacotes de recomendação que atendam suas necessidades sem depender que os detentores das fontes forneçam tais soluções. Isso pode ser muito interessante pois abre a possibilidade de melhorias e diversidade na recomendação.

Assim sendo, definimos o nosso problema da seguinte forma:

> *Dado um artigo de entrada x e um conjunto F de fontes F_j de uma área do conhecimento A, encontre um conjunto de artigos R que são os mais relacionados a x considerando um dado critério c. Cada artigo x_i pertencente a uma fonte F_j qualquer é representado por um conjunto de metadados M.*

Incluímos o critério c na definição do problema justamente para considerar o que discutimos anteriormente: diferentes usuários possuem diferentes pontos de vista e, por isso, visam objetivos diferentes na recomendação de artigos. Dessa forma, podemos lidar com a recomendação de artigos de acordo com o ponto de vista escolhido.

2.6 Métricas de Avaliação

Nesta seção, apresentamos as métricas com as quais avaliaremos nosso arcabouço. Escolhemos três métricas de avaliação que são amplamente utilizadas na literatura

[Baeza-Yates & Ribeiro-Neto, 2011; Järvelin & Kekäläinen, 2002]: revocação, DCG (Discounted Cumulated Gain) e NDCG (Normalized DCG).

Revocação. Determina a fração de documentos relevantes recuperados sobre o total de documentos relevantes que deveriam ter sido recuperados.

$$Revocação = \frac{Exemplos\ Positivos\ Recuperados}{Total\ de\ Exemplos\ Positivos} \tag{2.1}$$

Nos experimentos, as avaliações foram feitas considerando três níveis de relevância: muito relacionado, relacionado e não relacionado. Consideramos os dois primeiros níveis como positivos e o terceiro como negativo.

DCG (Discounted Cumulated Gain). É uma métrica que se utiliza normalmente quando se tem uma escala de avaliações graduada, como é o nosso caso. Essa métrica mede quão boa uma ordenação é considerando se os documentos mais relevantes estão no topo e também penalizando quando eles aparecem em posições mais baixas na ordenação. O DCG é calculado da seguinte maneira:

$$DCG_p = rel_1 + \sum_{i=2}^{p} \frac{rel_i}{log_2 i} \tag{2.2}$$

onde p é a posição onde o DCG é calculado, rel_k é o valor da relevância do documento na posição k. Configuramos os níveis de relevância da seguinte forma: documentos avaliados como "muito relacionado" receberam valor 2, os "relacionados" receberam 1 e os "não relacionados" receberam 0.

NDCG (Normalized DCG). Com essa métrica podemos saber quão próxima, ou quão distante uma dada ordenação está da ordenação ideal. Ela é calculada da seguinte forma:

$$NDCG_p = \frac{DCG_p}{IDCG_p} \tag{2.3}$$

onde IDCG_p é o DCG calculado sobre a ordenação ideal, na qual os documentos mais relevantes, com graduações mais elevadas, são colocados no topo e p é a posição em que o DCG é calculado.

2.7 Carga de Trabalho

Como mencionamos no Capítulo 1, uma das opções para se encontrar um artigo científico é consultar máquinas de busca. No entanto, uma máquina de busca de propósito geral retorna itens que não são necessariamente artigos científicos. Isso diminui a qualidade dos resultados devido à poluição. Como abordado na Seção 2.1, em um primeiro instante estamos interessados em recomendar um tipo de item específico, que é o artigo científico. Por isso, decidimos utilizar fontes de informação especializadas em literatura científica. Neste trabalho, utilizamos duas bibliotecas digitais bem conhecidas na área de Ciência da Computação, a biblioteca digital da ACM e o IEEE Xplore[2], e também um serviço oferecido por uma editora internacional que permite a busca por publicações em diversas áreas do conhecimento, incluindo Ciência da Computação, que é a plataforma ScienceDirect.

A biblioteca digital da ACM indexa artigos de periódicos e anais de conferências sobre tópicos em computação como redes de computadores, engenharia de software, arquitetura de computadores, Internet, entre outros. Essa biblioteca digital é uma coleção completa de todas as publicações da ACM, incluindo periódicos, anais de conferências, revistas, *newsletters* e títulos multimídia. A biblioteca digital da ACM contém a maior e mais completa coleção de artigos completos de computação atualmente: são dois milhões de páginas de texto completo, 272.000 artigos, 20.000 anais, 9 revistas, entre outros. Ela permite buscas genéricas e buscas avançadas, e apresenta as informações dos artigos de forma bastante rica: na página de exibição de um artigo ela mostra suas referências e quais artigos já o citaram além das categorias nas quais o artigo está incluso, por exemplo.

IEEE Xplore é uma ferramenta de acesso *online* a publicações técnicas e científicas publicadas pelo Institute of Electrical and Electronics Engineers (IEEE) e seus parceiros. O IEEE Xplore provê acesso a uma coleção de texto completo em formato PDF de artigos de alguns dos periódicos mais citados em engenharia elétrica, ciência da computação e eletrônica. O repositório sobre o qual o IEEE Xplore atua contém mais de dois milhões de artigos publicados em mais de 12.000 periódicos, anais de conferências e normas técnicas. A coleção é atualizada continuamente e pode ser acessada através de navegadores Web por meio de formulários de consulta. É possível consultar a coleção utilizando-se palavras-chave, frases, título de artigos, autores, entre outros. Uma vez localizados é possível ver, baixar e imprimir artigos, resultados de consultas, sumários e outros.

A plataforma ScienceDirect, da editora Elsevier, oferece uma coleção também

[2]http://ieeexplore.ieee.org

de texto completo de artigos e capítulos de livros de várias áreas do conhecimento como engenharia (incluindo computação), ciências da vida, ciências médicas e ciências sociais. São mais de 2.500 periódicos e 11.000 livros. Atualmente existem cerca de 9,5 milhões de artigos/capítulos e a taxa de crescimento é de cerca de 0,5 milhões adições por ano. A plataforma oferece opções de busca genérica e avançada, além de fornecer diversas ferramentas que enriquecem a experiência do usuário, inclusive com um sistema de recomendação de artigos.

Capítulo 3

Arcabouço Proposto

3.1 Visão Geral

Seja na tarefa de pesquisa bibliográfica ou em uma navegação casual em uma biblioteca digital, é comum que um pesquisador encontre um ou mais artigos que se encaixem em seu perfil de pesquisa. Após identificar que se trata realmente de um artigo de seu interesse, o pesquisador pode querer expandir o seu conhecimento procurando por outros artigos relacionados a esse artigo de referência. Para isso, ele pode utilizar as palavras-chave encontradas nesse artigo como entrada para formulários de consulta disponíveis na Web, o que de fato ocorre com muita frequência.

Após realizar cada consulta, o pesquisador deve analisar a lista de artigos retornados para identificar aqueles relacionados. Dado que ler por completo um artigo científico para determinar a sua relevância geralmente representa um grande esforço, esse pesquisador pode adotar alguma estratégia que facilite o processo de filtragem, como, por exemplo, se basear somente no título ou no resumo do artigo para decidir se deve baixá-lo da fonte de informação para então fazer a leitura completa. Em outros casos, o pesquisador se vê obrigado a usar essa mesma estratégia por falta de opção, já que o texto completo dos artigos científicos nem sempre está disponível publicamente.

Observando esse comportamento típico de um pesquisador que realiza manualmente sua pesquisa bibliográfica na Web, propomos um arcabouço para busca e recomendação de artigos científicos. O arcabouço gera consultas automaticamente, recupera uma lista de artigos potencialmente relacionados a um artigo fornecido como entrada e sugere aqueles mais relacionados a partir dos mesmos metadados que o pesquisador tem à sua disposição publicamente nas fontes de informação.

A arquitetura e o funcionamento geral do arcabouço são mostrados na Figura 3.1. O arcabouço recebe como entrada um artigo científico sobre o qual o usuário tenha al-

gum interesse. A utilização de um único artigo diminui o esforço por parte do usuário que não precisa gerar suas próprias consultas para encontrar artigos relacionados. Além disso, acreditamos que um único artigo tem o potencial para identificar de maneira precisa o interesse imediato do usuário.

O artigo de entrada é então fornecido ao módulo de extração de campos, que é o responsável por extrair os principais campos de um artigo científico (Seção 2.3). A separação dos campos é importante para a geração de consultas pois desse modo é possível selecionar de que parte do texto as consultas serão extraídas, aproveitando as vantagens inerentes à semântica de cada campo considerado.

A seguir, o conjunto de campos é fornecido ao módulo de geração de consultas. Mais do que gerar consultas, este módulo é responsável por identificar conceitos dentro do artigo científico. Diferentemente dos trabalhos de coleta na Web oculta [Raghavan & Garcia-Molina, 2001; Vieira et al., 2008], nosso objetivo não é fazer uma coleta exaustiva nas fontes de informação. Vieira et al. [2008], por exemplo, adotam estratégias de geração de consultas que visam recuperar o maior número de documentos possível e, para isso, geram consultas genéricas e sem nenhuma semântica específica. Isso pode ser uma boa estratégia no contexto de coleta de conteúdo Web, mas não é a mais apropriada para este trabalho. No nosso caso, extraímos conceitos do artigo de entrada de maneira a identificar seus assuntos principais e, consequentemente, identificar o interesse do usuário. As estratégias de geração de consultas estão definidas na Seção 4.2.1

Como a quantidade de consultas geradas pode ser muito grande, é necessário selecionar aquelas que melhor representem o conteúdo do artigo de entrada. Basicamente, a etapa de seleção de consultas consiste em atribuir um peso a cada uma delas e então selecionar as C consultas de maior peso (ver Seção 4.2.2). Dessa forma, evita-se que sejam feitas muitas requisições às fontes de informação, diminuindo o tráfego na rede e reduzindo o tempo de resposta aos usuários.

Uma vez que as consultas tenham sido selecionadas, elas são fornecidas ao módulo de submissão de consultas que efetua buscas por meio de formulários de consulta disponíveis na Web. A utilização desses formulários para a descoberta de artigos relacionados é um dos grandes diferenciais deste trabalho em relação àqueles apresentados na Seção 1.3. Em primeiro lugar, isso permite que novas fontes de informação sejam adicionadas facilmente ao arcabouço e também que os usuários escolham de que fontes gostariam de receber as recomendações. Nos trabalhos descritos na Seção 1.3, as sugestões de artigos são feitas sobre coleções definidas previamente. Em segundo lugar, não há necessidade de se armazenar coleções de artigos, dessa forma também não há necessidade de o arcabouço ser mantido como um servidor centralizado, permitindo que cada usuário possa ter a sua própria versão do arcabouço instalado em sua máquina e

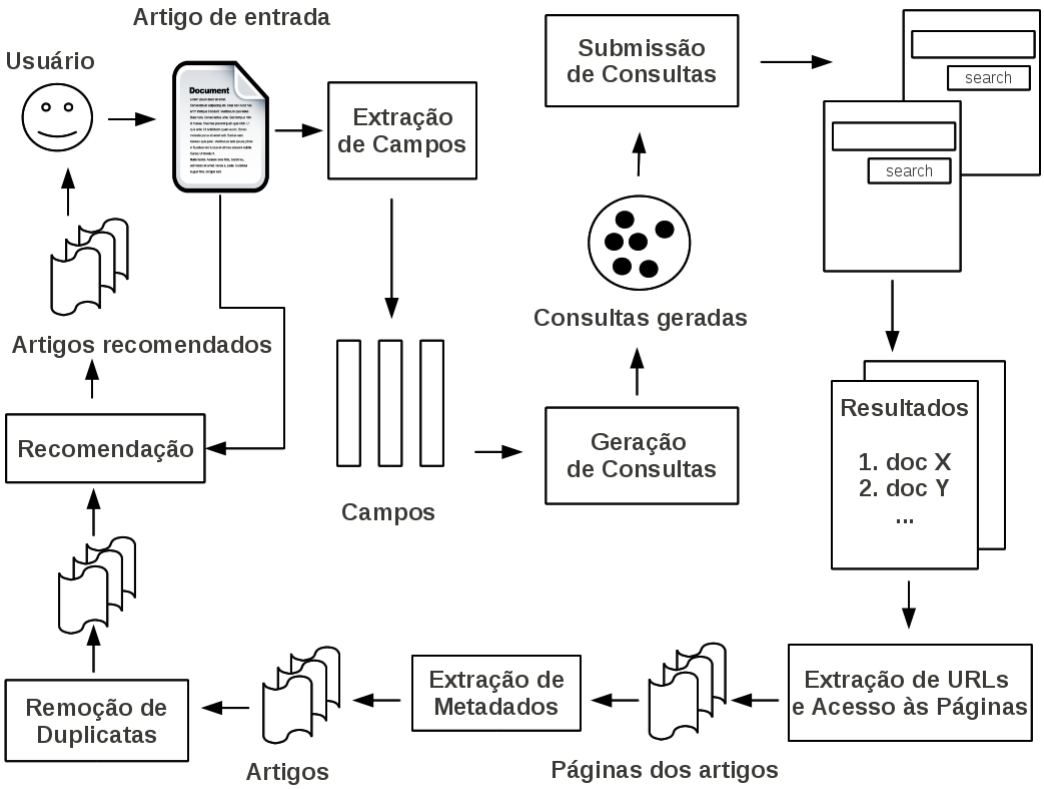


Figura 3.1. Arquitetura e funcionamento do arcabouço proposto

o configure de acordo com suas necessidades. Em terceiro lugar, não precisamos lidar com a atualização das coleções pois as consultas serão realizadas em fontes de informação que já fornecem os artigos mais recentes.

O processo de obtenção de artigos por meio de formulários de consulta é descrito mais detalhadamente a seguir. Após receber as consultas geradas, o módulo de submissão as submete a várias fontes de informação. Para cada consulta submetida, uma determinada fonte retorna uma página de resultados na qual é exibida uma lista de artigos retornados pela consulta. Nessa página de resultados, estão disponíveis pelo menos o título do artigo e uma URL que aponta para uma página que contém outras informações sobre ele (o resumo, por exemplo). Como necessitamos dessas outras informações, é preciso extrair a URL desse artigo e acessar a página apontada por ela. Essa tarefa é executada pelo módulo de extração de URLs e acesso às páginas. No final, obtemos um conjunto de páginas HTML com informação de cada artigo retornado.

Por fim, existe um pré-processamento antes de fornecer os artigos para o módulo de recomendação. Primeiro, o módulo de extração de metadados retira das páginas HTML somente os metadados relacionados ao conteúdo do artigo correspondente, como título e resumo. Depois, como um mesmo artigo pode ter sido retornado por mais de

fonte ou para mais de uma consulta, é realizada uma remoção de duplicatas. Finalmente, o módulo de recomendação recebe os artigos candidatos a recomendação e sugere aqueles mais relacionados de acordo com um critério definido previamente.

3.2 Extração dos Campos do Artigo de Entrada

A extração automática dos campos do artigo de entrada é fundamental para que possamos escolher de que parte do artigo os conceitos utilizados em uma consulta serão extraídos. Essa tarefa é muito importante pois a extração incorreta pode levar a recomendações ruins mesmo se o sistema como um todo funcionar perfeitamente. Neste trabalho, assumimos que o artigo de entrada se encontra no formato PDF, embora outros formatos também possam ser utilizados. A seguir, discutimos a extração de campos de um artigo em formato PDF.

Para a extração dos campos do artigo de entrada, podemos recorrer a estratégias comumente utilizadas para a extração de metadados de artigos científicos. Essas estratégias são em geral bastante robustas uma vez que em sua maioria são baseadas em técnicas de aprendizado de máquina. Por exemplo, Han et al. [2003] utilizam Support Vector Machines (SVM), Takasu [2003] Hidden Markov Models (HMM) e Peng & McCallum [2006] Conditional Random Fields (CRF). Apesar de bastante eficazes, estratégias baseadas em aprendizado de máquina requerem muito esforço de implementação.

Uma estratégia alternativa, também adotada frequentemente, é a utilização de *parsers* baseados em regras de extração definidas pelo usuário. Embora não seja uma estratégia tão robusta quanto aquelas baseadas em aprendizado de máquina, é bastante eficaz em contextos específicos e também relativamente mais fácil de ser implementada. O problema desse tipo de abordagem é que ela necessita que um especialista descubra padrões nos documentos e elabore as regras de extração.

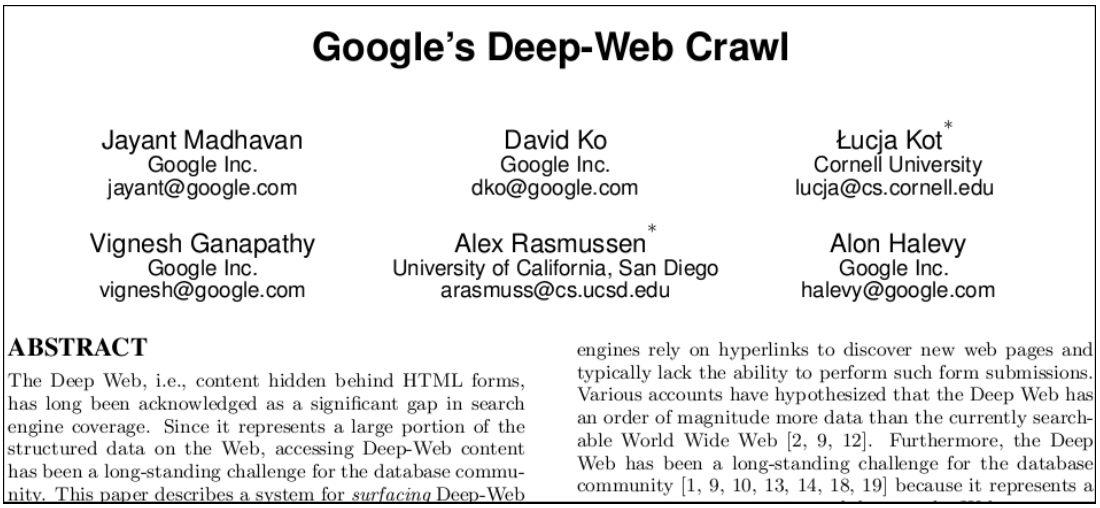
Nesta proposta inicial do arcabouço, optamos pela utilização de uma estratégia baseada em regras de extração. Embora menos robusta, uma vez que depende de intervenção humana, ela atende às nossas necessidades dentro deste trabalho. Nossa estratégia, desenvolvida de forma independente, é idêntica a uma estratégia proposta por Beel et al. [2010] para a extração do título de artigos científicos. A extração é baseada na ferramenta *pdftohtml*[1], que converte arquivos no formato PDF para os formatos HTML e XML e mantém informações sobre a formatação do texto (tamanho e nome da fonte, negrito, itálico, etc) e a posição em que ele ocorre (número da página,

[1] http://pdftohtml.sourceforge.net/

posição relativa ao topo da página, etc).

A Figura 3.2 mostra um trecho de um artigo PDF fornecido como entrada para a ferramenta pdftohtml. A figura mostra o título, a lista de autores e o ínicio do resumo que estão localizados na primeira página. A Figura 3.3 mostra um trecho da saída do mesmo artigo em formato XML após a utilização do *pdftohtml*.

Na Figura 3.3, a *tag* **page** contém um atributo chamado **number** que indica o



Google's Deep-Web Crawl

Jayant Madhavan
Google Inc.
jayant@google.com

David Ko
Google Inc.
dko@google.com

Łucja Kot*
Cornell University
lucja@cs.cornell.edu

Vignesh Ganapathy
Google Inc.
vignesh@google.com

Alex Rasmussen*
University of California, San Diego
arasmuss@cs.ucsd.edu

Alon Halevy
Google Inc.
halevy@google.com

ABSTRACT

The Deep Web, i.e., content hidden behind HTML forms, has long been acknowledged as a significant gap in search engine coverage. Since it represents a large portion of the structured data on the Web, accessing Deep-Web content has been a long-standing challenge for the database community. This paper describes a system for *surfacing* Deep-Web

engines rely on hyperlinks to discover new web pages and typically lack the ability to perform such form submissions. Various accounts have hypothesized that the Deep Web has an order of magnitude more data than the currently searchable World Wide Web [2, 9, 12]. Furthermore, the Deep Web has been a long-standing challenge for the database community [1, 9, 10, 13, 14, 18, 19] because it represents a

Figura 3.2. Trecho de um artigo PDF fornecido como entrada para a ferramenta pdftohtml

número da página. A *tag* **fontspec** identifica as diferentes configurações de formatação utilizadas no texto e **text** contém o texto que aparece no artigo em formato PDF. Em cada **text** existe um atributo **font** que contém o valor do atributo **id** de um **fontspec**, isto é, o atributo identifica o estilo de formatação utilizado no texto dentro do seu **text**.

Beel et al. [2010] identificam os títulos utilizando duas informações: (1) o título de um artigo sempre está localizado na primeira página e (2) o título normalmente possui a fonte de maior tamanho no texto. Assim, basta extrair todos os textos da primeira página cujo atributo **font** seja igual ao **id** do **fontspec** que possui o maior valor de **size**. Na Figura 3.3, o maior tamanho de fonte é 15 no **fontspec** de **id** igual a 0 e assim, o texto "Google's Deep-Web Crawl" seria extraído como o título do artigo.

A extração dos demais campos é feita de maneira similar. No entanto, como não existe um padrão de formatação muito bem definido para os outros campos, também utilizamos heurísticas baseadas em palavras-chave para identificá-los. Por exemplo, definimos o resumo como o segmento de texto que está entre duas palavras formatadas como negrito sendo que a primeira palavra é a palavra-chave *resumo*.

```
−<pdf2xml>
 −<page number="1" position="absolute" top="0" left="0" height="841" width="595">
    <fontspec id="0" size="15" family="Times" color="#000000"/>
    <fontspec id="1" size="9" family="Times" color="#000000"/>
    <fontspec id="2" size="6" family="Times" color="#000000"/>
    <fontspec id="3" size="7" family="Times" color="#000000"/>
    <fontspec id="4" size="5" family="Times" color="#000000"/>
    <fontspec id="5" size="10" family="Helvetica" color="#000000"/>
    <fontspec id="6" size="6" family="Helvetica" color="#000000"/>
   −<text top="72" left="194" width="221" height="17" font="0">
      <b>Google's Deep-Web Crawl</b>
    </text>
    <text top="118" left="97" width="93" height="11" font="1">Jayant Madhavan</text>
    <text top="118" left="283" width="48" height="11" font="1">David Ko</text>
    <text top="118" left="437" width="49" height="11" font="1">Łucja Kot</text>
    <text top="111" left="486" width="5" height="8" font="2">∗</text>
    <text top="130" left="118" width="51" height="9" font="3">Google Inc.</text>
    <text top="130" left="282" width="51" height="9" font="3">Google Inc.</text>
    <text top="130" left="426" width="78" height="9" font="3">Cornell University</text>
    <text top="140" left="100" width="88" height="9" font="3">jayant@google.com</text>
    <text top="140" left="268" width="77" height="9" font="3">dko@google.com</text>
```

Figura 3.3. XML de artigo PDF convertido com a ferramenta pdftohtml

3.3 Extração dos Metadados dos Artigos Retornados

Para obter os metadados dos artigos retornados, são necessárias duas extrações: primeiro na página de resultados e depois na página individual de cada artigo.

Quando uma consulta é submetida a uma determinada fonte, obtém-se como resposta uma página de resultados. Um exemplo de página de resultados pode ser vista na Figura 3.4. Embora a página de resultados contenha várias informações sobre o artigo retornado, nem todos os metadados estão presentes nessa página e o metadado mais importante, que é o resumo, não está completo. Por isso, primeiro é preciso extrair a URL de cada artigo na página de resultados, baixar a página individual do artigo apontado pela URL extraída e extrair os metadados do artigo da página correspondente.

Para extrair as URLs dos artigos em uma página de resultados, analisamos o código HTML de cada fonte e elaboramos uma regra de extração para cada uma delas. Como as páginas dos artigos são geradas dinamicamente, suas URLs compartilham um endereço base que pode ser utilizado para identificá-las em meio às demais URLs na página de resultados. A URL **http://portal.acm.org/citation.cfm?id=1410140.1410180**, por exemplo, aponta para a página de um artigo na biblioteca digital da ACM. Todas as URLs de páginas de artigos nessa biblioteca compartilham o endereço base

http://portal.acm.org/citation.cfm, diferindo apenas no valor do atributo **id**. Outros tipos de página, como as que exibem informação de um determinado autor, possuem URLs base diferentes. Sendo assim, utilizamos a URL base como regra de extração. Observe que, embora tenhamos utilizado a biblioteca digital da ACM no exemplo anterior, o mesmo procedimento pode ser aplicado a outras fontes de informação, como IEEE Xplore e ScienceDirect, também utilizadas neste trabalho (ver Seção 2.7)

Uma outra maneira interessante de extrair informações de documentos HTML é

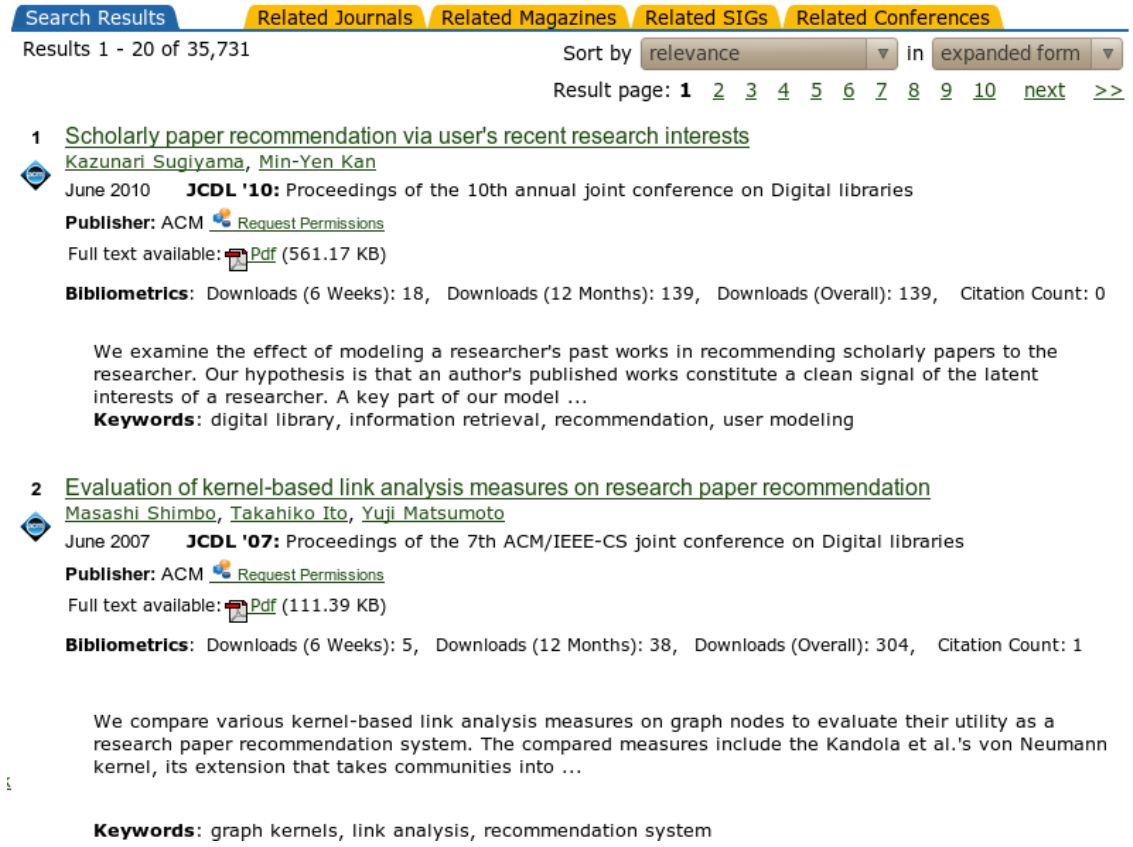


Figura 3.4. Página de resultado de uma consulta

usar sua estrutura de árvore para derivar consultas XPath[2] e então utilizar as consultas para obter as informações do documento [Dalvi et al., 2009]. Neste trabalho, utilizamos uma estratégia baseada em XPath para extrair metadados da página de cada artigo. Adotamos a ferramenta XPath Checker[3] para gerar a expressão XPath que retorna o conteúdo do metadado desejado. Para isso, basta instalar a ferramenta no navegador Firefox, selecionar a informação a ser extraída de uma determinada página, clicar com o botão direito e selecionar a opção "View Path". Uma vez obtida a consulta XPath,

[2]http://www.w3.org/TR/xpath20/
[3]https://addons.mozilla.org/en-US/firefox/addon/xpath-checker/

o passo seguinte é utilizar um interpretador dessa linguagem, que normalmente está disponível como uma biblioteca em várias linguagens de programação, e executá-la para obter o metadado em um dado documento. Nesse caso, a consulta extraída é a nossa regra de extração e pode ser utilizada para todos os artigos dentro do domínio de uma fonte.

3.4 Submissão de Consultas

Para submeter as consultas automaticamente aos formulários disponíveis nas fontes de informação, como, por exemplo o formulário mostrado na Figura 2.3, é preciso conhecer alguns dos componentes desses formulários que não são visíveis na página do navegador. A interface do formulário que é mostrada em um navegador serve simplesmente para facilitar a interação com os usuários, sendo que a informação real utilizada na submissão encontra-se em seu código HTML.

A submissão de um formulário HTML é simplesmente uma requisição HTTP que envia um conjunto de pares controle-valor a um determinado servidor. Para simular a submissão de um formulário em um navegador basta descobrir qual é o servidor que responde ao formulário, o tipo de requisição HTTP utilizada e o nome dos controles presentes no formulário. A seguir, mostraremos onde essas informações podem ser encontradas.

A Figura 3.5 mostra um exemplo de formulário exibido em um navegador e a

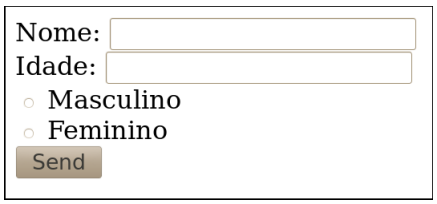


Figura 3.5. Formulário HTML exibido em um navegador

Figura 3.6 o seu código HTML. No elemento **form** do código HTML, o atributo **action** contém a URL do servidor que atende à submissão do formulário e o atributo **method** indica o tipo de requisição HTTP que deve ser utilizada (**get** ou **post**). Os controles são os objetos com os quais um usuário interage inserindo ou modificando valores. Na Figura 3.5, os controles correspondem às caixas de texto e aos botões de rádio. Já na Figura 3.6, os controles correspondem aos elementos **input**. O nome do controle que é utilizado na submissão encontra-se em seu atributo **name**.

Com essas informações, é possível submeter um formulário automaticamente utilizando qualquer ferramenta que permita o envio de requisições HTTP com os métodos

get ou **post**, como o **GNU Wget**[4], associando aos nomes dos controles valores de um conjunto pré-definido.

No protótipo implementado (Seção 3.7), o módulo de submissão de consultas

```
<FORM action="http://meusite.com/usuario/adicionar" method="post">
   <LABEL for="unome">Nome: </LABEL>
   <INPUT type="text" name="nome" id="unome"><BR>
   <LABEL for="uidade">Idade: </LABEL>
   <INPUT type="text" name="idade" id="uidade"><BR>
   <INPUT type="radio" name="sexo" value="M"> Masculino<BR>
   <INPUT type="radio" name="sexo" value="F"> Feminino<BR>
   <INPUT type="submit" value="Send">
</FORM>
```

Figura 3.6. Código fonte do formulário HTML da Figura 3.5

considera apenas três fontes, que são aquelas apresentadas na Seção 2.7 e as mesmas utilizadas nos experimentos. No entanto, é possível adicionar novas fontes, desde que se implemente os módulos de submissão de consultas e extração de metadados para cada uma delas.

3.5 Remoção de Duplicatas

Ao submeter várias consultas a diversas fontes de informação podem ocorrer situações em que um mesmo artigo seja retornado em mais de uma página de resultados. Nesse caso, é preciso fazer a remoção de duplicatas pois um mesmo artigo poderia aparecer mais de uma vez na lista recomendada para um usuário.

Neste trabalho existem duas situações principais que podem fazer com que tenhamos artigos replicados no conjunto candidato à recomendação. A primeira ocorre ao enviar consultas similares a uma mesma fonte de informação. Nesse caso, as consultas podem compartilhar termos em comum ou se referirem a tópicos muito relacionados, o que aumenta a probabilidade de retornarem artigos similares ou idênticos. A segunda situação ocorre ao enviar a mesma consulta a fontes de informação diferentes. Nesse caso, se as fontes compartilham um subconjunto de artigos em comum, então é possível que retornem os mesmos artigos para uma determinada consulta.

No primeiro caso, a remoção de duplicatas é relativamente fácil pois a informação disponível sobre os artigos recomendados provém exatamente da mesma fonte. Assim, o casamento exato de alguns metadados já é suficiente para identificar réplicas. Por exemplo, com o casamento dos títulos e das listas de autores é possível identificar todas as réplicas.

[4] http://www.gnu.org/software/wget/

No segundo caso, pode haver necessidade de um esforço maior pois as fontes de informação podem divergir quanto à informação disponível para um determinado artigo. Por exemplo, o nome dos autores pode aparecer abreviado em uma fonte e em outra não. Um outro problema pode acontecer se as fontes incluírem informação divergente sobre o título de um mesmo artigo, como, por exemplo, a falta de uma palavra ou a existência de algum símbolo codificado de forma diferente. Como as fontes utilizadas neste trabalho não apresentaram grandes variações na informação de um determinado artigo, o casamento exato resolveu a maioria dos casos de remoção de duplicatas.

3.6 Desafios na Automatização do Arcabouço

Acreditamos que uma característica bastante interessante do arcabouço proposto é a possibilidade de utilização de várias fontes para se fazer a recomendação. No entanto, a inserção de novas fontes ainda é uma tarefa que requer um certo esforço de desenvolvimento já que é preciso criar módulos adaptados para cada fonte. O ideal seria, por exemplo, que um usuário tivesse a possibilidade de fornecer uma URL de uma determinada fonte e, a partir disso, o arcabouço gerasse automaticamente módulos adaptados a essa fonte. A seguir, apresentamos as etapas necessárias para a realização dessa tarefa e quais são os possíveis problemas a serem resolvidos.

O primeiro passo é a extração do formulário de consulta da página HTML. A tarefa de extração é relativamente simples, basta extrair todo o conteúdo inserido no elemento **form**. No entanto, como a página pode ter mais de um formulário, é preciso identificar corretamente o formulário que efetua buscas. Para isso, uma possível estratégia que pode ser utilizada é identificar palavras-chave, como "search", "query" ou "find", dentro ou ao redor do formulário [Wu et al., 2003].

O segundo passo é a identificação dos elementos necessários à submissão automática. Na Seção 3.4 vimos que esses elementos são: a URL do servidor que responde a consulta, o tipo de requisição HTTP utilizada e o controle responsável por receber as consultas. Existem dois pontos que tornam essa tarefa particularmente difícil. O primeiro é a utilização de conteúdo dinâmico nas páginas HTML. Pode acontecer, por exemplo, de o formulário ser submetido via JavaScript e a URL do servidor não estar disponível no atributo **action** do elemento **form**. O segundo ponto é que o formulário pode conter vários controles além daquele que recebe a consulta. A identificação dos controles e seu preenchimento automático é um problema conhecido e não possui uma solução trivial [Lage et al., 2004].

O último passo é a extração de URLs na página de resultados e a extração de

metadados dos artigos retornados. Essas duas tarefas foram abordadas na Seção 3.3. Conforme foi observado, as URLs na página de resultados de uma determinada fonte compartilham um prefixo comum. O problema nesse caso seria identificar automaticamente esse prefixo. No caso da extração de metadados, como uma fonte difere de outra quanto ao estilo e a formatação das páginas, a regra de extração de uma fonte normalmente não se aplica a outras. Uma possível estratégia para extração dos metadados seria observar os padrões das páginas em uma mesma fonte. Como as páginas de um mesma fonte são formatadas da mesma forma, bastaria observar as estruturas que se repetem em todas as páginas e removê-las do documento HTML. No final, restariam apenas os metadados do artigo científico. Observe que nesse caso ainda seria necessário identificar cada metadado extraído.

3.7 Protótipo

Nesta dissertação, levantamos algumas questões que tornam as abordagens para recomendação de artigos científicos presentes na literatura inviáveis de serem implementadas. Por isso, propusemos um arcabouço que levasse em consideração aspectos práticos e que permitissem àqueles que não têm privilégio de acesso a informação fornecer soluções em recomendação de artigos científicos.

Nesta seção, apresentamos um protótipo que demonstra a viabilidade do arcabouço proposto. A Figura 3.7 mostra a interface inicial do protótipo. Trate-se de um sistema Web, acessado via navegador, que recebe como entrada um artigo em formato PDF fornecido pelo usuário e mostra uma lista de artigos relacionados ao artigo.

Embora o protótipo tenha sido implementado em uma arquitetura cliente-

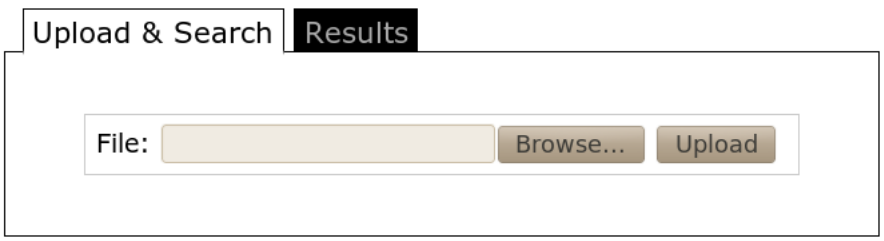


Figura 3.7. Interface de entrada do protótipo do arcabouço proposto

servidor, ele pode ser viabilizado totalmente como uma aplicação *standalone* na parte do cliente. Esse tipo de arquitetura permitiu que, com poucas modificações, ele também pudesse ser utilizado na preparação dos experimentos (Seção 5.2).

O usuário inicia sua interação com o sistema clicando no botão “Browse” e en-

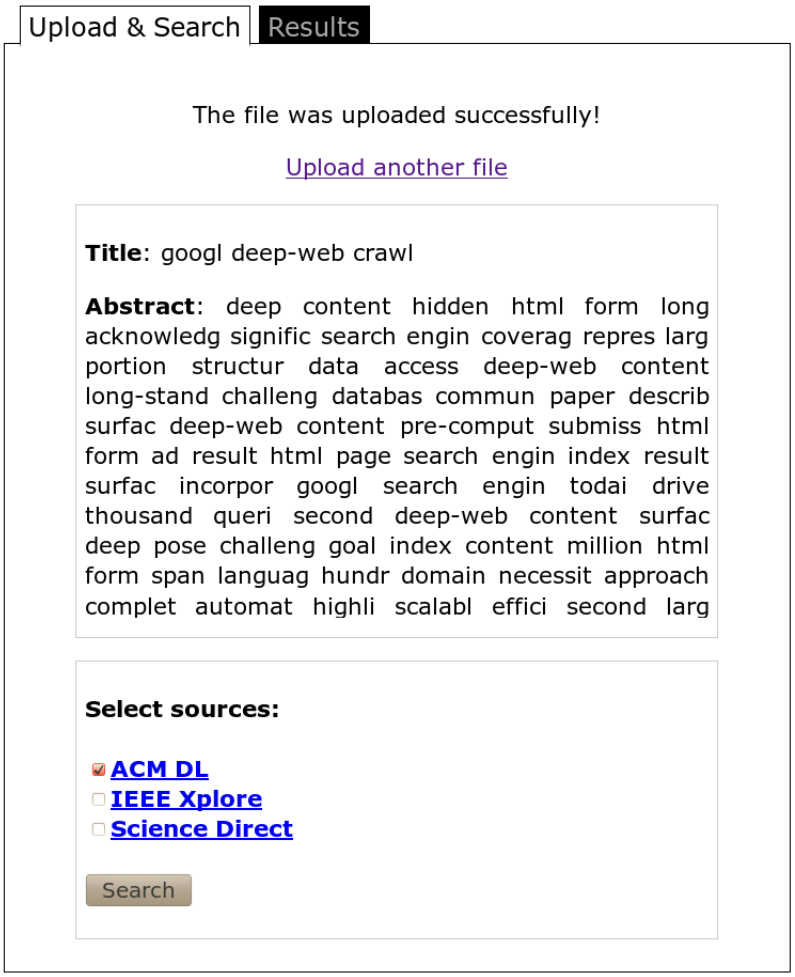


Figura 3.8. Protótipo: resultado da extração de campos e seleção de fontes

tão selecionando o artigo para o qual as recomendações serão feitas. No caso deste protótipo, o artigo a ser fornecido deverá estar disponível no computador pessoal do usuário. Após selecionar o artigo, basta clicar no botão "Upload" e o artigo será enviado para o sistema.

Depois de o usuário fornecer o artigo para o sistema, é mostrada a página que aparece na Figura 3.8. Nessa página, são exibidos os campos extraídos do artigo científico (a extração é feita conforme descrito na Seção 3.2) e uma lista de fontes na qual o usuário pode escolher de onde deseja receber as recomendações. Para receber as recomendações, o usuário deve selecionar pelo menos uma das fontes listadas e então clicar no botão "Search".

Depois que o usuário clica no botão "Search", o sistema submete as consultas geradas (Seção 4.2) às fontes selecionadas, processa os resultados e então exibe uma lista de artigos recomendados. A Figura 3.9 mostra um exemplo das recomendações

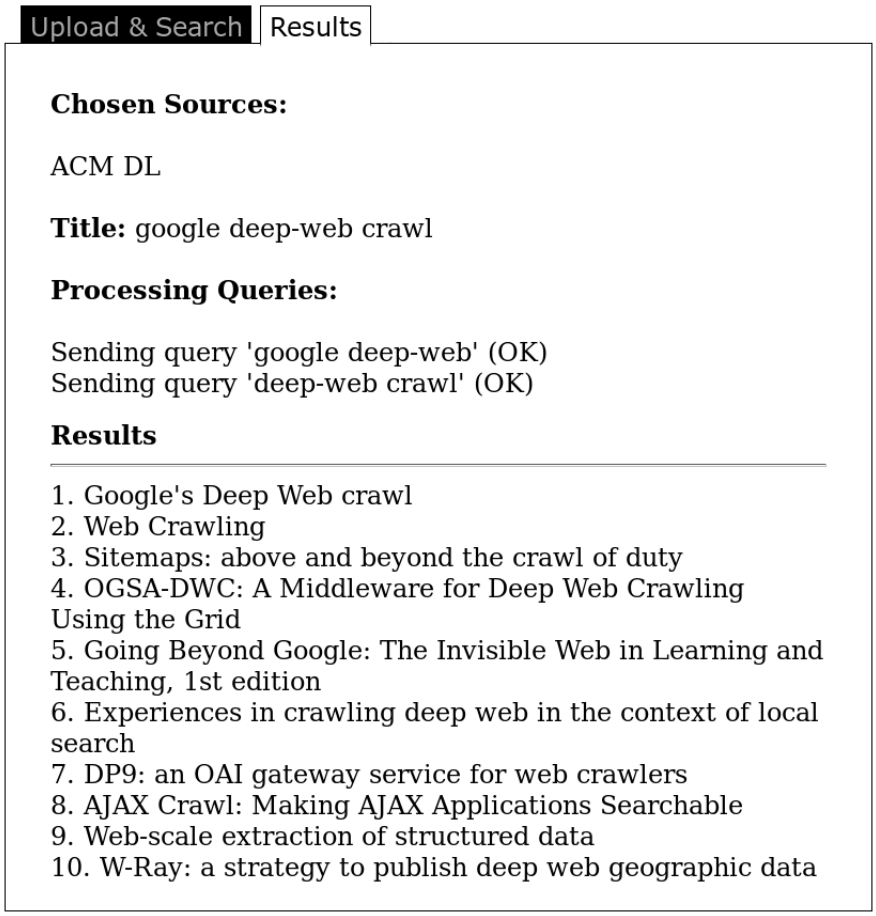


Figura 3.9. Protótipo: lista de resultados

obtidas. Nessa figura, podemos ver quais foram as fontes selecionadas para enviar as consultas, o título do artigo para o qual as recomendações serão feitas e quais consultas estão sendo submetidas (nesse exemplo, 2-gramas do título do artigo). Por fim, são mostrados os 10 primeiros artigos recomendados. O primeiro resultado da recomendação é o próprio artigo alvo encontrado na fonte pesquisada, ele é exibido na lista apenas para fins demonstrativos.

Capítulo 4

Recomendação de Artigos Científicos

4.1 Cenário de Recomendação

Como discutido na Seção 2.5, o problema de recomendação de artigos científicos pode ser visto sob diferentes perspectivas. Para cada uma, pode ser necessário adotar estratégias diferentes. Na definição do problema (Seção 2.5), o critério c é utilizado para definir uma perspectiva particular do problema.

Nesta seção, descrevemos um possível cenário para o problema de recomendação e mostramos como gerar consultas e recomendar artigos dentro desse cenário utilizando o arcabouço proposto. Além disso, através da descrição desse cenário de recomendação e das estratégias propostas, esperamos contribuir com algumas sugestões em como aplicar o arcabouço em outras tarefas de recomendação de artigos científicos.

Especificamente, no caso deste trabalho estamos interessados nas seguintes situações: (1) um revisor recebe um artigo cujo tema ele não é um especialista e precisa se informar rapidamente sobre trabalhos semelhantes para fazer a avaliação; (2) um estudante acaba de entrar para a iniciação científica e recebe um artigo de seu orientador para iniciar seus estudos; (3) um pesquisador, depois de algumas buscas na Web, encontra um artigo interessante e gostaria de encontrar mais artigos como aquele. Nas situações descritas anteriormente, nos parece que o critério mais apropriado para recomendar artigos relacionados seja o de similaridade, isto é, recomendar artigos similares aos que o usuário tem em mãos.

Complementando o nosso cenário de recomendação, levamos em consideração as seguintes restrições : (1) apenas um artigo de entrada que indica as preferências

do usuário está disponível para o sistema de recomendação; (2) todas as informações necessárias para gerar consultas e fazer recomendação só podem ser retiradas do artigo de entrada; (3) somente o título e o resumo dos artigos está disponível publicamente nas fontes de informação. Lidar com essas restrições é fundamental para tornar o arcabouço prático de ser implementado.

Sendo assim, considerando o cenário descrito acima, apresentamos as estratégias de geração de consultas e de ordenação de artigos científicos adotadas na nossa implementação do arcabouço proposto.

4.2 Geração de Consultas

Um dos aspectos mais importantes do arcabouço proposto é a busca de artigos científicos por meio de formulários de consulta disponíveis na Web. Essa etapa é muito importante pois possibilita ao arcabouço a independência de coleções armazenadas a priori. Por outro lado, como não há artigos armazenados previamente, as estratégias de recomendação pouco podem fazer se as consultas geradas não forem capazes de recuperar artigos relevantes. Nesse contexto, assumimos que uma consulta é um conjunto de termos sem qualquer estrutura definida. A seguir, discutimos alguns pontos relevantes do processo de geração de consultas.

O primeiro deles é a escolha da fonte dos termos que serão usados para gerar as consultas. O nosso cenário nos permite utilizar apenas o artigo fornecido pelo usuário para gerar consultas, isto é, utilizar apenas os termos presentes no artigo para gerá-las. Esse fato limita o arcabouço a encontrar apenas os artigos relacionados que compartilham o mesmo vocabulário, não retornando assim outros artigos que adotam vocabulário diferente para abordar o mesmo objeto de estudo.

Em outros trabalhos de sumarização de conteúdo, além dos termos presentes no próprio documento, são utilizadas bases de conhecimento, descritores de categorias e sinônimos para expandir o conjunto de termos associados ao documento. No entanto, como isso demanda armazenamento de informação a priori e é dependente da área de conhecimento, optamos por não adotar esse tipo de estratégia, o que poderia deixar o arcabouço limitado quanto aos possíveis campos de atuação. O estudo da expansão de consultas utilizando termos externos ao artigo de entrada demanda por si só um novo trabalho e, por isso, deixaremos essa tarefa para trabalhos futuros.

Ainda com relação à geração de consultas, mesmo utilizando um único artigo existem diversas opções de onde se podem retirar os termos para as consultas. Artigos científicos têm uma particularidade em relação a outros tipos de documento que é o seu

formato estruturado. Conforme já mencionado, os artigos científicos, são divididos em partes bem definidas, que neste trabalho são chamadas de campos. Como cada campo é responsável por descrever um determinado tópico utilizando termos específicos, isso deveria ser levado em consideração na geração de consultas. No nosso arcabouço, o módulo responsável por essa tarefa recebe como entrada segmentos de texto de cada campo separadamente e escolhe quais deles irá utilizar de acordo com o critério de recomendação estabelecido previamente.

Um outro ponto relevante se refere à ordenação da lista de consultas candidatas. Nessa etapa, as consultas devem ser ordenadas de tal forma que aquelas que tenham maior potencial de recuperar documentos relacionados sejam colocadas no topo da lista. A ordenação é necessária porque a primeira etapa pode gerar uma grande quantidade de consultas candidatas. Como a submissão de consultas é uma tarefa que consome muito tempo e estamos lidando com a criação de um arcabouço que seja útil para ser utilizado no dia-a-dia, limitamos o número de consultas a serem submetidas como sendo as k mais bem posicionadas na lista.

Em vários trabalhos, utiliza-se ganho de informação, frequência na coleção e frequência inversa nos documentos, entre outras métricas globais, para se obter a importância de termos ou ordenar as consultas. De maneira que pudéssemos tornar o arcabouço viável de ser utilizado na prática, focamos somente na informação presente dentro do próprio artigo de entrada para atribuir peso às consultas. Não consideramos nenhum tipo de informação dependente de coleção para gerar estatísticas. A razão para não utilizar informação global é que muitas vezes elas variam em cada área do conhecimento e o arcabouço deveria conhecer e armazenar tal informação a priori. As estatísticas são menos críticas de serem mantidas do que uma coleção de artigos completos e poderiam eventualmente ser incorporadas ao arcabouço, mas ainda assim ele estaria limitado à geração das mesmas previamente.

4.2.1 Estratégias para Geração de Consultas

A estratégia mais simples que poderíamos imaginar seria enviar todo o conteúdo do artigo dado como entrada através do formulário de consulta. Neste caso, em que há uma grande quantidade de termos na consulta, temos o que é chamado na literatura de *consulta longa* [Chen & Zhang, 2009]. No entanto, as máquinas de busca normalmente não trabalham bem com esse tipo de consulta. Elas têm melhor resultado quando se deparam com as chamadas *consultas curtas* que são aquelas consultas com poucos termos (quase sempre abaixo de 10 palavras) que utilizamos no dia-a-dia. Por isso, a estratégia frequentemente adotada é substituir a consulta longa por consultas menores

que capturem as principais ideias do assunto pesquisado.

O grande desafio na geração de consultas é, portanto, descobrir os conceitos mais importantes presentes no texto longo fornecido como entrada. Um conceito pode ser uma palavra, um conjunto de palavras combinadas livremente, uma frase (palavras em sequência), uma expressão e assim por diante [Bendersky & Croft, 2008; Bharat & Broder, 1998]. Em particular, estamos interessados nos termos que melhor representam o artigo científico, ou seja, nos termos que melhor possam sintetizar seu conteúdo. Novamente chamamos a atenção para a escolha do campo de onde as consultas serão extraídas, já que campos como o título e o resumo têm exatamente a função de sumarizar o conteúdo do artigo.

Primeiramente, consideramos consultas com apenas uma palavra, isto é, um conceito descrito por apenas uma palavra. É evidente que ao submeter essa consulta a um formulário obtemos respostas muito genéricas cujo grau de relação com o artigo de entrada é muito baixo, pois uma única palavra dificilmente consegue descrever o artigo e por isso não consideramos consultas desse tipo. Sendo assim, todas as consultas geradas devem conter mais de uma palavra. Consultas com mais de uma palavra são potencialmente mais específicas e assim podem retornar documentos mais úteis, no entanto elas não devem ser específicas ou longas demais para não cairmos no problema das consultas longas.

Uma estratégia simples para gerar consultas com mais de uma palavra seria obter as palavras mais importantes do texto individualmente e combiná-las. Em um primeiro instante isso não nos parece o mais adequado porque estaríamos criando consultas aleatórias que não representam conceitos e que dificilmente um usuário utilizaria. Assim, partindo da observação que temos do dia-a-dia, podemos notar que as consultas que normalmente fazemos são compostas por termos que ocorrem em sequência no texto e que conjuntamente têm uma semântica associada e representam um conceito.

Devido à grande quantidade de alternativas e, principalmente, à dificuldade de analisá-las, decidimos adotar estratégias que já foram utilizadas com sucesso em trabalhos anteriores [Yang et al., 2009; Zhang et al., 2007; Kumar & Srinathan, 2008] e que atendam às restrições definidas no nosso cenário. Assim, consideramos duas estratégias que geram consultas com mais de uma palavra e representativas em relação aos conceitos considerados. A primeira, *n-gramas* é uma estratégia fácil de ser implementada, de baixo custo computacional e que pode ser aplicada a uma grande variedade de artigos. A segunda, denominada de frases substantivas (*noun-phrases*), é uma estratégia mais sofisticada que considera a classe gramatical das palavras para identificar conceitos. Discutimos as duas estratégias mais detalhadamente a seguir.

4.2.1.1 N-gramas

Utilizando-se a estratégia de n-gramas, as consultas candidatas são geradas a partir de subsequências de n palavras extraídas do texto de entrada. Para isso, definimos um tamanho de janela e deslizamos essa janela ao longo do texto extraindo as sequências que se encontram na janela em cada instante. Iniciamos com a janela na primeira palavra do texto e terminamos na última palavra.

Antes de aplicar a estratégia, no entanto, fazemos um pré-processamento do texto. Removemos caracteres especiais (., ;, !, ?, #, etc.) e *stop words*, e colocamos todas as letras em formato minúsculo. Os caracteres especiais e pontuações normalmente não acrescentam nenhum valor adicional a uma consulta e por isso são ignorados. Pelo mesmo motivo são excluídas as *stop words*, palavras que não carregam valor semântico como artigos, preposições, etc. As palavras são colocadas em minúsculo para que se possa agrupar as mesmas consultas escritas de formas diferentes.

Também aplicamos a técnica de *stemming*, na qual se reduz uma palavra ao seu radical. Por exemplo, *living*, *lives* e *lived* seriam transformadas em *liv*. Através dessa técnica é possível agrupar consultas semelhantes em uma única consulta diminuindo a redundância, de modo que seja possível enviar um conjunto de consultas com maior variedade ao formulário de consulta. Isso é bom pois um artigo científico é composto por vários subtópicos e aumentamos a chance de trazer mais artigos relacionados. Utilizamos o algoritmo de Porter [Porter, 1997], que é bastante conhecido e muito utilizado. Através de alguns testes, observamos que as interfaces de busca utilizadas não respondiam bem às consultas que contivessem radicais de palavras, por isso, utilizamos o *stemming* para agrupar as consultas e depois utilizamos a primeira ocorrência da consulta sem *stemming* para ser submetida às interfaces.

Como exemplo, usando uma estratégia de 2-gramas o texto "A Source Independent Framework for Research Paper Recommendation" geraria as seguintes consultas: "source independent", "independent framework", "framework research", "research paper" e "paper recommendation".

4.2.1.2 Frases Substantivas

Part-of-speech tagging (POST) é uma técnica que tem sido utilizada com sucesso em trabalhos anteriores para descobrir os principais conceitos em uma sequência de texto em várias aplicações de recuperação de informação [Bendersky & Croft, 2008; Yang et al., 2009]. Essa técnica consiste em rotular cada palavra do texto de acordo com a sua classe gramatical (*part-of-speech*), isto é, verbo, adjetivo, artigo e assim por diante, por meio de um classificador pré-treinado. Neste trabalho, estamos interessados em

conceitos expressos por subsequências de substantivos. A ideia por trás dessa estratégia é que os substantivos naturalmente representam "coisas" do mundo real, ou seja, eles dão nome a lugares, objetos, seres, entre outros. Por isso, seriam o tipo de objeto gramatical ideal para extrair os conceitos que procuramos no artigo científico. Essa estratégia foi utilizada anteriormente por Yang et al. [2009] em um trabalho similar ao nosso e também em outros trabalhos de geração de consultas [Zhang et al., 2007; Bendersky & Croft, 2008].

Ao contrário da estratégia de n-gramas, não pré-processamos o texto de entrada ao utilizar o POST. O classificador, chamado de POS-tagger, diferencia sentenças com e sem pontuação, com palavras maiúsculas ou minúsculas e não consegue classificar uma palavra somente a partir do seu radical. Por isso, utilizamos o texto da maneira como é extraído do documento de entrada. Após a geração das consultas candidatas, fazemos o *stemming* para agrupar as consultas semelhantes e procedemos da mesma forma como é feito com os n-gramas.

Para dar um exemplo, considere o mesmo texto utilizado anteriormente para os n-gramas: "A source independent framework for research paper recommendation". O classificador POS-tagger poderia rotular cada palavra da seguinte forma "A/DT source/JJ independent/JJ framework/NN for/IN research/NN paper/NN recommendation/NN", em que DT é um artigo, NN um substantivo, IN uma preposição e JJ um adjetivo[1]. Uma frase substantiva pode ser uma sequência de dois ou mais substantivos ou uma sequência de adjetivos seguida por uma sequência de substantivos. No exemplo anterior, as consultas geradas seriam "source independent framework" e "research paper recommendation".

Essa abordagem que utiliza um classificador gramatical limita o arcabouço quanto ao idioma utilizado. Contudo, é razoável assumir que uma parte considerável da pesquisa científica é conduzida em inglês e que os principais artigos são publicados em veículos internacionais que adotam o inglês. Sendo assim, uma razão considerável dos principais artigos científicos está disponível em inglês.

Para aplicar essa estratégia, utilizamos uma implementação baseada em *Conditional Random Fields (CRF)* disponível publicamente na Web[2] [Phan, 2006]. A partir da classificação das palavras basta construir uma máquina de estados para identificar os padrões de frases substantivas.

[1] http://bioie.ldc.upenn.edu/wiki/index.php/POS_tags
[2] http://crftagger.sourceforge.net/

4.2.2 Estratégias para Ordenação de Consultas

Atendendo às restrições do cenário de recomendação, propusemos a utilização de duas estratégias que utilizam somente informação presente no artigo de entrada para dar peso e ordenar as consultas: (1) a frequência dos termos no artigo e (2) a posição em que os termos aparecem no artigo. A seguir discutimos essas duas estratégias.

4.2.2.1 Frequência dos Termos

A intuição por trás deste esquema de atribuição de pesos é que quanto mais vezes uma consulta (isto é, a lista de termos que forma a consulta) aparece no documento de entrada, maior é a probabilidade dessa consulta capturar o assunto principal do artigo, ou seja, o conceito principal. O título e o resumo do artigo normalmente são curtos e não fornecem informação de frequência muito rica. Por outro lado, o corpo do artigo geralmente é longo e pode nos dar informação melhor sobre a frequência das consultas.

Definimos o peso w para uma consulta q com relação à frequência dos termos como

$$w(q) = \gamma \times \frac{frequência(q)}{consulta\_max} + \delta \times \frac{frequência\_palavra(q)}{frequência\_palavra\_max} \tag{4.1}$$

onde $frequência(a)$ é o número de vezes que uma expressão a ocorre em todo o documento, q é uma consulta, $consulta\_max$ é a maior frequência entre todas as consultas, γ atribui a importância da frequência da consulta, $frequência\_palavra$, é o somatório da frequência de cada palavra que compõe a consulta, $frequência\_palavra\_max$ é o maior valor retornado pela função $frequência\_palavra$, δ atribui a importância à frequência individual das palavras que compõe a consulta q. A função $frequência\_palavra$ é definida da seguinte forma:

$$frequência\_palavra(q) = \sum_k \frac{frequência(i_k)}{palavra\_max} \tag{4.2}$$

onde $palavra\_max$ é a maior frequência de uma única palavra em todo o documento e i_k representa a k-ésima palavra na consulta q.

Como uma consulta formada por mais de uma palavra pode não ocorrer com muita frequência, além da frequência da consulta no documento, adicionamos à Equação 4.1 um segundo fator que conta a frequência de cada palavra que compõe a consulta. Dessa forma, criamos um artifício para desempatar consultas com a mesma frequência.

4.2.2.2 Posição dos Termos

Neste esquema de atribuição de pesos, levamos em consideração a posição (campo) do documento de entrada onde os termos que compõem a consulta se encontram. Consideramos que alguns campos são mais importantes que outros e, desse modo, se os termos das consultas aparecem em todos eles, deveriam receber mais peso que os outros (a consulta deveria ter uma posição mais alta na ordenação).

Neste trabalho, lidamos com três campos principais: título, resumo e corpo do artigo. A função do título e do resumo é sintetizar o conteúdo do artigo, por isso, eles são compostos por palavras selecionadas de forma bastante criteriosa. Por esse motivo, acreditamos que esses campos podem conter os melhores termos para compor as consultas e assim os termos presentes aí deveriam receber mais peso.

Utilizamos a Equação 4.1 para estabelecer o peso da consulta e calculamos a frequência da seguinte forma:

$$frequência(i) = \theta \times título(i) + \phi \times resumo(i) + \omega \times corpo(i) \tag{4.3}$$

onde *título*, *resumo* e *corpo* são funções binárias que indicam a presença de uma expressão i nos respectivos campos e θ, ϕ e ω definem a importância de cada função.

4.3 Estratégias para Ordenação de Artigos

Para ordenar os artigos a serem recomendados, utilizamos estratégias de recomendação baseadas em conteúdo. Não utilizamos estratégias baseadas em filtro colaborativo porque o cenário de recomendação descrito na Seção 4.1 não prevê a utilização de perfis de usuário e nem que as referências presentes em um artigo estejam disponíveis publicamente na fonte de informação, o que impossibilita o uso desse tipo de estratégia.

Sendo assim, o critério de similaridade que utilizaremos para recomendar artigos científicos será o de similaridade do conteúdo textual. Para isso, serão adotadas estratégias que se baseiam em uma métrica muito comum e bastante utilizada para este fim, a similaridade do cosseno.

É muito importante que se compreenda que as estratégias de ordenação propostas nesta subseção não representam o produto final do arcabouço. As estratégias são simples, porém são apenas uma parte de todo o sistema de recomendação. Na primeira etapa do processo, a busca por artigos feita através dos formulários de consulta funciona como o principal filtro na recomendação, sendo função das estratégias de ordenação diminuir o esforço de navegação dos usuários ao colocar os artigos mais relevantes no

topo da lista de recomendação. A seguir, apresentamos as três estratégias propostas.

4.3.1 Cosseno Clássico

Na similaridade do cosseno, um artigo é representado como um vetor n-dimensional em que cada posição no vetor representa um termo e o seu valor é a importância do termo no artigo. O maior valor de similaridade ocorre quando os dois artigos são idênticos e ele é igual a 1. A medida do cosseno é definida de acordo com a seguinte equação [Baeza-Yates & Ribeiro-Neto, 2011]:

$$Cosseno\text{-}C(i,j) = \frac{\sum_k w_{ik} w_{jk}}{(\sum_k w_{ik}^2 \times \sum_k w_{jk}^2)^{1/2}} \tag{4.4}$$

onde i e j, no nosso caso, são artigos, i é o artigo de entrada, j um dos artigos recuperados, w_{ik} é o peso ou importância do termo k no artigo i e w_{jk} é o peso ou importância do termo k no artigo j. O peso de um termo é calculado como sua frequência normalizada sem o fator IDF.

Nessa estratégia, ao calcular a similaridade consideramos o título e o resumo do artigo como um texto único, tanto no artigo de entrada quanto nos artigos a serem recomendados. A essa estratégia daremos o nome de Cosseno Clássico (Cosseno-C) por não incluir nenhuma alteração em seu uso tradicional.

4.3.2 Cosseno Linear

Nesta segunda estratégia, levamos em consideração o fato que normalmente os artigos são documentos estruturados. Na Cosseno Clássico, o título e o resumo do artigo são fornecidos como uma única entrada para o cálculo da similaridade. Já no Cosseno Linear, aplicamos a Equação 4.4 em cada campo separadamente e então fazemos uma combinação linear dos valores obtidos para cada um deles. Assim, a nova equação é:

$$Cosseno\text{-}L(i,j) = \alpha \times Cosseno\text{-}C(t_i, t_j) + \beta \times Cosseno\text{-}C(r_i, r_j) \tag{4.5}$$

onde i é o artigo de entrada, j é um artigo recuperado, t_i é o título do artigo i, t_j é o título do artigo j, r_i é o resumo do artigo i, r_j é o resumo do artigo j, α é o parâmetro que configura a importância do título e β é o parâmetro que configura a importância do resumo.

Através dessa estratégia, é possível verificar se algum campo específico tem potencial para melhorar a ordenação final caso receba uma importância maior. Embora

usemos apenas dois campos, essa estratégia poderia ser considerada em cenários com mais informação e com objetivos diferentes.

4.3.3 Cosseno com Pesos Lineares

Robertson et al. [2004] levantam a discussão sobre as desvantagens de se utilizar uma combinação linear de campos para se ordenar documentos estruturados. Ao invés disso, eles propõem uma combinação linear dos campos ao atribuir peso aos termos. Considerando essa nova abordagem, modificamos a maneira de calcular o valor de w na Equação 4.4. A representação do artigo é a mesma da Seção 4.3.1 e o valor de w é calculado conforme proposto por Robertson et al. [2004]:

$$w(i) = \alpha \times frequência(i,t) + \beta \times frequência(i,r) \tag{4.6}$$

onde $frequência(a,b)$ representa a frequência do termo a no campo b, t é o título, r é o resumo, α é um parâmetro que configura a importância do título e β é um parâmetro que configura a importância do resumo.

No trabalho original, os autores aplicam essa estratégia à fórmula do BM25, mas como isso só altera o modo como os pesos são calculados, aplicamos a mesma estratégia à similaridade do cosseno.

Capítulo 5

Experimentos

5.1 Preâmbulo

Através dos experimentos realizados buscamos responder às seguintes questões: (1) quais as melhores combinações de estratégias para geração de consultas; (2) qual é o campo de um artigo científico que fornece os melhores termos para a geração de consultas; (3) como a quantidade de documentos relacionados retornados varia de uma fonte de informação para outra e se a combinação das mesmas melhora os resultados; (4) se é possível obter boas recomendações utilizando estratégias simples baseadas na similaridade do cosseno.

Nesta etapa de avaliação não fizemos nenhuma comparação direta com as abordagens de recomendação de artigos científicos apresentados no Capítulo 1. Acreditamos que um estudo comparativo com tais abordagens não seria o mais apropriado no nosso caso. Neste trabalho optamos por mostrar como construir um arcabouço que lida com restrições presentes no mundo real e propomos estratégias específicas para solucionar as diversas limitações que esse cenário nos impõe. Por outro lado, as demais abordagens não consideram os mesmos tipos de restrição e, a menos que disponham de acesso privilegiado aos campos presentes nas fontes de informação, dificilmente poderiam ser implementadas na prática. O nosso objetivo principal não é apresentar estratégias de recomendação de artigos científicos mais eficazes e sim levantar alguns aspectos relevantes que deveriam ser considerados na implementação de um sistema de recomendação como o que é apresentado aqui. Tendo em vista essa observação, propusemos um *baseline* baseado no comportamento mais esperado de um usuário típico (ver Seção 5.3.1).

Conforme veremos na próxima seção, existem diversas possibilidades de combinação de estratégias para geração de consultas, recomendação de artigos e configuração

de parâmetros. Uma análise detalhada de cada alternativa poderia aumentar consideravelmente a quantidade de artigos a serem avaliados e, assim, tornar os experimentos inviáveis já que o processo de avaliação de cada artigo retornado envolve julgamento de relevância feito por voluntários. Por isso, algumas decisões tiveram que ser tomadas empiricamente e deixamos para trabalhos futuros uma investigação mais profunda sobre outros aspectos e alternativas que não consideramos nestes experimentos.

5.2 Configuração dos Experimentos

Os experimentos envolveram as seguintes três etapas: (1) solicitamos a alguns voluntários que fornecessem um artigo sobre cujo tópico abordado tivessem um bom domínio; (2) para cada artigo fornecido, rodamos uma implementação do nosso arcabouço para buscar e ordenar uma lista de artigos relacionados; (3) solicitamos a cada voluntário que avaliasse a lista de artigos recomendados para o seu próprio artigo. Cada lista continha os artigos retornados por todas as estratégias de geração de consultas e eles foram apresentados ao respectivo voluntário em ordem aleatória. Não mostramos as posições reais em que os artigos foram retornados porque geralmente usuários tendem a confiar nos documentos que aparecem nas primeiras posições e assim poderiam ser induzidos erroneamente a avaliar melhor esses documentos. Neste experimento, 10 voluntários, alunos de pós-graduação em Ciência da Computação, participaram das avaliações.

A Figura 5.1 apresenta uma visão geral de como os experimentos foram projetados. Como podemos ver, utilizamos três campos do artigo científico para extrair os termos das consultas: título, resumo e corpo. Para cada campo, são usadas duas estratégias para geração de consultas, n-gramas e frases substantivas, e para cada uma dessas estratégias existem outras duas para se ordenar as consultas geradas, frequência dos termos e posição dos termos. Combinamos as estratégias para geração e ordenação de consultas, obtendo-se assim um total de quatro estratégias para geração das consultas a serem avaliadas.

Para cada combinação de estratégias, criamos um *pool* com as k melhores consultas. Nestes experimentos, escolhemos arbitrariamente $k = 10$, que a princípio parece um número razoável de consultas para encontrar artigos relacionados. É importante observar que o sistema de recomendação deve respeitar políticas de acesso às fontes de informação, submetendo uma consulta por vez e aguardando um certo intervalo de tempo até que a próxima consulta possa ser submetida. Sem isso, poderíamos sobrecarregar a fonte de informação enviando mais requisições do que ela é capaz de responder e assim ter o acesso a ela revogado. Em contrapartida, o tempo de resposta ao usuário

aumenta muito, o que pode ser um problema. Em estudos futuros seria interessante analisar a qualidade do arcabouço versus o tempo de resposta e, consequentemente, o número de consultas. Observe que neste experimento se torna impraticável ajustar a quantidade de consultas a serem submetidas pois seria necessário avaliar todos os artigos retornados por cada uma delas.

Depois de criar o *pool* de consultas, submetemos cada uma através de formulários

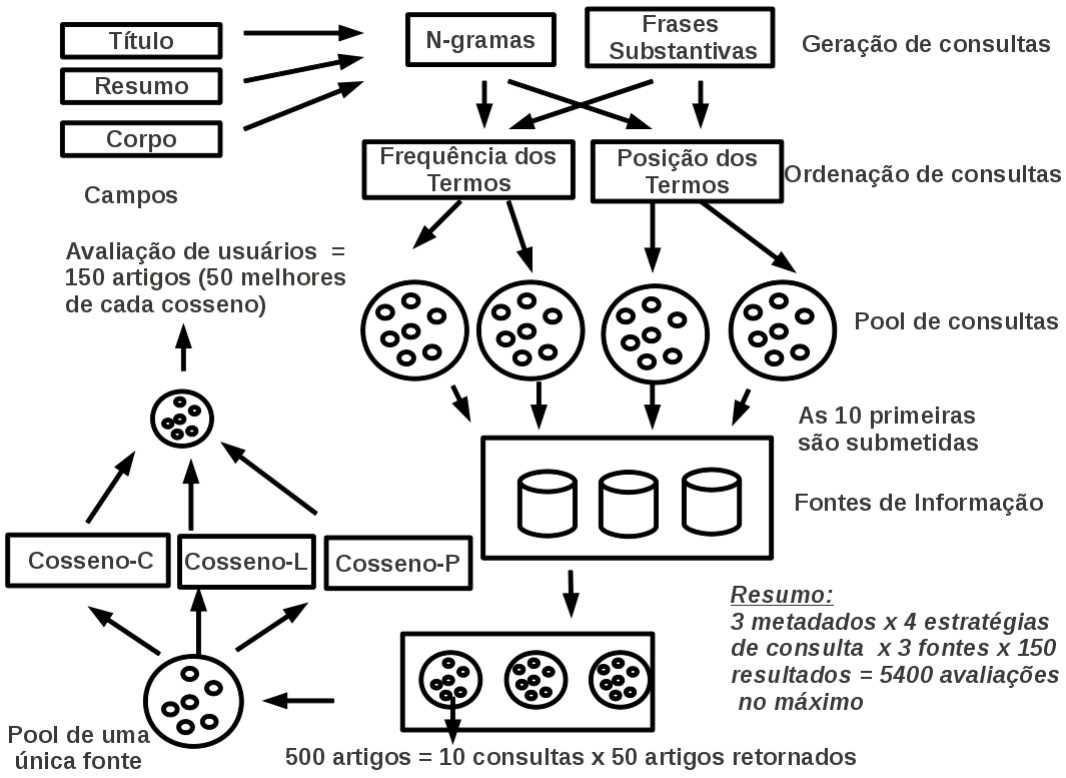


Figura 5.1. Visão geral dos experimentos

disponíveis às três fontes de informação escolhidas para este trabalho: ACM Digital Library, IEEE Xplore, duas bibliotecas digitais bem conhecidas na área de Ciência de Computação, e ScienceDirect, um serviço que provê acesso a artigos de diversas áreas, incluindo Ciência da Computação. A escolha de Ciência da Computação se deve à dificuldade em se obter voluntários de outros áreas. Para cada consulta em uma fonte, extraímos os r primeiros artigos retornados pela fonte. O parâmetro r seria mais um a ser ajustado e que também requer uma investigação futura. Nestes experimentos escolhemos $r = 50$. Uma vez que todas as consultas tenham sido submetidas e os campos dos artigos tenham sido extraídos, criamos três *pools*, cada um com artigos provindos de uma única fonte de informação.

Para cada *pool* de artigos gerado, aplicamos três estratégias de ordenação

baseadas na medida do cosseno (ver Seção 4.2) e, por fim, criamos um novo *pool*, contendo os 50 artigos mais bem posicionados por cada estratégia a fim de reduzir a quantidade de artigos a serem avaliados. No pior caso, o novo *pool* pode conter até 150 artigos. Observe que este é o número de avaliações a serem feitas para testar um único campo com uma única combinação das estratégias de ordenação em apenas uma fonte.

Sumarizando o número de artigos a serem avaliados: são três campos, quatro estratégias de geração de consultas e três fontes de informação. Como no final cada pool pode conter até 150 artigos, cada voluntário poderia ter que avaliar, no pior caso, $3 \times 4 \times 3 \times 150 = 5400$ documentos. Obviamente esse número acabou sendo reduzido quando removemos os artigos replicados e na média cada voluntário avaliou cerca de 1000 artigos, o que ainda é uma quantidade bastante elevada. Cada artigo foi avaliado como muito relacionado, relacionado e não relacionado.

É importante destacar o número de avaliações mencionado no parágrafo anterior para entender o porquê de não considerarmos outras alternativas para geração de consultas e recomendação, ou mesmo variações dentro das próprias estratégias, ao realizar estes experimentos. Em um sistema complexo como o nosso, variar um parâmetro significa aumentar consideravelmente a quantidade de artigos a serem avaliados pelos voluntários. Por isso, sugerimos que em trabalhos futuros seja feita uma análise individual dos pontos mais importantes do arcabouço, como o número de consultas a serem submetidas e o tamanho dos n-gramas, pois assim é possível fazer uma análise mais detalhada com uma quantidade razoável de artigos a serem avaliados.

Nestes experimentos, utilizamos a seguinte configuração de parâmetros: para a primeira estratégia de geração de consultas, consideramos n-gramas de tamanho 2 - que normalmente capturam a maioria dos conceitos presentes no texto; para os os parâmetros da Equação 4.1, utilizamos $\gamma = 2$ e $\delta = 1$, dando mais importância à frequência da consulta do que aos seus componentes; para os parâmetros θ, ϕ e ω da Equação 4.3, utilizamos os valores 3, 2 e 1 respectivamente, assumindo que os termos do título são os mais importantes, seguidos daqueles presentes no resumo e no corpo do artigo. Enfatizamos que a escolha dos valores para esses parâmetros foi baseada no senso comum e que não fizemos nenhum procedimento mais complexo para ajustá-los.

Os resultados que serão apresentados a seguir correspondem às médias obtidas com as avaliações de cada um dos 10 voluntários. Para verificar a significância estatística desses resultados, realizamos um *t-test* pareado com um intervalo de confiança de 90%. Assim, ao relatar situações de empate ou de superioridade de qualquer estratégia consideramos se elas são estatisticamente significantes.

5.3 Resultados

Nesta seção, apresentamos os resultados dos experimentos. Primeiro, analisamos as estratégias de geração de consultas e, depois, as estratégias de ordenação de artigos.

5.3.1 Geração de Consultas

Para avaliar comparativamente nossas estratégias de geração de consultas, propusemos um *baseline* que simula o comportamento de um usuário típico. Quando um usuário está de posse de um artigo que representa seu foco de interesse, uma estratégia trivial que poderia ser adotada para encontrar artigos relacionados em uma fonte de informação seria submeter o título completo do artigo como uma consulta. Outras possíveis soluções seriam: (1) submeter o artigo completo como consulta, mas como já mencionamos na Seção 4.2.1 isso normalmente não funciona bem; (2) após ler o artigo completo, formular as suas próprias consultas, o que também não é uma estratégia apropriada dado o esforço a ser empregado, principalmente se o usuário não conhecer bem o objeto de estudo. Portanto, adotamos a abordagem baseada no título completo como nosso *baseline* para a tarefa de geração de consultas.

A Tabela 5.1 mostra a revocação relativa obtida pelo *baseline* em cada uma das fontes de informação, ACM Digital Library (ACM), IEEE Xplore (IEEE) e ScienceDirect (SD), e também para um *pool* composto por todos os artigos retornados por cada fonte individualmente. A coluna "Relacionado e Muito Relacionado" mostra os valores da revocação considerando os artigos avaliados nessas duas categorias como itens positivos e a coluna "Muito Relacionado" mostra os valores de revocação considerando como itens positivos somente os artigos avaliados como muito relacionados. Utilizamos a Tabela 5.1 na comparação com as nossas estratégias.

Tabela 5.1. Revocação relativa do *baseline*

	Fonte	Relacionado e Muito Relacionado	Muito Relacionado
Baseline	ACM	0,09	0,16
	IEEE	0,02	0,03
	SD	0,03	0,02
	Todas	0,12	0,19

5.3.1.1 Estratégias para Geração de Consultas

As Tabela 5.2 e 5.3 mostram os valores de revocação relativa obtidos pelas estratégias de geração de consulta propostas neste trabalho. A diferença entre essas duas tabelas

está no modo como o cálculo da revocação foi realizado. Na Tabela 5.2, foram considerados itens positivos os artigos avaliados como muito relacionados ou como relacionados, enquanto que na Tabela 5.3 apenas os artigos avaliados como muito relacionados foram considerados como itens positivos. A análise dos resultados será realizada sobre a Tabela 5.2. Quando houver diferença entre a análise nas duas tabelas, isso será feito explicitamente, do contrário pode-se assumir que os mesmos comentários se aplicam a ambas as tabelas.

A primeira coluna da Tabela 5.2 mostra qual campo foi utilizado para criar as consultas: título, resumo ou corpo. Há uma quarta opção que chamamos de *Todos* na qual incluímos todas as consultas geradas para cada campo individualmente em um grande *pool* para representar uma estratégia que utiliza todos os campos para gerar consultas. Através dessa estratégia, tentamos observar se é possível aumentar a quantidade de documentos relevantes retornados simplesmente fazendo a união das consultas derivadas de cada campo. A segunda coluna mostra qual a fonte de informação utilizada na submissão das consultas. Com isso podemos verificar se utilizar mais de uma fonte melhora os resultados. As próximas colunas mostram o valor da revocação obtida em cada estratégia: *ng-freq* é a estratégia de geração de consultas candidatas baseada em n-gramas combinada com o esquema de atribuição de pesos que utiliza a frequência dos termos, *ng-pos* corresponde à estratégia que combina n-gramas com o esquema de atribuição de pesos que considera a posição do termo, *fs-freq* é a estratégia de geração de consultas que combina frases substantivas com a atribuição de pesos que utiliza frequência dos termos e, por fim, *fs-pos* é a estratégia que combina frases substantivas com a atribuição de pesos que considera a posição do termo.

A Tabela 5.2 contém muita informação, sendo difícil fazer uma análise sem considerá-la sob diversos aspectos. Por isso, essa análise se dará sob o foco individual em cada aspecto.

Iniciamos a análise pelas estratégias de geração de consultas, n-gramas e frases substantivas. Quando somente o título é utilizado para gerar as consultas, é possível observar que as estratégias de n-gramas apresentam valores de revocação cerca de três vezes maiores que aqueles apresentados por frases substantivas na ACM Digital Library e quando todas as fontes são utilizadas conjuntamente. Nas demais fontes, as duas estratégias obtêm valores similares de revocação. Quando o resumo ou o corpo do artigo é utilizado para gerar as consultas, apesar das estratégias baseadas em n-gramas apresentarem valores de revocação superiores aos das estratégias baseadas em frases substantivas, isso não representa uma diferença estatisticamente significante. Quando utilizamos as consultas geradas usando-se todos os campos, as estratégias baseadas

Tabela 5.2. Revocação relativa das estratégias de geração de consulta: Muito relacionado e relacionado

	Fontes	Estratégias de geração de consultas			
		ng-freq	ng-pos	fs-freq	fs-pos
Título	ACM	0,12	0,12	0,04	0,04
	IEEE	0,04	0,04	0,03	0,03
	SD	0,04	0,04	0,02	0,02
	Todas	0,16	0,16	0,06	0,06
Resumo	ACM	0,21	0,14	0,18	0,18
	IEEE	0,15	0,10	0,13	0,12
	SD	0,08	0,07	0,06	0,06
	Todas	0,32	0,23	0,30	0,28
Corpo	ACM	0,17	0,19	0,17	0,18
	IEEE	0,12	0,13	0,12	0,13
	SD	0,07	0,09	0,06	0,06
	Todas	0,28	0,28	0,28	0,29
Todos	ACM	0,29	0,22	0,23	0,21
	IEEE	0,19	0,15	0,17	0,15
	SD	0,12	0,11	0,09	0,08
	Todas	0,40	0,32	0,35	0,34
Todas combinações		0,69		0,52	

em n-gramas têm ganho de 26% na ACM Digital Library e nos demais casos há um empate estatístico. Por fim, ao combinar todas as estratégias baseadas em n-gramas, última linha da Tabela 5.2, observamos que elas conseguem recuperar cerca de 70% dos artigos relacionados enquanto que as estratégias baseadas nas frases substantivas recuperam apenas cerca de 50%, o ganho observado é de aproximadamente 33%.

A análise anterior revela que apenas em alguns casos há uma diferença estatisticamente significante entre as estratégias de n-gramas e frases substantivas e que essa diferença sempre se dá em favor das estratégias baseadas em n-gramas. Esse é um resultado bastante surpreendente porque n-gramas constituem uma estratégia muito simples e de baixo custo se comparada a frases substantivas. Além disso, frases substantivas têm a desvantagem de serem dependentes do idioma, pois a classificação gramatical baseia-se no vocabulário e nas estruturas de cada idioma. Embora a maioria dos resultados represente situações de empate, ainda assim, é interessante observar que as estratégias baseadas em n-gramas quase sempre apresentam maior revocação que as estratégias baseadas em frases substantivas. Sendo assim, dado que

os ganhos observados na revocação, ainda que sejam em poucos casos, se dão em favor de n-gramas e que sua implementação é bastante simples, concluímos que essa é a estratégia mais apropriada para a geração de consultas.

Agora comparamos nossa melhor estratégia de geração de consultas, n-gramas, com os resultados obtidos pelo *baseline* (Tabela 5.1, coluna "Relacionado e Muito Relacionado"). Verificamos por meio de testes estatísticos que há um empate entre as duas estratégias quando utilizamos o título para gerar consultas usando-se n-gramas. Para os demais casos, n-gramas obtêm uma grande superioridade sobre o *baseline*, como podemos ver em Resumo-Todas e Baseline-Todas, cujos valores são respectivamente 0,32 e 0,12. Dessa forma, podemos dizer que se o usuário não quiser ter o trabalho de ler o artigo completo e gerar suas próprias consultas, nossas estratégias de geração de consultas representam uma alternativa melhor para recuperar mais artigos relacionados do que utilizar o título do artigo como consulta.

Tabela 5.3. Revocação relativa para as estratégias de geração de consultas: apenas artigos muito relacionados

	Fontes	Estratégias de geração de consultas			
		ng-freq	ng-pos	fs-freq	fs-pos
Título	ACM	0,13	0,13	0,03	0,03
	IEEE	0,04	0,04	0,04	0,04
	SD	0,02	0,02	0,01	0,01
	Todas	0,16	0,16	0,07	0,07
Resumo	ACM	0,29	0,13	0,21	0,19
	IEEE	0,17	0,11	0,14	0,13
	SD	0,04	0,05	0,02	0,02
	Todas	0,43	0,24	0,34	0,31
Corpo	ACM	0,24	0,25	0,22	0,22
	IEEE	0,15	0,17	0,16	0,15
	SD	0,03	0,05	0,03	0,02
	Todas	0,38	0,36	0,37	0,35
Todos	ACM	0,39	0,27	0,25	0,23
	IEEE	0,21	0,18	0,17	0,16
	SD	0,07	0,07	0,04	0,03
	Todas	0,57	0,39	0,43	0,38
Todas combinações		0,69		0,48	

5.3.1.2 Estratégias para Ordenação de Consultas

A seguir, analisamos as estratégias de ordenação de consultas, a primeira considera a frequência e a segunda a posição dos termos no artigo científico. Na Tabela 5.2 contrastamos as colunas *ng-freq* e *ng-pos* correspondentes às estratégias baseadas em n-gramas e *fs-freq* e *fs-pos* correspondentes às estratégias baseadas em frases substantivas.

Utilizando somente o título para gerar as consultas, podemos ver que as estratégias de ordenação de consultas obtêm exatamente o mesmo valor de revocação, tanto para n-gramas quanto para frases substantivas. Isso ocorre porque geramos, em média, menos de 10 consultas utilizando o título, que é um segmento de texto pequeno, e como nestes experimentos utilizamos as 10 melhores consultas para recuperar artigos relevantes, nesse caso não foi necessário fazer uma ordenação para que as consultas fizessem parte do *pool*. Utilizando somente o resumo ou todos os campos conjuntamente na estratégia baseada em n-gramas, observamos um ganho médio de 30% da estratégia baseada em frequência sobre a estratégia baseada em posição dos termos. Esse ganho pode ser explicado da seguinte maneira: na estratégia baseada na posição dos termos todas as consultas derivadas do resumo provavelmente também aparecem no conjunto de consultas derivadas do corpo e apenas algumas poucas correspondem a consultas derivadas do título. Isso faz com que o peso atribuído às consultas não sejam muito distintos uns dos outros, não sendo possível diferenciar bem as boas consultas das consultas ruins. Nos demais casos, em geral também não há diferença significativa entre as estratégias.

Como as estratégias de ordenação de consultas são aplicadas ao mesmo conjunto de consultas e recuperam, em geral, o mesmo número de artigos relevantes, poderíamos imaginar a princípio que elas retornam os mesmos documentos. Se de fato as estratégias retornassem os mesmos artigos, os valores agregados na penúltima linha da Tabela 5.2 deveriam estar próximos ao maior valor possível que é 0,88 (os 0,12 restantes foram obtidos pelo *baseline*, Tabela 5.1). Mas observe que esses valores agregados, 0,40, 0,32, 0,35 e 0,34, na penúltima linha da Tabela 5.2, estão longe do maior valor possível, o que indica que os artigos relevantes estão distribuídos entre as estratégias.

Nas estratégias baseadas em n-gramas, o valor agregado de frequência é 0,40 e o da posição dos termos é 0,32 (penúltima linha da Tabela 5.2). O valor agregado dessas duas estratégias (última linha da tabela) é 0,69, o que mostra que praticamente não existe interseção entre elas. No caso das estratégias baseadas em frases substantivas, o valor agregado de frequência é 0,35 e o da posição dos termos é de 0,34. O valor agregado de ambas na última linha da tabela é de 0,52, o que significa que as estratégias não

retornam tantos artigos diferentes como no caso das estratégias baseadas em n-gramas mas, ainda assim, a união das duas estratégias representa um aumento de quase 50% sobre a revocação obtida individualmente por cada uma delas. Dessa forma, concluímos que em geral as estratégias de ordenação de consultas trazem a mesma quantidade de artigos relevantes mas que esses artigos são diferentes.

5.3.1.3 Fontes de Informação e Campos de um Artigo Científico

Na Tabela 5.2 podemos ver que as diferentes fontes de informação têm desempenho diferente uma da outra com uma tendência clara: a ACM Digital Library é a melhor fonte, seguida por IEEE Xplore e ScienceDirect. De qualquer forma, o resultado mais importante que gostaríamos de destacar sobre as fontes de informação é a quarta linha de cada campo. É possível perceber que a utilização de todas as fontes conjuntamente tem o potencial de recuperar mais documentos relevantes e mais documentos diferentes do que quando se utiliza uma única fonte. O ganho médio é de 42% sobre os maiores valores de revocação da melhor fonte, que é a ACM Digital Library. Como dissemos na Seção 2.4, é bem provável que na busca por trabalhos relacionados um usuário queira obter artigos de fontes de informação diferentes, aumentando assim as chances de receber bons artigos e obter maior diversidade.

O outro objeto de análise é o desempenho relativo de cada campo como gerador de consultas. Se contrastarmos cada linha do grupo de resultados para título, resumo e corpo do artigo, podemos ver que o resumo e o corpo são os campos que geram os maiores valores de revocação. O desempenho ruim do título pode ser explicado pela quantidade de consultas produzidas, menos de 10. Apesar de o corpo do artigo produzir inúmeras consultas, é interessante notar que as estratégias de ordenação não conseguiram selecionar consultas melhores que aquelas produzidas pelo resumo. Por sua vez, os resultados referentes ao resumo confirmaram a hipótese que mesmo sendo um campo não muito extenso ele é capaz de fornecer boas consultas, se comparado aos demais campos, devido à sua elaboração mais cuidadosa, contendo, portanto, palavras mais selecionadas que o restante do artigo. A diferença entre os valores obtidos pelo corpo e pelo resumo não é estatisticamente significante. Dessa forma, concluímos que o resumo é o melhor campo pois obtém resultados comparáveis ao corpo porém a um menor custo.

Por fim, observamos que o uso de todas as consultas geradas por cada um dos três campos é a estratégia que obtém a revocação mais alta. Esse resultado mostra que podemos aumentar a revocação utilizando um número maior de consultas e mesclando consultas derivadas de campos diferentes. Observe que nesse caso são submetidas 10

consultas geradas pelo resumo, 10 pelo corpo e uma quantidade gerada pelo título que totaliza menos de 10 consultas. Isso representa um aumento de mais de 200% no número de consultas enviadas. No entanto, o ganho médio observado na revocação foi de apenas 41% sobre o melhor campo individual, que é o resumo. Esse último fato nos faz concluir que o ganho observado pelo uso de consultas de todos os campos se dá a um custo demasiadamente alto e que o resumo é realmente um bom campo para se extrair consultas.

5.3.2 Ordenação de Artigos

Nesta subseção, focamos a análise nas estratégias de ordenação utilizadas para a recomendação de artigos. O objetivo é verificar se as estratégias são capazes de ordenar os artigos retornados pelas consultas de tal forma que os mais relacionados fiquem nas primeiras posições, seguidos pelos relacionados e então pelos não-relacionados. Para isso, utilizamos como métrica de avaliação o NDCG, que mede a qualidade de uma ordenação com itens avaliados em vários níveis.

Como o resumo do artigo constitui o melhor campo para a geração de consultas, faremos a análise apenas dos resultados que incluam esse campo (ver Tabela 5.4). Os valores obtidos com os demais campos são apresentados nas Tabelas 5.5, 5.6 e 5.7 apenas para fins comparativos. Além disso, para as estratégias que necessitavam de parametrização mostramos os resultados produzidos com o melhor parâmetro de título (α) e resumo (β) que foram variados de 0 a 1 em intervalos de 0,05.

A Tabela 5.4 mostra os valores de NDCG obtidos por cada uma da estratégias de recomendação. A primeira coluna indica a fonte de informação utilizada, a segunda a estratégia de recomendação e as demais os valores de NDCG para cada estratégia de geração de consultas. Utilizamos os mesmos rótulos da tabela da Subseção 5.3.1.1: ng corresponde a n-gramas, fs a frases substantivas, $freq$ a frequência dos termos e pos a estratégia de posição dos termos.

Como podemos ver, estratégias simples baseadas na similaridade do cosseno conseguem colocar artigos relevantes nas primeiras posições. Na maioria dos casos, os valores de NDCG obtidos ficam acima de 0,7, o que é um excelente resultado considerando que utilizamos apenas o título e o resumo para determinar as similaridades.

Na Tabela 5.4 ainda podemos ver que as três estratégias de recomendação, Cosseno Clássico (Cosseno-C), Cosseno Linear (Cosseno-L) e Cosseno de Peso Linear (Cosseno-P), têm valores muito próximos, a diferença é menor que 10% na maioria dos casos, em todos as três fontes utilizadas. A hipótese de que o título e o resumo utilizados separadamente poderiam contribuir para melhorar a ordenação não foi ob-

Tabela 5.4. NDCG das estratégias de ordenação utilizando o resumo para gerar consultas

	Recomendação	Estratégias de Geração de Consultas			
		ng-freq	ng-pos	fs-freq	fs-pos
ACM	Cosseno-C	0,76	0,74	0,72	0,73
	Cosseno-L	0,77	0,74	0,75	0,75
	Cosseno-P	0,81	0,74	0,81	0,81
IEEE	Cosseno-C	0,71	0,63	0,67	0,66
	Cosseno-L	0,73	0,72	0,68	0,67
	Cosseno-P	0,75	0,71	0,69	0,68
SD	Cosseno-C	0,56	0,58	0,54	0,55
	Cosseno-L	0,59	0,55	0,51	0,51
	Cosseno-P	0,60	0,54	0,51	0,51
Todas	Cosseno-C	0,75	0,71	0,71	0,73
	Cosseno-L	0,76	0,72	0,73	0,73
	Cosseno-P	0,79	0,76	0,77	0,76

servada. Por outro lado, os resultados mostram que a estratégia do Cosseno Clássico sem nenhuma parametrização é uma boa opção. A tabela também mostra que os melhores resultados em geral são obtidos na ACM Digital Library (que normalmente retorna mais documentos relevantes que as outras fontes), e depois no IEEE Xplore e ScienceDirect respectivamente.

Tabela 5.5. NDCG das estratégias de ordenação utilizando o título para gerar consultas

	Recomendação	Estratégias de Geração de Consultas			
		ng-freq	ng-pos	fs-freq	fs-pos
ACM	Cosseno-C	0,70	0,70	0,45	0,45
	Cosseno-L	0,71	0,71	0,50	0,50
	Cosseno-P	0,70	0,70	0,50	0,50
IEEE	Cosseno-C	0,64	0,64	0,49	0,49
	Cosseno-L	0,67	0,67	0,46	0,46
	Cosseno-P	0,63	0,63	0,51	0,51
SD	Cosseno-C	0,51	0,51	0,37	0,37
	Cosseno-L	0,52	0,52	0,41	0,41
	Cosseno-P	0,50	0,50	0,41	0,41
Todas	Cosseno-C	0,69	0,69	0,45	0,45
	Cosseno-L	0,70	0,70	0,48	0,48
	Cosseno-P	0,68	0,68	0,47	0,47

Tabela 5.6. NDCG das estratégias de ordenação utilizando o corpo para gerar consultas

	Recomendação	Estratégias de Geração de Consultas			
		ng-freq	ng-pos	fs-freq	fs-pos
ACM	Cosseno-C	0,69	0,72	0,72	0,73
	Cosseno-L	0,77	0,72	0,77	0,76
	Cosseno-P	0,80	0,75	0,81	0,82
IEEE	Cosseno-C	0,69	0,66	0,72	0,70
	Cosseno-L	0,68	0,72	0,71	0,70
	Cosseno-P	0,69	0,72	0,72	0,70
SD	Cosseno-C	0,53	0,58	0,54	0,55
	Cosseno-L	0,61	0,60	0,57	0,55
	Cosseno-P	0,61	0,56	0,57	0,55
Todas	Cosseno-C	0,70	0,74	0,71	0,73
	Cosseno-L	0,74	0,75	0,74	0,74
	Cosseno-P	0,76	0,76	0,77	0,76

Tabela 5.7. NDCG das estratégias de ordenação utilizando todos os campos para gerar consultas

	Recomendação	Estratégias de Geração de Consultas			
		ng-freq	ng-pos	fs-freq	fs-pos
ACM	Cosseno-C	0,76	0,71	0,72	0,72
	Cosseno-L	0,77	0,71	0,75	0,74
	Cosseno-P	0,81	0,72	0,80	0,80
IEEE	Cosseno-C	0,71	0,64	0,66	0,66
	Cosseno-L	0,73	0,70	0,66	0,66
	Cosseno-P	0,73	0,70	0,69	0,67
SD	Cosseno-C	0,56	0,57	0,52	0,53
	Cosseno-L	0,56	0,60	0,55	0,54
	Cosseno-P	0,58	0,54	0,55	0,54
Todas	Cosseno-C	0,76	0,71	0,70	0,71
	Cosseno-L	0,77	0,73	0,74	0,73
	Cosseno-P	0,79	0,73	0,76	0,75

Capítulo 6

Conclusões e Trabalhos Futuros

6.1 Conclusões

Devido ao grande esforço e à grande quantidade de artigos científicos publicados periodicamente, sistemas de recomendação têm sido utilizados para ajudar estudantes e pesquisadores na tarefa de pesquisa bibliográfica. Dado um foco de interesse, tais sistemas sugerem artigos científicos automaticamente, reduzindo a quantidade de trabalho manual necessário para completar essa tarefa. O principal problema com as atuais abordagens de recomendação de artigos científicos [Chandrasekaran et al., 2008; McNee et al., 2002; Bogers & van den Bosch, 2008] é que elas necessitam de condições especiais para funcionarem, como o acesso a uma coleção completa de artigos e o acesso a informação específica como referências bibliográficas, perfis de usuários, entre outras. Em tais condições, as recomendações se restringem a fontes de informação pré-definidas e só podem ser disponibilizadas por aqueles com privilégios de acesso aos artigos.

Para resolver esse problema, propusemos nesta dissertação um arcabouço para busca e recomendação de artigos científicos que não depende do armazenamento de nenhuma coleção à priori, obtendo os artigos candidatos a recomendação através de formulários de consulta disponíveis na Web. Para inferir os interesses de um usuário, o arcabouço utiliza apenas um artigo fornecido como entrada. Assim, mostramos como gerar consultas utilizando somente informações presentes no artigo fornecido pelo usuário e também como fazer recomendação utilizando somente dados disponíveis publicamente nas fontes de informação. Esses aspectos tornam a implementação do arcabouço proposto mais factível e permitem que ele possa ser estendido de acordo seus objetivos específicos.

Avaliamos o arcabouço proposto em relação aos seguintes aspectos: estratégias para geração e ordenação de consultas, campos utilizados para gerar as consultas, fontes

de informação utilizadas para submeter as consultas e estratégias para recomendação dos artigos.

Observamos que uma estratégia simples para geração de consultas como a baseada em n-gramas é tão boa quanto a que utiliza frases substantivas, porém a um menor custo. Para ordenar as consultas as estratégias baseadas na frequência e na posição dos termos são capazes de recuperar a mesma quantidade de artigos relevantes. No entanto, o conjunto de artigos retornados por cada estratégia é bastante diferente um do outro. O resumo do artigo científico, apresentou valores de revocação comparáveis ao do corpo do artigo, sendo esses valores até melhores em alguns casos, com o benefício adicional de gerar consultas a um custo menor. Também observamos que a utilização de consultas geradas considerando-se o título, o resumo e o corpo do artigo conjuntamente pode aumentar a revocação. Os resultados também mostraram que a utilização de mais de uma fonte de informação aumenta a quantidade de artigos relevantes recuperados.

Para recomendar artigos relacionados, testamos três estratégias de recomendação baseadas na similaridade do cosseno: o Cosseno Clássico, o Cosseno Linear, que é o cosseno aplicado ao título e o resumo separadamente e combinados linearmente, e o Cosseno de Peso Linear, que atribui peso aos termos combinando linearmente o peso da posição em que aparecem no texto. Os resultados mostraram que não existe uma diferença significativa entre eles, o que dá uma vantagem ao Cosseno Clássico que não depende de nenhuma parametrização. O valor médio do NDCG foi de 0,7, o que é um resultado muito interessante considerando-se que as estratégias são bastante simples e que utilizam apenas o título e o resumo para determinar a similaridade dos artigos retornados.

Ao propor as estratégias para geração e ordenação de consultas, consideramos um cenário específico para a recomendação de artigos. Nesse cenário, consideramos restrições encontradas no mundo real e definimos o objetivo da recomendação como sendo encontrar artigos similares. A definição precisa do objetivo da recomendação é fundamental para a tomada de decisões uma vez que objetivos diferentes quase sempre implicam em abordagens diferentes. Neste trabalho, mostramos uma possível alternativa de se utilizar o arcabouço proposto para resolver um dos vários problemas relacionados à recomendação de artigos. Por fim, dado o alto custo da avaliação dos experimentos, que requer a participação de voluntários, não foi possível analisar em profundidade determinados aspectos das estratégias propostas e, assim, deixamos para trabalhos futuros a análise individual de cada aspecto bem como a inclusão de outras alternativas.

6.2 Trabalhos Futuros

Após uma avaliação geral do arcabouço proposto centrada em um cenário específico, são necessárias análises individuais e mais aprofundadas de certos aspectos. A seguir apontamos algumas alternativas a serem abordadas em trabalhos futuros.

Neste trabalho, consideramos apenas três campos presentes nos artigos: título, resumo e corpo. Entretanto, também existem outros campos que com certa frequência estão presentes nos artigos como palavras-chave, categorias e seções específicas como introdução e trabalhos relacionados. As duas últimas parecem ser boas fontes de termos para gerar consultas, o problema é que nem sempre elas encontram-se explicitamente separadas do restante do artigo. As citações bibliográficas em si também podem fornecer boas consultas, a questão aqui seria extrair os respectivos componentes como o nome dos autores, o título do artigo e o título do veículo de publicação, o que poderia ser feito utilizando-se técnicas como o ONDUX [Cortez et al., 2010]. Assim, poderíamos expandir a análise de metadados para os casos em que os artigos são providos com informações estruturais mais ricas.

Um segundo aspecto com relação à entrada fornecida pelo usuário é que poderíamos aumentar a quantidade de artigos fornecidos como entrada para inferir os interesses do mesmo. Dessa forma, poderíamos utilizar estratégias de ordenação de consultas baseadas em estatísticas globais, como o IDF. Utilizar vários documentos como entrada levanta várias questões sobre como gerar e selecionar consultas e, por isso, consideramos apenas o caso mais simples nesta etapa inicial.

A geração das consultas é uma das etapas mais importantes deste trabalho e também uma das mais difíceis e, por isso, requer uma caracterização mais detalhada. Os próximos passos seriam analisar: (1) o tamanho dos n-gramas e, consequentemente, o tamanho ideal das consultas; (2) a quantidade de consultas submetidas; (3) a quantidade de artigos recuperados (somente o primeiro, os 10 primeiros e assim por diante).

Utilizar informações retornadas pelas fontes também parece ser uma boa opção. Por exemplo, o total de artigos encontrados pode indicar se uma consulta é muito específica ou muito genérica. Já os termos presentes nos artigos retornados pela consulta poderiam ser utilizados como novas consultas. Além disso, a posição em que o artigo aparece no resultado retornado após a submissão de uma consulta poderia ser utilizado na ordenação final do arcabouço.

Uma vez que lidamos com um cenário de pior caso, em um estudo mais avançado poderíamos considerar cenários em que outros metadados, além do título e resumo, estejam disponíveis publicamente. A presença das referências bibliográficas possibilitaria

a utilização de filtro colaborativo [McNee et al., 2002] para se fazer a recomendação. Poderíamos simplesmente inserir as citações que aparecem nos artigos retornados ou utilizar técnicas mais elaboradas que envolvem a criação de uma rede de citações. No último caso, pode acontecer de as consultas retornarem artigos que ocupem posições distantes na rede e assim não termos uma configuração representativa das ligações.

Por fim, expandir o arcabouço para contemplar diferentes objetivos na recomendação de artigos e extrapolar a pesquisa bibliográfica recomendando outros tipos de item como notícias, artigos da Wikipedia, livros, vídeos, entre outros, seriam caminhos interessantes a serem seguidos.

Referências Bibliográficas

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

Agarwal, N.; Haque, E.; Liu, H. & Parsons, L. (2005). Research paper recommender systems: A subspace clustering approach. In *Proceedings of the 6th International Conference on Web-Age Information Management*, pp. 475–491.

Baeza-Yates, R. & Ribeiro-Neto, B. (2011). *Modern Information Retrieval*. Pearson Education Limited, Harlow, England, 2nd edition.

Beel, J.; Gipp, B.; Shaker, A. & Friedrich, N. (2010). SciPlore Xtract: extracting titles from scientific PDF documents by analyzing style information. In *Proceedings of the 14th European Conference on Research and Advanced Technology for Digital Libraries*, pp. 413–416.

Bendersky, M. & Croft, W. B. (2008). Discovering key concepts in verbose queries. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 491–498.

Bharat, K. & Broder, A. (1998). A technique for measuring the relative size and overlap of public web search engines. In *Proceedings of the 7th International Conference on World Wide Web*, pp. 379–388.

Bogers, T. & van den Bosch, A. (2008). Recommending scientific articles using citeulike. In *Proceedings of the 2008 ACM Conference on Recommender systems*, pp. 287–290.

Burke, R. (2007). Hybrid web recommender systems. In *The Adaptive Web*, Lecture Notes in Computer Science, chapter 12, pp. 377–408. Springer-Verlag, Berlin, Heidelberg.

Chandrasekaran, K.; Gauch, S.; Lakkaraju, P. & Luong, H. P. (2008). Concept-based document recommendations for CiteSeer authors. In *Proceedings of the 5th International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems*, pp. 83–92.

Chen, Y. & Zhang, Y.-Q. (2009). A query substitution-search result refinement approach for long query web searches. In *Proceedings of the 2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology - Volume 01*, pp. 245–251.

Cortez, E.; da Silva, A. S.; Gonçalves, M. A. & de Moura, E. S. (2010). ONDUX: on-demand unsupervised learning for information extraction. In *Proceedings of the 2010 International Conference on Management of Data*, pp. 807–818.

Dalvi, N.; Bohannon, P. & Sha, F. (2009). Robust web extraction: an approach based on a probabilistic tree-edit model. In *Proceedings of the 35th SIGMOD International Conference on Management of Data*, pp. 335–348.

Dasdan, A.; D'Alberto, P.; Kolay, S. & Drome, C. (2009). Automatic retrieval of similar content using search engine query interface. In *Proceeding of the 18th ACM Conference on Information and Knowledge Management*, pp. 701–710.

Giles, C. L.; Bollacker, K. D. & Lawrence, S. (1998). CiteSeer: an automatic citation indexing system. In *Proceedings of the 3rd ACM Conference on Digital Libraries*, pp. 89–98.

Gori, M. & Pucci, A. (2006). Research paper recommender systems: A random-walk based approach. In *Proceedings of the 2006 IEEE/WIC/ACM International Conference on Web Intelligence*, pp. 778–781.

Han, H.; Giles, C. L.; Manavoglu, E.; Zha, H.; Zhang, Z. & Fox, E. A. (2003). Automatic document metadata extraction using support vector machines. In *Proceedings of the 3rd ACM/IEEE-CS Joint Conference on Digital Libraries*, pp. 37–48.

He, Q.; Pei, J.; Kifer, D.; Mitra, P. & Giles, L. (2010). Context-aware citation recommendation. In *Proceedings of the 19th International Conference on World Wide Web*, pp. 421–430.

Huang, S.; Xue, G.-R.; Zhang, B.-Y.; Chen, Z.; Yu, Y. & Ma, W.-Y. (2004). TSSP: A reinforcement algorithm to find related papers. In *Proceedings of the 2004 IEEE/WIC/ACM International Conference on Web Intelligence*, pp. 117–123.

Huang, Z.; Chung, W.; Ong, T.-H. & Chen, H. (2002). A graph-based recommender system for digital library. In *Proceedings of the 2nd ACM/IEEE-CS Joint Conference on Digital Libraries*, pp. 65–73.

Järvelin, K. & Kekäläinen, J. (2002). Cumulated gain-based evaluation of IR techniques. *ACM Trans. Inf. Syst.*, 20(4):422–446.

Kullback, S. & Leibler, R. A. (1951). On information and sufficiency. *The Annals of Mathematical Statistics*, 22(1):79–86.

Kumar, N. & Srinathan, K. (2008). Automatic keyphrase extraction from scientific documents using n-gram filtration technique. In *Proceeding of the 8th ACM Symposium on Document Engineering*, pp. 199–208.

Lage, J. P.; da Silva, A. S.; Golgher, P. B. & Laender, A. H. F. (2004). Automatic generation of agents for collecting hidden web pages for data extraction. *Data Knowl. Eng.*, 49(2):177–196.

Lakkaraju, P.; Gauch, S. & Speretta, M. (2008). Document similarity based on concept tree distance. In *Proceedings of the 19th ACM Conference on Hypertext and Hypermedia*, pp. 127–132.

Liben-Nowell, D. & Kleinberg, J. (2003). The link prediction problem for social networks. In *Proceedings of the 12th International Conference on Information and Knowledge Management*, pp. 556–559.

McNee, S. M.; Albert, I.; Cosley, D.; Gopalkrishnan, P.; Lam, S. K.; Rashid, A. M.; Konstan, J. A. & Riedl, J. (2002). On the recommending of citations for research papers. In *Proceedings of the 2002 ACM Conference on Computer Supported Cooperative Work*, pp. 116–125.

McNee, S. M.; Kapoor, N. & Konstan, J. A. (2006). Don't look stupid: avoiding pitfalls when recommending research papers. In *Proceedings of the 2006 20th Anniversary Conference on Computer Supported Cooperative Work*, pp. 171–180.

Nascimento, C.; Laender, A. H. F.; Silva, A. S. & Gonçalves, M. A. (2011). A source independent framework for research paper recommendation. In *Proceedings of the 11th ACM/IEEE Joint Conference on Digital Libraries*. (a ser publicado).

Peng, F. & McCallum, A. (2006). Information extraction from research papers using conditional random fields. *Inf. Process. Manage.*, 42(4):963–979.

Phan, X.-H. (2006). CRFTagger: CRF English POS Tagger. http://crftagger.sourceforge.net/.

Pohl, S.; Radlinski, F. & Joachims, T. (2007). Recommending related papers based on digital library access records. In *Proceedings of the 7th ACM/IEEE-CS Joint Conference on Digital Libraries*, pp. 417–418.

Porter, M. F. (1997). An algorithm for suffix stripping. In *Readings in information retrieval*, pp. 313–316. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

Pudhiyaveetil, A. K.; Gauch, S.; Luong, H. & Eno, J. (2009). Conceptual recommender system for CiteSeerX. In *Proceedings of the 3rd ACM Conference on Recommender Systems*, pp. 241–244.

Raghavan, S. & Garcia-Molina, H. (2001). Crawling the hidden web. In *Proceedings of the 27th International Conference on Very Large Data Bases*, pp. 129–138.

Resnick, P.; Iacovou, N.; Suchak, M.; Bergstrom, P. & Riedl, J. (1994). GroupLens: an open architecture for collaborative filtering of netnews. In *Proceedings of the 1994 ACM Conference on Computer Supported Cooperative Work*, pp. 175–186.

Robertson, S.; Zaragoza, H. & Taylor, M. (2004). Simple BM25 extension to multiple weighted fields. In *Proceedings of the 13th ACM International Conference on Information and Knowledge Management*, pp. 42–49.

Small, H. (1973). Cocitation in the scientific literature: A new measure of the relationship between two documents. *Journal of the American Society for Information Science*, 24(4):265–269.

Strohman, T.; Croft, W. B. & Jensen, D. (2007). Recommending citations for academic papers. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 705–706.

Sugiyama, K. & Kan, M.-Y. (2010). Scholarly paper recommendation via user's recent research interests. In *Proceedings of the 10th Annual Joint Conference on Digital Libraries*, pp. 29–38.

Takaki, T.; Fujii, A. & Ishikawa, T. (2004). Associative document retrieval by query subtopic analysis and its application to invalidity patent search. In *Proceedings of the 13th ACM International Conference on Information and Knowledge Management*, pp. 399–405.

Takasu, A. (2003). Bibliographic attribute extraction from erroneous references based on a statistical model. In *Proceedings of the 3rd ACM/IEEE-CS Joint Conference on Digital Libraries*, pp. 49–60.

Toda, G. A.; Cortez, E.; da Silva, A. S. & de Moura, E. S. (2010). A probabilistic approach for automatically filling form-based web interfaces. *Proceedings of the VLDB Endowment*, 4(3):151–160.

Torres, R.; McNee, S. M.; Abel, M.; Konstan, J. A. & Riedl, J. (2004). Enhancing digital libraries with TechLens+. In *Proceedings of the 4th ACM/IEEE-CS Joint Conference on Digital Libraries*, pp. 228–236.

Vieira, K.; Barbosa, L.; Freire, J. & Silva, A. (2008). Siphon++: a hidden-web crawler for keyword-based interfaces. In *Proceeding of the 17th ACM Conference on Information and Knowledge Management*, pp. 1361–1362.

Wang, F.; Chen, B. & Miao, Z. (2008). A survey on reviewer assignment problem. In *Proceedings of the 21st International Conference on Industrial, Engineering and other Applications of Applied Intelligent Systems*, pp. 718–727.

Wu, Z.; Raghavan, V.; Qian, H.; Rama, V.; Meng, W.; He, H. & Yu, C. (2003). Towards automatic incorporation of search engines into a large-scale metasearch engine. In *Proceedings of the 2003 IEEE/WIC International Conference on Web Intelligence*, pp. 658–661.

Xue, X. & Croft, W. B. (2009). Automatic query generation for patent search. In *Proceeding of the 18th ACM Conference on Information and Knowledge Management*, pp. 2037–2040.

Yang, Y.; Bansal, N.; Dakka, W.; Ipeirotis, P.; Koudas, N. & Papadias, D. (2009). Query by document. In *Proceedings of the 2nd ACM International Conference on Web Search and Data Mining*, pp. 34–43.

Zhang, W.; Liu, S.; Yu, C.; Sun, C.; Liu, F. & Meng, W. (2007). Recognition and classification of noun phrases in queries for effective retrieval. In *Proceedings of the 16th ACM Conference on Information and Knowledge Management*, pp. 711–720.

Zhang, Y.; Callan, J. & Minka, T. (2002). Novelty and redundancy detection in adaptive filtering. In *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 81–88.