UM MÉTODO DE AGRUPAMENTO HIERÁRQUICO PARA RESOLUÇÃO DE AMBIGÜIDADE ENTRE NOMES DE AUTORES EM CITAÇÕES BIBLIOGRÁFICAS

RICARDO GONÇALVES COTA

UM MÉTODO DE AGRUPAMENTO HIERÁRQUICO PARA RESOLUÇÃO DE AMBIGÜIDADE ENTRE NOMES DE AUTORES EM CITAÇÕES BIBLIOGRÁFICAS

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

Orientador: Alberto Henrique Frade Laender
Co-orientador: Marcos André Gonçalves

Belo Horizonte

Agosto de 2009

Cota, Ricardo Gonçalves
C843m Um método de agrupamento hierárquico para resolução de ambigüidade entre nomes de autores em citações bibliográficas / Ricardo Gonçalves Cota. — Belo Horizonte, 2009
xxii, 50 f. : il. ; 29cm

Dissertação (mestrado) — Universidade Federal de Minas Gerais
Orientador: Alberto Henrique Frade Laender
Co-orientador: Marcos André Gonçalves

1. Bibliotecas digitais - Tese. 2. Referências bibliográficas - Tese. 3. Ambigüidade - Tese.
4. Agrupamento hierárquico. I. Orientador II Título

CDU 519.6*73

A todos aqueles me ajudaram nessa conquista.

Agradecimentos

Ao longo deste caminho, contei com o apoio de diversas pessoas. Agora quero compartilhar esta alegria com todos vocês e prestar meu reconhecimento.

Ao meu orientador, professor Alberto Henrique Frade Laender, pelos ensinamentos e orientação que foram além da minha formação acadêmica.

Ao professor Marcos André Gonçalves, pelo interesse e motivação, ainda quando este trabalho era um projeto da disciplina Bibliotecas Digitais.

Aos meus pais, Iracilda e Nísio, pela minha formação como pessoa e preparação para a vida.

Às minhas irmãs, Fabiana e Cecília por estarem sempre presentes.

A toda minha família, especialmente, à tia Neli e ao tio Rubens, por mais uma vez terem me acolhido.

Aos amigos de longa data, que deram força nos momentos certos.

Aos amigos da ATAN, especialmente Marcelo e Raoni, que muitas vezes foram companheiros durante as disciplinas.

Aos amigos do LDB, pela convivência e sugestões que tornaram este trabalho melhor. Especialmente ao Jean e à Monique, por terem compartilhado seus conhecimentos e participarem deste trabalho.

A Deus por me guiar em vários momentos.

A todos que contribuiram neste trabalho e fazem parte da minha vida, o meu agradecimento.

Abstract

In this dissertation, we propose a heuristic-based hierarchical clustering (HHC) method to deal with the name disambiguation problem in collections of bibliographic citations. The method successively fuses clusters of citations of compatible authors based on several heuristics and similarity measures on the components of the citations (e.g., co-authors' names, title of the work, name of the publication venue). In each phase, the information of fused clusters is aggregated, providing more information for the next round of fusion. Experiments with a dataset taken from the DBLP Computer Science Bibliography collection show gains of up to 12% against a previous method that uses the same pattern matching function but does not consider hierarchical clustering. Experiments also show gains of up to 21% against a supervised baseline, which is based on SVM and 15,5% against an unsupervised one based on K-Means. Both baselines use the same evidence considered by our method as well as privileged information about the correct number of clusters, i.e., both baselines require that the correct number of final clusters be known *a priori*, which is unfeasible for large colections. We also present a new tool which uses the HHC method to deal the specific content from a DL. Finally, we present a case study where the developed tool was used to disambiguate the authors' names in citations extracted from the Brazilian Digital Library of Computing (BDBComp). The quality of the generated group in this study suggests that this tool can be used in digital libraries to help in the task of maintaining consistency of their citations. For example, appearances of an author name can be displayed in a unique format, no matter how they appear in the orginal metadata.

Keywords: hierarchical clustering, name disambiguation, digital libraries, bibliographic citations.

Resumo

Nesta dissertação é proposto um método de agrupamento hierárquico baseado em heurísticas (HHC) para tratar o problema de resolução de ambigüidade entre nomes de autores em citações bibliográficas. O método sucessivamente funde grupos de citações de autores com nomes compatíveis, baseando-se em várias heurísticas que exploram os componentes das citações (nomes de co-autores, título do trabalho e nome de veículo de publicação). Em cada fase do agrupamento, a informação dos grupos fundidos é agregada (todas as palavras que formam os títulos e os nomes dos veículos de publicação das citações contidas nos grupos são agrupadas), fornecendo maior informação para a próxima iteração de fusão. Experimentos realizados com uma coleção de teste extraída da DBLP Computer Science Bibliography mostram ganhos acima de 12% sobre um método anterior que utiliza o mesmo algoritmo de identificação de padrões para resolução de ambigüidade entre nomes de autores. Comparado a uma estratégia supervisionada baseada no classificador SVM, O método HHC a supera em 21%. Em relação a uma estratégia baseada na utilização de um algoritmo de agrupamento não-supervisionado (K-Means), o ganho é de 15,5%. As duas linhas de base utilizam as mesmas evidências consideradas pelo método HHC, além de informação privilegiada sobre o número correto de grupos. Isto é, ambas requerem que o número correto de autores seja conhecido *a priori*, o que é inviável para coleções de citações muito grandes, como acontece em bibliotecas digitais reais. Também é a apresentada uma ferramenta que utiliza o método HHC para tratar o conteúdo específico de uma biblioteca digital. Por fim, utilizando a ferramenta desenvolvida, é realizado um estudo de caso, envolvendo o repositório da BDBComp - Biblioteca Digital Brasileira de Computação. Analisando os resultados desse estudo, podemos concluir que, embora a ferramenta desenvolvida não consiga resolver todos os casos de ambigüidade e até introduza alguns erros, existem ganhos significativos na sua utilização. Bibliotecas digitais podem se beneficiar da ferramenta desenvolvida para recatalogar seu conteúdo, possibilitando, por exemplo, agrupar os resultados de uma busca por nome de autor ou padronizar a exibição do nome de um autor nas suas citações.

Palavras-chave: agrupamento hierárquico, resolução de ambigüidade, bibliotecas digitais, citações bibliográficas.

Lista de Figuras

Lista de Tabelas

Lista de Algoritmos

Sumário

Capítulo 1

Introdução

1.1 Motivação

Citações bibliográficas[1] desempenham um importante papel em muitos sistemas relacionados ao mundo acadêmico. Dentre eles, podemos citar a plataforma de currículos Lattes[2] e as bibliotecas digitais de publicações científicas como a DBLP[3] e a BDBComp[4]. Usuários desses sistemas geralmente utilizam citações bibliográficas para encontrar informação de interesse e pesquisadores as utilizam, por exemplo, para estimar o impacto de um artigo. Além disso, quando o conteúdo de diferentes bibliotecas digitais é integrado, elas servem como identificadores dos documentos associados. Portanto, as citações bibliográficas devem ser mantidas o mais consistentes e atualizadas possível. Isto não é uma tarefa fácil, considerando os vários tipos de problema que podem ocorrer, dentre os quais podemos citar erros na catalogação dos dados, variedade de formatos, nomes de autores ambíguos e abreviação dos nomes de veículos de publicação [Lee et al., 2007].

Com a crescente utilização da Web, bibliotecas digitais bibliográficas vêm se tornando um importante recurso para comunidades acadêmicas, pois fornecem aos seus usuários a possibilidade de buscar por publicações de forma centralizada. Por outro lado, estudos sobre as citações bibliográficas catalogadas em uma biblioteca digital desse tipo são importantes para se obter, por exemplo, a cobertura do conhecimento de uma área, tendências de pesquisa, qualidade e impacto das publicações de uma sub-comunidade específica ou pesquisador, padrões de colaboração em redes sociais,

[1] Entende-se por citação bibliográfica um conjunto de dados bibliográficos relativos a um artigo específico, por exemplo, nomes de autores, título do artigo, veículo e ano de publicação.

[2] http://lattes.cnpq.br

[3] http://dblp.uni-trier.de

[4] http://www.lbd.dcc.ufmg.br/bdbcomp

etc. Esse tipo de análise e informação pode ser utilizado, por exemplo, por agências de fomento a pesquisa para a tomada de decisão na concessão de fundos e bolsas para pesquisadores. Assim, para prover serviços com grande ganho de informação de forma confiável, uma biblioteca digital deve possuir citações bibliográficas de alta qualidade. No entanto, nem sempre isso é garantido, uma vez que na catalogação das citações muitas vezes não se valida os dados a serem inseridos.

As citações bibliográficas catalogadas nas bibliotecas digitais podem ser obtidas de diversas maneiras. Entre elas, podemos citar o auto-arquivamento [Silva et al., 2007], que consiste na submissão de metadados e objetos bibliográficos (por exemplo, artigos, relatórios técnicos, livros, etc.) ao repositório de uma biblioteca digital pelos próprios autores ou por pessoas responsáveis pela catalogação desses objetos. Alternativamente, muitas bibliotecas digitais reúnem dados que são freqüentemente obtidos de diversas fontes, tornando o processo de geração de conteúdo mais dinâmico e totalmente descentralizado. Esta forma de aquisição de conteúdo é proposta, por exemplo, pela Open Archives Initiative (OAI) [Lagoze & de Sompel, 2001]. A abordagem OAI consite em, periodicamente, realizar a colheita (*harvesting*) de dados de diferentes fontes através de um protocolo simples e bem definido, denominado Open Archives Initiative Protocol for Metadata Harvesting[5]. Os dados assim obtidos podem ser processados, integrados aos dados de outras fontes e então carregados em um repositório central. Em ambos os casos, a qualidade das citações bibliográficas resultantes influenciará os serviços disponibilizados pela biblioteca digital.

1.2 Caracterização do Problema

Um dos problemas que ocorrem nas coleções de citações bibliográficas construídas pela agregação de conteúdo de diferentes fontes é a falta de padronização dos dados e a inexistência de identificadores únicos, o que gera vários tipos de ambigüidade. Entre eles, um dos mais comuns é a ambigüidade entre os nomes dos autores. Como resultado disso, baseando-se apenas na similaridade textual entre os nomes dos autores para agrupar as citações de uma biblioteca digital, pode-se equivocadamente associar a um autor as publicações de outro ou dividir o conjunto de publicações de um autor em vários sub-conjuntos.

Formalmente, o problema de resolução de ambigüidade entre nomes de autores no contexto de coleções de citações bibliográficas pode ser dividido em dois sub-problemas [Lee et al., 2005]: citações separadas (*split citation*) e citações agrupadas (*mixed cita-*

[5]http://www.openarchives.org/OAI/openarchivesprotocol.html

tion). O primeiro consiste em identificar citações bibliográficas de um específico autor que estão divididas em várias classes, cada classe associada a uma variação do nome desse autor, como se fossem diferentes pessoas. Isto ocorre principalmente devido às variações que o nome de um autor pode apresentar. Como exemplo de causa dessas variações, podemos citar erros de digitação, abreviações e supressão ou troca de sobrenomes intermediários. Um caso de citações separadas pode ser visto na Figura 1.1, onde é apresentada uma tela com o resultado da consulta "Laender" submetida ao Citeseer[6]. Neste caso, os resultados retornados são todos de autoria de um mesmo pesquisador, Alberto Henrique Frade Laender, porém o seu nome aparece em três formas distintas na lista de citações.

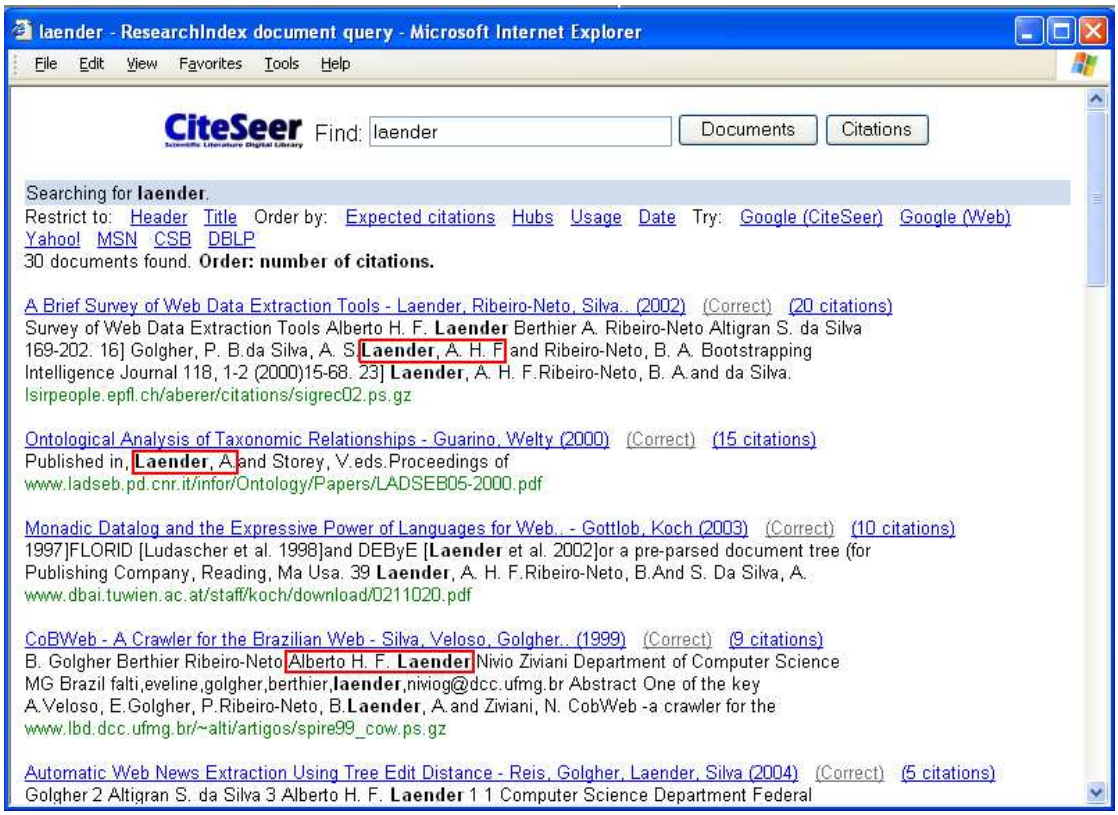

Figura 1.1. Exemplo de citações separadas: um único autor e diferentes grafias.

No segundo sub-problema, diferentes autores apresentam a mesma grafia, principalmente devido à abreviação ou supressão de parte de seus nomes. Por exemplo, o nome A. Gupta pode, em uma publicação, representar Apurba Gupta e, em outra, Anoop Gupta, como na Figura 1.2. Existe ainda o caso de autores que possuem exatamente o mesmo nome, o que torna o problema de resolução de ambigüidade em citações bibliográficas ainda mais complexo.

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

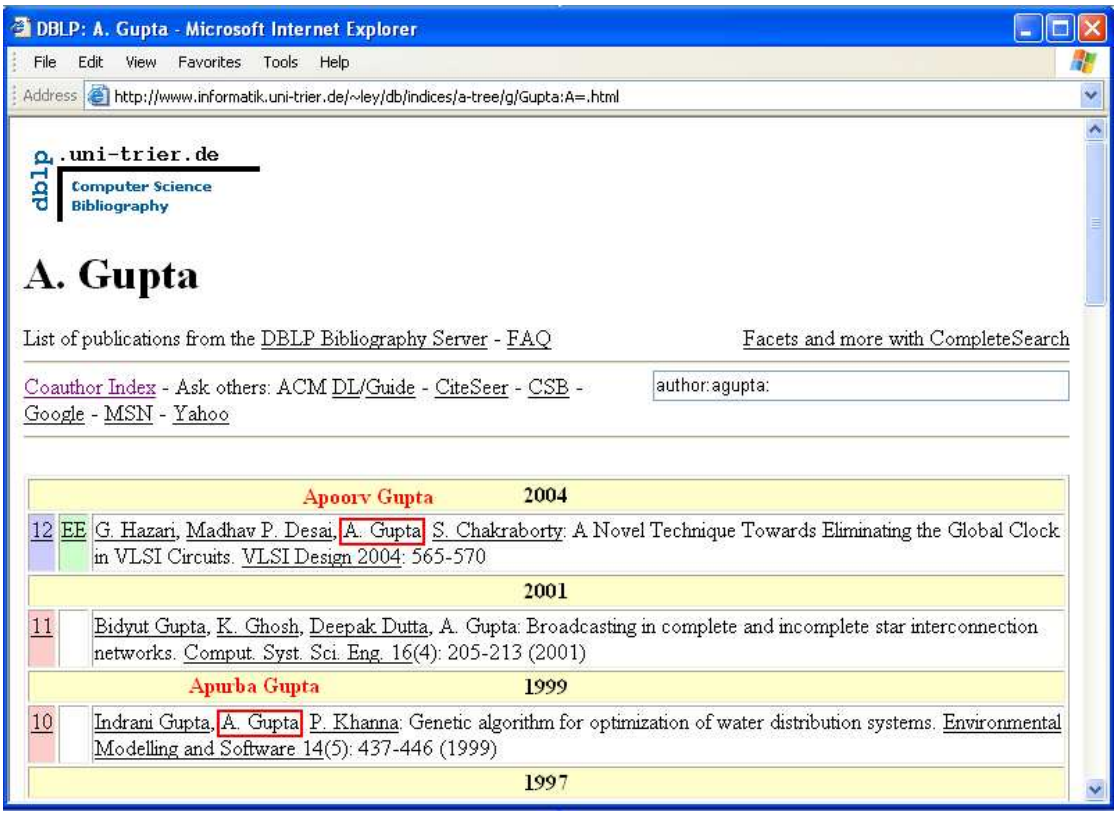

Figura 1.2. Exemplo de citações agrupadas: vários autores e uma grafia.

1.3 Contribuições

Nesta dissertação é proposto um método de agrupamento hierárquico baseado em heurísticas (*Heuristic-based Hierarchical Clustering* - HHC) para tratar o problema de resolução de ambigüidade entre nomes de autores em citações bibliográficas. O método HHC sucessivamente funde grupos de citações de autores com nomes compatíveis baseando-se em várias heurísticas que exploram os componentes das citações (nomes dos co-autores, título do trabalho, nome do veículo de publicação). Em cada fase do agrupamento hierárquico, os dados dos grupos fundidos são agregados (os termos que formam os títulos e os nomes dos veículos de publicação das citações contidas nos grupos são agrupados), fornecendo maior informação para a próxima iteração de fusão. Experimentos feitos com uma coleção de dados extraída da DBLP mostram ganhos de até 21% sobre outros métodos.

Por não encontrar em nossa pesquisa ferramentas que auxiliem os administradores de bibliotecas digitais na resolução de ambigüidade entre nomes de autores e devido aos bons resultados obtidos pelo método HHC, viu-se a oportunidade de desenvolver uma ferramenta para resolução de ambigüidade entre nomes de autores de uma coleção de

citações bibliográficas, correspondente ao conjunto de publicações catalogadas em uma biblioteca digital. A ferramenta possibilita a extração das citações de uma biblioteca digital, utilizando o protocolo OAI-PMH [Open Archives Initiative, 2006] e a resolução de ambigüidade entre nomes dos autores das citações obtidas, com a geração da saída do resultado do processamento gravada em arquivos em um diretório local. Essa organização permite que a ferramenta trabalhe de forma independente da organização interna da coleção das citações.

Utilizando a ferramenta desenvolvida, foi realizado um estudo de caso com o repositório da BDBComp - Biblioteca Digital Brasileira de Computação. Nele foram extraídas 5277 citações, nas quais se verificou a existência de 14856 nomes, não necessariamente de autores distintos, que foram passados como entrada para o método HHC. Ao término da execução, foram definidos 8058 grupos de citações, o que corresponderia ao total de autores no repositório. Também foram avaliados os principais grupos formados e os erros ocorridos. Os resultados obtidos mostraram que, embora a ferramenta não consiga resolver todos os casos de ambigüidade e até introduza alguns erros, agrupando citações de pesquisadores diferentes, existem ganhos significativos na sua utilização.

Assim, considerando o problema de resolução de ambigüidade entre nomes de autores, as contribuições desta dissertação são:

- Proposta de um método de agrupamento hierárquico baseado em heurísticas para resolução de ambigüidade entre nomes de autores em citações bibliográficas com ganhos significativos sobre métodos anteriores [Cota et al., 2007];
- Desenvolvimento de uma ferramenta para resolução desse tipo de ambigüidade quando considerado o conteúdo específico de uma biblioteca digital;
- Estudo de caso, utilizando a ferramenta desenvolvida, envolvendo o repositório da BDBComp.

1.4 Trabalhos Relacionados

O problema de resolução de ambigüidade entre nomes de autores em citações bibliográficas (um caso específico do problema de resolução de entidades) tem sido bastante estudado recentemente [Oliveira, 2005; McRae-Spencer & Shadbolt, 2006; Malin, 2005; Lee et al., 2005; On et al., 2005, 2006; Han et al., 2004, 2005a,b; Huang et al., 2006]. Nesta seção, é apresentada uma revisão desses trabalhos, comparando-os, quando pertinente, ao método HHC proposto nesta dissertação.

Oliveira [2005] propõe um método para resolução de ambigüidade entre nomes de autores e criação de um índice unificado que identifique todas as citações de um autor, independente das variações que seu nome apresenta. Inicialmente, é feita uma pré-avaliação com a aplicação de uma função para casamento de padrão que compara os nomes dos autores contidos no repositório de uma biblioteca digital. Quando esses nomes são considerados candidatos a representar o mesmo autor, a similaridade das fontes adicionais de evidência (nomes dos co-autores, título do trabalho e nome do veículo de publicação) é avaliada para decidir se as duas citações representam trabalhos de um mesmo autor. O método utiliza um algoritmo de casamento de padrão chamado de Comparação por Fragmentos. O algoritmo recebe como entrada duas cadeias de caracteres, correspondentes a dois nomes de autores, representados em uma forma canônica, e compara os fragmentos que compõem esses nomes (incluindo iniciais) utilizando a função de distância de edição de Levenshtein [Levenshtein, 1966].

A seguir, para validar se as citações que possuem nome de autor compatíveis entre si são realmente de uma mesma pessoa, são testadas combinações lógicas baseadas na similaridade das evidências adicionais. Os resultados experimentais mostraram que esse método, embora muito preciso, ou seja, gera grupos de citações de um autor de grande qualidade interna, tende a gerar muitos grupos quando as evidências avaliadas isoladamente entre duas citações são insuficientes para identificar a similaridade entre elas. O método HHC também utiliza o algoritmo de comparação por fragmentos para determinar a similaridade entre nomes de autores, mas se difere por utilizar apenas a evidência de co-autoria para gerar grupos iniciais. Além disso, é introduzida uma etapa de fusão hieráquica onde os termos dos títulos e veículos de publicação das citações de um grupo são agregados e utilizados como critério de fusão desses grupos. Isto explora a intuição que utilizá-los isoladamente para comparar publicações entre si carrega menos informação devido à variedade de termos utilizados.

Uma abordagem em vários passos é adotada em [McRae-Spencer & Shadbolt, 2006]. Cada passo utiliza heurísticas que consideram informações de auto-citação, co-autoria e a URL das publicações. Assim, são exploradas as situações em que um autor tende a referenciar as suas publicações anteriores, possui uma publicação com outro pesquisador que já foi co-autor em uma publicação anterior e possui seus trabalhos agrupados em uma mesma página da Web.

Em [Malin, 2005] são propostos dois métodos para resolver ambigüidade entre nomes de pessoas contidos em diversas fontes. A primeira abordagem é baseada em uma estratégia de agrupamento hieráquico, onde inicialmente cada fonte (em nosso caso uma citação) forma um grupo isolado e em seguida os grupos mais similares são reunidos até que um critério de parada seja atingido ou todas as fontes formem ape-

nas um grupo. Cada fonte é modelada como um vetor booleano, onde cada posição indica a ocorrência/ausência de um nome na fonte. A similaridade entre dois grupos é calculada pelo somatório das similaridades entre as fontes dos grupos, calculadas par a par pela função do cosseno. O método HHC também utiliza uma estratégia de agrupamento hierárquico. Entretanto, para medir a similaridade entre dois grupos, o método avalia as evidências de título e veículo de publicação das respectivas entradas, mas não utiliza informação sobre co-autoria, que é utilizada apenas na etapa preliminar de agrupamento. Ao invés de estimar a similaridade dos grupos pelo somatório das similaridades de suas entradas, o método HHC avalia a semelhança das listas de palavras que ocorrem nas entradas dos grupos. Assim, a sua performance é consideravelmente superior, porque não computa a similaridade entre todas as combinações de pares das entradas dos grupos.

O segundo método proposto em [Malin, 2005] utiliza redes sociais. Nela uma rede é configurada para cada nome ambíguo. Cada nó dessa rede corresponde a um nome diferente e dois nós estão interligados se eles ocorrem simultaneamente em pelo menos uma fonte. As arestas são ponderadas com o número de co-ocorrências nas fontes. Então, são percorridos caminhos aleatórios no grafo resultante. Cada caminho inicia em um nó que representa um nome ambíguo e continua até que outro nó que represente um nome ambíguo seja atingido ou o número máximo de passos seja atingido. A partir dos caminhos encontrados, a probabilidade de se atingir um nó B dado um caminho originado em um nó A é estimada e utilizada para determinar a similiaridade entre A e B. Essa similaridade é utilizada em um processo de agrupamento para remover os nós que tiverem valor abaixo de um limiar. Dessa forma, cada sub-grafo resultante corresponde a uma entidade específica.

Esses dois métodos foram avaliados utilizando uma coleção derivada do Internet Movie Database[7] (IMDb), que possui dados sobre elencos de filmes. De acordo com o autor, o segundo método é mais robusto para a tarefa de resolução de ambigüidade entre nomes, pois considera características referentes a toda a coleção e não apenas a similaridade entre ocorrências.

On et al. [2005] apresentam um estudo comparativo de estratégias para tratar o problema de citações separadas. O estudo é baseado em um arcabouço de dois passos: o primeiro reduz o número de candidatos por meio de blocagem; o segundo avalia a similaridade entre dois nomes utilizando informação de co-autoria. Combinando quatro métodos no primeiro passo com sete métodos no segundo (sendo dois desses métodos supervisionados e os demais, baseados em similaridade textual), os autores apresentam

[7]The Internet Movie Database. http://www.imdb.com

resultados experimentais que permitem definir combinações que são escaláveis e efetivas para se resolver a ambigüidade entre nomes em citações bibliográficas, avaliando a precisão e a escalabilidade dessas combinações. Em contraste com o método proposto nesta dissertação, as estratégias propostas por On et al. [2005] resolvem apenas parte do problema, uma vez que em casos reais os dois sub-problemas (citações separadas e citações agrupadas) ocorrem ao mesmo tempo.

Esses dois sub-problemas são formalmente definidos em [Lee et al., 2005], sendo propostas abordagens para tratamento independente dos mesmos, avaliando a precisão e a escalabilidade das técnicas empregadas. Para o sub-problema de citações separadas são utilizados os métodos apresentados em On et al. [2005] e para o sub-problema de citações agrupadas são propostas duas soluções. Na primeira, as citações são comparadas entre si e, caso a similaridade entre duas citações esteja acima de um determinado limiar, elas são consideradas como sendo de um mesmo autor. A similaridade entre as citações é estimada por uma combinação das similaridades entre seus atributos (nomes dos co-autores, título do trabalho e nome do veículo da publicação). Na segunda solução, para evitar a computação de similaridades entre todas as citações, para cada citação é gerado um conjunto de candidatos de forma eficiente e uma citação é comparada apenas às citações do mesmo conjunto candidato.

Em [On et al., 2006] é proposta uma abordagem que utiliza um grafo de autoria das publicações. Nessa abordagem, cada autor é representado por um nó do grafo e dois nós possuem uma aresta entre eles se os autores correspondentes que os representam possuem co-autoria em pelo menos um trabalho. Em seguida, a estrutura do grafo resultante é utilizada como evidência para agrupar as publicações de um mesmo autor. Segundo os autores, se dois nós representam duas ocorrências do nome de uma mesma entidade, existe uma elevada probabilidade de que eles sejam fortemente conectados entre si através de uma teia de relacionamentos implícita na coleção. Assim, dois nós que possuem poucos nós em comum, mas bastante conectados entre si, podem ser mais similares que dois nós que possuem vários nós em comum, mas pouco conectados entre si. O trabalho considera apenas a ocorrência de citações separadas.

Técnicas de aprendizado de máquina têm sido largamente utilizadas para tratar o problema de ambigüidade entre nomes de autores em citações. Han et al. [2004] propõem duas estratégias supervisionadas de aprendizado de máquina com base na similaridade das evidências de nomes de co-autores, título do trabalho e nome do veículo de publicação. A primeira utiliza o modelo *Naive Bayes*, um modelo estatístico freqüentemente utilizado em tarefas de resolução de ambigüidade em relação ao significado de palavras, para capturar todos os padrões referentes às citações de publicações dos autores. A segunda abordagem é baseada em *Support Vector Machines*

(SVM), que são modelos discriminativos utilizados basicamente como um classificador. Neste caso, cada citação é modelada como um vetor de características, onde as dimensões desse vetor são definidas pelos nomes dos co-autores e palavras-chave contidas no título do trabalho ou no nome do veículo de publicação, sendo o peso dos termos dado pela sua freqüência na citação. Uma diferença básica entre as duas técnicas é que a primeira necessita apenas de exemplos positivos para aprender sobre os padrões das citações referentes às publicações de um autor, enquanto SVM necessita também de exemplos negativos para aprender como classificar essas citações. Em [Han et al., 2005a] é apresentada uma versão hierárquica não supervisionada da primeira estratégia. Uma diferença significativa entre o método HHC e esses trabalhos é que ele não requer treinamento.

Um outro método que utiliza uma técnica de aprendizado não supervisionado, denominada *K-way Spectral Clustering*, é proposta em [Han et al., 2005b]. Como evidências também são utilizados a lista de nomes de co-autores, o título das publicações e o nome do veículo de publicação. Nesse método o parâmetro K, que determina o número de autores presentes na coleção considerada, deve ser definido *a priori*, o que pode ser inviável em casos reais.

A representação vetorial de evidências adicionais também é utilizada para agrupar citações de um mesmo autor em [Huang et al., 2006], mas ao invés de utilizar vetores de características para representar uma citação, os autores utilizam vetores para representar diferenças entre atributos de duas citações. Por exemplo, entre as citações P1 e P2, é criado um vetor para armazenar as similaridades entre os títulos, URLs, nomes de veículos de publicação e afiliação. Para evitar a comparação de todas as publicações entre si, elas são inicialmente divididas em classes candidatas a serem de um mesmo autor. Em seguida, os vetores de similaridade entre as entradas de uma classe são gerados. Esses vetores são passados para um classificador SVM, porém o objetivo não é que ele classifique os vetores como de citações de um mesmo autor (+) ou de autores diferentes (-), mas sim encontrar um nova função de distância entre elas. Nesse caso, a distância é dada pela confiança do vetor ser classificado como (+). Essas distâncias obtidas entre as citações são submetidas ao algoritmo de agrupamento DBSCAN para que ele determine o número e a formação dos grupos. Assim como em [Oliveira, 2005], e também no método HHC, esse método não exige conhecimento anterior sobre o número de grupos gerados.

Uma vez que o problema de resolução de ambigüidade entre nomes não é restrito a um único contexto, é importante citar que outros métodos têm sido descritos na literatura para abordar esse problema. Eles exploram outras fontes de evidência ou são direcionados para outras aplicações (por exemplo, resolução de ambigüidade entre

nomes que aparecem em resultados de uma consulta submetida a máquinas de busca da Web). Song et al. [2007] propõem um método de dois passos não-supervisionado que considera a distribuição de probabilidade de tópicos em relação às pessoas como evidência para a resolução de ambigüidade entre nomes. Vu et al. [2007] propõem o uso de diretórios da Web como base de conhecimento para resolução de ambigüidade entre nomes de pessoas que aparecem em resultados de uma consulta submetida a máquinas de busca da Web. Bekkerman & McCallum [2005] apresentam dois métodos para tratar o mesmo problema, um baseado na estrutura de *links* das páginas da Web, o outro utilizando agrupamento duplo aglomerativo/conglomerativo, que é baseado em um método de agrupamento distributivo de múltiplas formas. Uma discussão mais detalhada desses métodos está fora do escopo desta dissertação, já que eles adotam estratégias e utilizam evidências distintas daquelas do método HHC.

1.5 Organização da Dissertação

Os demais capítulos desta dissertação estão organizados da seguinte forma. O Capítulo 2 define os conceitos utilizados no texto e descreve detalhamente o método HHC. O Capítulo 3 apresenta os resultados obtidos pelo método HHC na remoção de ambigüidade entre nomes de autores de uma coleção de testes, comparando-os aos resultados obtidos por outros métodos utilizados como linhas de base. O Capítulo 4, descreve uma ferramenta que utiliza o método HHC para tratar o conteúdo específico de uma biblioteca digital e apresenta um estudo de caso envolvendo o repositório da BDBComp. Finalmente, o Capítulo 5 apresenta as conclusões e algumas considerações sobre o trabalho e sugere possíveis trabalhos futuros.

Capítulo 2

Método Proposto

O método de resolução de ambigüidade proposto nesta dissertação, denominado Método de Agrupamento Hierárquico Baseado em Heurísticas (*Heuristic-based Hierarchical Clustering* - HHC) [Cota et al., 2007], trata ao mesmo tempo os sub-problemas de citações separadas e citações agrupadas combinando funções de similaridade textual com heurísticas que utilizam evidências presentes nas citações. Ao considerar candidatos compatíveis para geração dos grupos, isto é, para agrupar citações de autores cujos nomes são considerados similares por alguma função de comparação baseada em texto, o método trata o sub-problema de citações separadas (ou seja, reduz as chances de as citações de um autor estarem divididas em diferentes grupos). Por outro lado, ao utilizar heurísticas baseadas na intuição que as citações de um mesmo autor possuem aspectos semelhantes (co-autores em comum, títulos semelhantes, veículos de publicação em comum), o método trata o sub-problema de citações agrupadas. Citações que não possuem características similares não são inseridas no mesmo grupo, diminuindo as chances de que citações de diferentes autores de nomes compatíveis sejam inseridas no mesmo grupo.

Tratar os dois problemas ao mesmo tempo é uma tarefa difícil, uma vez que os objetivos são conflitantes e uma boa solução deve encontrar um balanço entre eles. Para agrupar citações de um mesmo autor que possuem baixa similaridade (sem nomes de co-autores em comum, de temas de pesquisa diferentes e correspondentes a trabalhos publicados em veículos de áreas distintas), é necessário reduzir os limiares de comparação, fazendo com que, eventualmente, citações de autores diferentes sejam inseridas em um mesmo grupo. Por outro lado, se os limiares forem muito estritos, poucas citações seriam agrupadas e os trabalhos de um mesmo autor poderiam estar divididos em vários grupos.

Este capítulo apresenta um descrição detalhada do método HHC. Antes, porém,

na próxima seção são definidos alguns conceitos básicos necessários para descrever o método.

2.1 Conceitos Básicos

Nesta seção são definidos alguns conceitos básicos e estruturas utilizadas ao longo do texto. Inicialmente define-se um *registro de citação* como uma estrutura que contém os metadados relativos à citação de uma publicação específica. Geralmente, as bibliotecas digitais armazenam ou exportam seu conteúdo de acordo com algum formato padrão. Entretanto, os padrões definem apenas a estrutura dos campos dos registros de citação. O conteúdo desses campos é constituído de texto livre e, por isso, pode ocorrer ambigüidade entre os valores de um mesmo campo. A Figura 2.1 mostra um registro de citação extraído da BDBComp por meio do protocolo OAI-PMH, *Open Archives Initiative Protocol for Metadata Harvesting*, utilizando o padrão de metadados Dublin Core[1]. A Figura 2.2 mostra o mesmo registro de citação no formato BIBTEX, usado pelo formatador de texto LATEX [Lamport, 1994].

Um *registro de autoria* possui os metadados que serão utilizados no processo de resolução de ambigüidade entre nomes de autores. Cada registro de autoria representa a participação de um autor na co-autoria de uma publicação, sendo que os nomes dos demais autores são utilizados, juntamente com o título da publicação e o nome do veículo de publicação, como fontes de evidência adicional no processo de resolução de ambigüidade. A partir de um registro de citação com N autores, são gerados N registros de autoria, cada um utilizado para remover a ambigüidade de seu respectivo autor principal. A Figura 2.3 mostra os registros de autoria gerados a partir do registro de citação da Figura 2.1.

Define-se, ainda, um *nome curto* como aquele que possui apenas a inicial do primeiro nome e o último sobrenome completo, como por exemplo, A. Gupta, A. Laender e M. Gonçalves. Os nomes em outros formatos são genericamente denominados de *nomes longos*.

[1] www.dublincore.org

```
<record>
    <header>
        <identifier>sbbd1999meta004</identifier>
        <datestamp>2003-03-22</datestamp>
        <setspec>bdbcomp:sbbd1999</setspec>
    </header>
    <metadata>
        <oai_dc>
            <title>DEByE - uma ferramenta para extracao de
            dados semi-estruturados</title>
            <creator>Alberto Laender</creator>
            <creator>Elaine S. Silva</creator>
            <creator>Altigran S. da Silva</creator>
            <date>1999</date>
            <type>Text</type>
            <identifier>sbbd1999article004</identifier>
            <source>sbbd1999</source>
            <language>por</language>
            <coverage>Florianópolis, SC, Brasil</coverage>
            <rights>Sociedade Brasileira de Computação</rights>
        </oai_dc>
    </metadata>
    <about>
        <provenance>http://www.inf.ufsc.br/sbbd99/</provenance>
    </about>
</record>
```

Figura 2.1. Exemplo de um registro de citação no formato OAI-PMH - Dublin Core

```
@inproceedings{DBLP:conf/sbbd/LaenderSS99,
  author    = {Alberto H. F. Laender and
               Elaine S. Silva and
               Altigran Soares da Silva},
  title     = {DEByE - Uma ferramenta para Extração de Dados Semi-Estruturados},
  booktitle = {SBBD},
  year      = {1999},
  pages     = {155-169},
  crossref  = {DBLP:conf/sbbd/1999},
  bibsource = {DBLP, http://dblp.uni-trier.de}
}
```

Figura 2.2. Exemplo de um registro de citação no formato BIBTEX

Finalmente, define-se um *grupo* como um conjunto de registros de autoria que representam uma lista de citações de um mesmo autor. Neste contexto, um método perfeito para resolução de ambigüidade entre nomes de autores deve gerar, ao final de sua execução, apenas um grupo para cada autor. Em cada grupo deve haver apenas registros de autoria desse autor, representando corretamente a sua lista de citações, por exemplo, relativa às suas publicações catalogadas em uma biblioteca digital.

```
<registro_autoria>
    <autor>Alberto Laender</autor>
    <posicao>1</posicao>
    <co_autores>
        <co_autor>Elaine S. Silva</co_autor>
        <co_autor>Altigran S. da Silva</co_autor>
    </co_autores>
    <titulo>DEByE - uma ferramenta para extracao de dados
    semi-estruturados<titulo>
    <veiculo_publicacao>sbbd1999</veiculo_publicacao>
</registro_autoria>
<registro_autoria>
    <autor>Elaine S. Silva</autor>
    <posicao>2</posicao>
    <co_autores>
        <co_autor>Alberto Laender</co_autor>
        <co_autor>Altigran S. da Silva</co_autor>
    </co_autores>
    <titulo>DEByE - uma ferramenta para extracao de dados
    semi-estruturados<titulo>
    <veiculo_publicacao>sbbd1999</veiculo_publicacao>
</registro_autoria>
<registro_autoria>
    <autor>Altigran S. da Silva</autor>
    <posicao>3</posicao>
    <co_autores>
        <co_autor>Elaine S. Silva</co_autor>
        <co_autor>Alberto Laender</co_autor>
    </co_autores>
    <titulo>DEByE - uma ferramenta para extracao de dados
    semi-estruturados<titulo>
    <veiculo_publicacao>sbbd1999</veiculo_publicacao>
</registro_autoria>
```

Figura 2.3. Exemplo de registros de autoria gerados a partir dos metadados de uma publicação

2.2 Funções de Similaridade e Casamento de Padrão Utilizadas

Nesta seção são apresentadas as duas medidas de similaridade entre cadeias de caracteres e o algoritmo de casamento de padrão, denominado Comparação por Fragmentos, utilizados pelo método HHC.

2.2.1 Medida do Cosseno

A medida do cosseno [Baeza-Yates & Ribeiro-Neto, 1999] calcula a similaridade entre duas cadeias de caracteres considerando um espaço vetorial formado pelos termos que as compõem. Cada cadeia é representada nesse espaço por um vetor de acordo com a ocorrência de seus termos. A similaridade entre duas cadeias de caracteres é dada então pelo cosseno dos vetores que as representam. Nesta dissertação, a medida do cosseno foi utilizada para avaliar a similaridade entre títulos das publicações e nomes de veículos de publicação.

2.2.2 Coeficiente de Jaccard

O coeficiente de Jaccard relativo a duas cadeias de caracteres $A_1 e A_2$ é dado por:

$$J(A_1, A_2) = \frac{|T_{A_1} \cup T_{A_2}|}{|T_{A_1} \cap T_{A_2}|} \tag{2.1}$$

onde T_{A_i} é o multiconjunto de termos da cadeia A_i.

O coeficiente de Jaccard será utilizado adiante para comparar nomes de autores.

2.2.3 Algoritmo de Comparação por Fragmentos

O algoritmo de comparação por fragmentos [Oliveira, 2005] avalia cada fragmento de duas cadeias de caracteres que representam nomes de pessoas. A avaliação dos fragmentos não requer casamento exato, pois a comparação é feita utilizando a função de distância de edição de Levenshtein [1966]. Os parâmetros de entrada são duas cadeias de caracteres normalizadas (c_1, c_2) e um limiar (Lim) utilizado para a distância de edição permitida para considerar fragmentos compatíveis. O algoritmo retorna verdadeiro se as cadeias são compatíveis (podem representar variação de nomes de uma mesma pessoa) e falso caso contrário.

Para que dois nomes de pessoas sejam considerados compatíveis, eles devem possuir em comum, no mínimo, a mesma inicial do primeiro nome e o último sobrenome

(sendo a distância de edição entre esses fragmentos menor que o limiar). Para verificar a primeira condição, uma entre as seguintes quatro condições deve ser satisfeita ($c_i[j]$ representa o j-*ésimo* termo da cadeia de caracteres i):

- Se tanto $c_1[1]$ quanto $c_2[1]$ possuirem mais de um caractere, então a distância de edição entre elas deve ser menor ou igual a Lim;
- Se $c_1[1]$ possuir mais de um caractere e $c_2[1]$ apenas um, então o primeiro caractere de $c_1[1]$ deve ser igual a $c_2[1]$;
- Se $c_2[1]$ possuir mais que um caractere e $c_1[1]$ apenas um, então o primeiro caractere de $c_2[1]$ deve ser igual a $c_1[1]$;
- Se $c_1[1]$ e $c_2[1]$ possuirem apenas um caractere, então ambos devem ser iguais.

Na seqüência, o algoritmo avalia os fragmentos intermediários, que podem aparecer em qualquer ordem e abreviados através de iniciais. Primeiramente, os fragmentos são avaliados por extenso. Caso encontre quaisquer $c_1[i]$ e $c_2[j]$ cuja distância de edição seja menor que o limiar, os dois fragmentos são marcados para que futuras comparações não sejam realizadas. O algoritmo então compara fragmentos por extenso de c_1 com iniciais em c_2 e vice-versa. Por último, iniciais em c_1 são comparadas às iniciais em c_2.

Ao final, caso haja pelo menos um componente em c_1 e em c_2 que não esteja marcado, as cadeias de caracteres são consideradas incompatíveis, caso contrário, são consideradas compatíveis. Uma descrição detalhada do algoritmo pode ser encontrada em [Oliveira, 2005].

2.3 Descrição do Método HHC

O método HHC envolve duas fases principais de processamento: uma para criação dos grupos iniciais de registros de autoria e outra para fusão desses grupos. Antes, porém, há uma fase de pré-processamento onde os registros de autoria são gerados a partir dos registros de citação da coleção considerada. No pré-processamento, o método recebe como entrada a lista de todos os registros de citação da coleção e a partir dela, gera os registros de autoria correspondentes, como ilustrado na Figura 2.5a.

A primeira fase do processamento forma grupos iniciais de registros de autoria a partir da lista de entrada, visando quebrar a complexidade inicial do problema e evitar comparações desnecessárias entre os registros de autoria. A Figura 2.5b ilustra o resultado dessa fase. Repare que o segundo e o terceiro registros de autoria foram inseridos no mesmo grupo porque eles possuem um co-autor de nome compatível (as cadeias

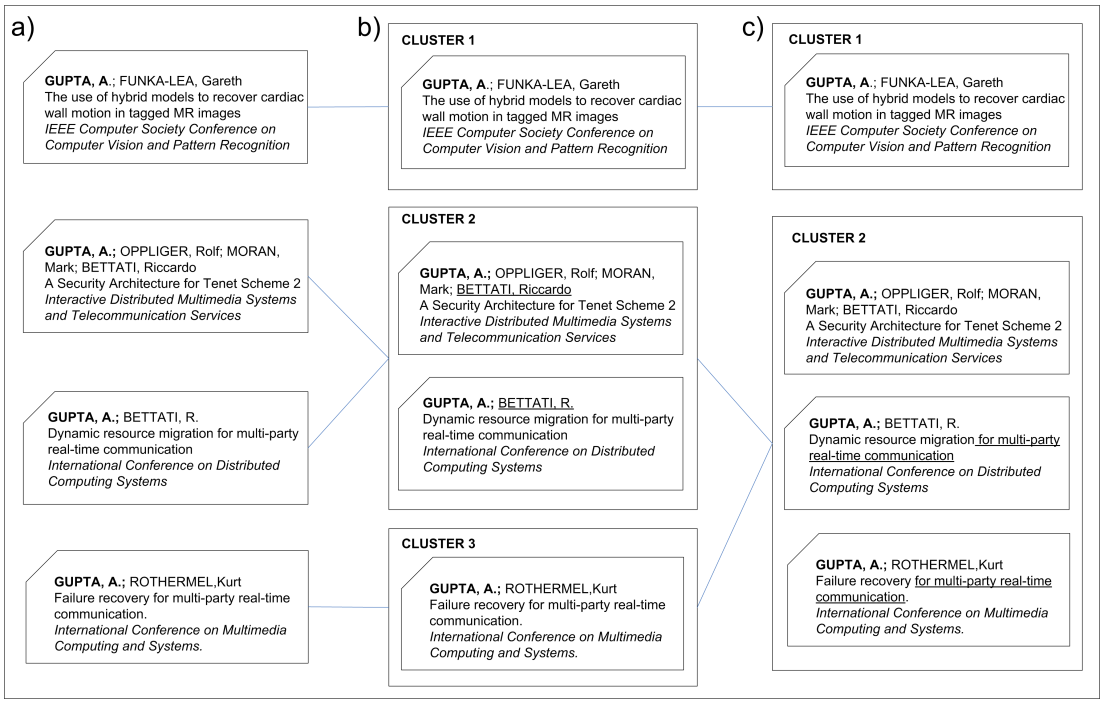

Figura 2.4. Fases do método

Figura 2.5. a) Lista inicial de registros de autoria. b) Grupos gerados na primeira fase do método HHC. c) Resultado final após a fase de fusão.

de caracteres "Riccardo Bettati" e "Bettati, R." são consideradas nomes compatíveis, segundo o algoritmo de comparação por fragmentos). Esta estratégia foi baseada em uma heurística que considera que raramente dois autores com nomes compatíveis que possuem um co-autor em comum seriam pessoas diferentes no mundo real. Como será visto nos resultados experimentais, esta heurística apresenta bons resultados. Este processamento inicial produz grupos de grande qualidade interna, mas a lista de citações de um autor fica bastante fragmentada, ou seja, são gerados vários grupos para um mesmo autor e cada grupo contém parte das citações.

Deve ser ressaltado que a primeira fase do método objetiva reduzir a complexidade do problema inicial, uma vez que separar os registros de autoria em grupos iniciais para em seguida realizar a fusão hieráquica em cima desses grupos reduz o número de comparações possíveis. O problema original possui complexidade quadrática, uma vez que todos registros devem ser comparados entre si. Este processo é conhecido como *blocking*. Nesta dissertação utilizamos o algoritmo de comparação por fragmentos, mas poderia ter sido utilizado qualquer outra estratégia para realizar a blocagem, por exemplo, as propostas de On et al. [2005].

Ao avaliar se existe co-autor em comum entre dois registros de autoria, novamente

nos deparamos com o problema de resolução de ambigüidade entre nome de autores. Nesta dissertação foram experimentadas duas estratégias para determinar se dois nomes de co-autores em diferentes citações bibliográficas correspondem a uma mesma pessoa: uma baseada no coeficiente de Jaccard dos nomes; outra baseada no algoritmo de comparação por fragmentos. Na primeira, se o valor do coeficiente de Jaccard estiver acima de um dado limiar, os dois nomes são considerados como sendo do mesmo autor. Na segunda, o algoritmo de comparação por fragmentos é usado para indicar se há ou não o casamento entre os nomes.

Na segunda fase, o método tenta reduzir a fragmentação dos grupos, utilizando informação adicional dos registros de autoria dos grupos iniciais para fundir grupos de um mesmo autor. Para isso, ele verifica se a similaridade dos termos dos títulos dos trabalhos ou dos termos que formam os nomes dos veículos de publicação está acima do respectivo limiar. Isto é baseado em uma heurística que pressupõe que um pesquisador tende a produzir vários trabalhos em um mesmo tópico e publicações em um mesmo veículo. Cada vez que dois ou mais grupos de autores compatíveis forem fundidos, a informação dos grupos fundidos é agregada (todas as palavras dos títulos são consideradas em conjunto) provendo maior informação para a próxima iteração da fusão. Nesta fase, de maneira hierárquica, os grupos compatíveis existentes são comparados par-a-par. Para determinar se dois grupos devem ser fundidos, o método estima a similaridade deles baseando-se na similaridade de seus campos: título e veículo de publicação. A Figura 2.5c ilustra um exemplo de fusão, onde dois grupos foram fundidos por possuirem vários termos em comum entre os títulos dos seus trabalhos.

Ao final de sua execução o método retorna uma lista G contendo os grupos gerados. A partir dessa lista, uma biblioteca digital pode reconstruir seus registros catalogados, removendo assim, a ambigüidade entre os nomes dos autores. Isto pode ser feito, por exemplo, atribuindo o identificador do grupo de um registro de autoria à ocorrência do seu autor no registro de citação de onde foi gerado.

O Algoritmo 1 apresenta uma descrição detalhada do método HHC. Recebendo como entrada a lista de todos os registros de citação de uma coleção, inicialmente são gerados os registros de autoria, conforme previamente mencionado (linha 5).

Em seguida, a lista de registros de autoria é dividida em duas listas distintas: uma com os registros de autoria cujo nome do autor é curto (linha 6); a outra, com os demais registros (linha 7). Na linha 8, a lista de grupos é criada vazia e na linha 9 a variável usada para definir o identificador de um grupo é iniciada com o valor zero.

A seguir é verificado se a lista de registros de autoria com nomes longos possui pelo menos um registro (linha 11). Caso positivo, o primeiro grupo é criado com o primeiro registro de autoria dessa lista (linhas 12 a 14), sendo os demais registros de

Algoritmo 1 HHC

Entrada: Lista R de registros de citação;
Saida: Lista G de grupos de registros de citação;

```
1: Sejam A, L e C listas de registros de autoria;
2: Seja G uma lista de grupos;
3: Seja I um inteiro;
4:
5: A := CriaRegistrosAutoria(R);
6: C := ObtemRegistrosNomeCurto(A);
7: L := ObtemRegistrosNomeLongo(A);
8: G := "  ";
9: I := 0;
10:
11: se L possui pelo menos 1 registro então
12:    g := CriaNovoGrupo(RemovePrimeiroRegistroAutoria(L), I);
13:    I := I + 1;
14:    Insere (g,G);
15:    ProcessaLista(L,G,I);
16: fim se
17:
18: ProcessaLista(C,G,I);
19:
20: repita
21:    para cada g1 em G faça
22:       para cada g2 em G e g1 ≠ g2 faça
23:
24:          t_t1 := obtemTermosTitulos(g1);
25:          t_t2 := obtemTermosTitulos(g2);
26:          t_v1 := obtemTermosVeiculos(g1);
27:          t_v2 := obtemTermosVeiculos(g2);
28:
29:          se sim(t_t1, t_t2) > l1 ou sim(t_v1,t_v2 ) > l2 então
30:             Funde (g1, g2);
31:          fim se
32:       fim para
33:    fim para
34: enquanto houver grupos fundidos
35:
36: retorna G;
```

L processados conforme explicado adiante (linha 15). Na linha 17, são processados os registros da lista que apresentam autores de nomes curtos. Note que se a lista L estiver vazia, o primeiro grupo será criado no processamento da lista C.

De acordo com o método HHC, o processamento das duas listas é feito da mesma forma. A separação dos registros em duas listas visa apenas utilizar os nomes que carregam maior informação para iniciar a formação dos grupos. Com isso, evitam-se erros logo no início do processamento e a sua conseqüente propagação durante a execução do método.

Algoritmo 2 ProcessaLista

Entrada: Lista R de Registros de Autoria
Entrada: inteiro I
Saida: Lista G de Grupos

```
 1: para cada r em R faça
 2:   para cada g em G faça
 3:     se o nome do autor de r for compatível com o nome autor do grupo g e existe
        co-autor em comum entre r e g então
 4:       insereRegistroAutoria(r,g)
 5:     fim se
 6:   fim para
 7:   se o registro r não foi inserido em algum grupo então
 8:     g1 : = CriaNovoGrupo(r,I);
 9:     I := I + 1;
10:     Insere(G,g1);
11:   fim se
12: fim para
```

O processamento de uma lista de registros de autoria é feito conforme o Algoritmo 2. Para cada registro da lista de entrada, o laço da linha 2 percorre a lista G de grupos e compara o nome do autor do registro com o nome do autor do grupo (linha 3). Se eles forem compatíveis e o registro de autoria e o grupo tiverem, no mínimo, um co-autor de nome compatível, o registro de autoria é inserido no grupo (linha 4); caso contrário, um novo grupo é criado com este registro de autoria (linhas 5 a 8).

Os grupos gerados na primeira fase são diretamente repassados para a segunda fase, os laços das linhas 20 e 21 são responsáveis por fazer a comparação hierárquica dos grupos. Para determinar se dois grupos devem ser fundidos, o método estima a similaridade entre eles baseando-se na similaridade de seus campos: título do trabalho e nome veículo de publicação (linha 28). Se a similaridade de um desses campos estiver acima de um limiar, os dois grupos são fundidos. Este processo continua até que não haja mais grupos a serem fundidos. O resultado final é a lista G de grupos com seus

respectivos registros de autoria.

Capítulo 3

Resultados Experimentais

Este capítulo apresenta os resultados dos experimentos realizados para avaliar a eficácia do método HHC. Antes, porém, são descritas a coleção de teste, as medidas de avaliação e as linhas de base (*baselines*) usadas.

3.1 Coleção de Teste

Para testar o método proposto nesta dissertação, foi utilizada uma coleção de registros de citação obtida do repositório da DBLP. Cada registro dessa coleção contém o título de uma determinada publicação de um autor, a lista de nomes de co-autores e o nome do respectivo veículo de publicação (conferência ou periódico). Essa coleção inclui 4293 registros de 222 autores distintos, uma média de aproximadamente 19 registros por autor. Observa-se ainda a presença de 208 grafias diferentes e a ocorrência de 2270 nomes curtos, o que corresponde a 55% do total de nomes de autores na coleção. A Tabela 3.1 apresenta dados detalhados sobre a coleção. Uma classe de nomes ambíguos é definida como sendo um conjunto de registros onde o nome dos autores são idênticos quando transformados na forma curta.

Antes de aplicarmos o método HHC aos registros de citação, foi realizada a remoção de *stopwords* de todos os títulos e nomes de veículos de publicação. Foram consideradas *stopwords* específicas de uma coleção de citações bibliográficas na área de Ciência da Computação, tais como: *proceedings*, *journal*, *international*, *algorithm*, *problem*, *implementation*, *computer*. Após a remoção das *stopwords*, os termos dos títulos e os nomes dos veículos de publicação foram reduzidos para o seu radical, utilizando o algoritmo de Porter [Porter, 1997].

Tabela 3.1. Detalhes das classes de nomes ambíguos da coleção DBLP.

Classe	*Número de publicações*	*Número de autores*
agupta	576	26
akumar	243	14
cchen	801	61
djohnson	368	15
jmartin	112	16
jrobinson	171	12
jsmith	924	30
ktanaka	280	10
mbrown	153	13
mjones	260	13
mmiller	405	12

3.2 Medidas de Avaliação

Para avaliação dos grupos gerados a partir da aplicação do método HHC, foram utilizadas as medidas PMG (Pureza Média por Grupo), PMA (Pureza Média por Autor) e K (média geométrica entre PMG e PMA), definidas originalmente em [Lapidot, 2002]. A medida PMG avalia a pureza dos grupos gerados automaticamente em relação aos grupos de referência gerados manualmente, ou seja, se esses grupos incluem apenas elementos que estão presentes nos grupos de referência. Assim, quanto mais puros forem os grupos gerados automaticamente, mais próxima de 1 será esta medida, que varia de 0 a 1. A fórmula para o cálculo de PMG é:

$$PMG = \frac{1}{N} \sum_{i=0}^{q} \sum_{i=0}^{R} \frac{{n_{ij}}^2}{n_i}, \tag{3.1}$$

onde: R é o número de grupos de referência (gerados manualmente);

N é o número total de publicações da base de teste;

q é o número de grupos gerados automaticamente;

n_{ij} é o número total de elementos do grupo i gerado automaticamente pertencente ao grupo j gerado manualmente;

n_i é o número total de elementos do grupo i gerado automaticamente.

A medida PMA mede o grau de fragmentação dos grupos gerados automaticamente em relação aos grupos de referência. Seus valores podem variar entre 0 e 1 e, quanto menos fragmentado estiver o resultado final, mais próximo de 1 será o valor de

PMA. A fórmula para o cálculo PMA é:

$$PMA = \frac{1}{N} \sum_{i=0}^{q} \sum_{i=0}^{R} \frac{{n_{ij}}^2}{n_j}, \tag{3.2}$$

onde n_j é o número total de elementos do grupo j gerado manualmente e os demais parâmetros possuem o mesmo significado da fórmula anterior.

A medida K corresponde à média geométrica entre PMG e PMA, sendo expressa por:

$$K = \sqrt{PMA\ X\ PMG} \tag{3.3}$$

O valor de K determina o equilíbrio entre as duas medidas. O melhor caso possível é atingido quando tanto PMG quanto PMA são iguais a 1.

3.3 Linhas de Base

Para avaliação do método HHC, foram utilizados três linhas de base. A primeira é o resultado obtido em [Oliveira, 2005]. As outras duas correspondem aos resultados obtidos com métodos baseados em técnicas de aprendizado de máquina (SVM e K-means). Para uma comparação mais justa, a comparação com os métodos baseados em SVM e K-means é feita apenas na fase de fusão dos grupos. Para utilizar os algoritmos SVM e K-means, os grupos iniciais foram projetados no espaço vetorial como em [Han et al., 2005a]. A evidência de co-autoria não foi projetada no espaço vetorial, pois ela foi considerada na geração dos grupos iniciais. As dimensões desse espaço correspondem apenas aos termos que aparecem em dois campos do registro de autoria, título e veículo de publicação. Se o mesmo termo aparecer em ambos os campos, duas dimensões são criadas. O esquema utilizado para pesar a ocorrência dos termos nos vetores foi o *Term Frequency - Inverse Document Frequency* (TF-IDF) [Baeza-Yates & Ribeiro-Neto, 1999]. Os valores de IDF para termos que ocorrem nos dois campos foram calculados isoladamente para cada um deles. Por exemplo, suponha que em uma coleção de 10 documentos um mesmo termo ocorra 5 vezes no campo título e 2 vezes no campo veículo de publicação. Então os valores de IDF são respectivamente 0,5 e 0,2. A seguir apresentamos uma breve descrição dos métodos baseados em técnicas de aprendizado de máquina considerados como linhas de base.

3.3.1 Supporting Vector Machines (SVM)

Supporting Vector Machines - SVM [Vapnik, 1998] é uma técnica originalmente desenvolvida para problemas de classificação em duas classes. SVM tenta encontrar um hiperplano ótimo que separe os elementos de um conjunto de treinamento em duas classes. Este trabalho utiliza uma versão estendida da técnica de SVM que faz uso da estratégia de uma classe *versus* todas as outras, isto é, elementos de uma classe são classificados como positivos enquanto elementos das demais são classificados como negativos. O problema foi modelado com cada um dos 222 autores sendo considerados como uma classe. Para treinar o classificador SVM foi utilizada uma parte do conjunto de grupos iniciais. Assim, a tarefa do classificador SVM era atribuir corretamente a classe que representava o autor dos registros de autoria dos grupos de teste. Para isso, foi utilizada a versão implementada pelo software LIBSVM [Chang & Lin, 2001].

3.3.2 K-Means

K-means é um algoritmo de agrupamento simples que consiste em escolher aleatoriamente K centros de grupos e, em seguida, cada ponto no espaço vetorial é associado ao grupo que possui a menor distância do seu centro ao ponto. Após todos os pontos serem associados a um grupo, os centros dos grupos são recalculados em função dos pontos que formam cada grupo. A distância dos pontos aos novos centros de cada grupo é recalculada e caso a distância ao centro de um grupo seja menor que a distância ao centro do grupo atual, o ponto troca de grupo. O processo é repetido até que nenhum ponto troque de grupo ou o número de trocas esteja abaixo de um limiar. Foi utilizada a versão implementada pelo software WEKA [Witten & Frank, 2005].

3.4 Experimentos

Devido à forma de construção dos grupos, era esperado que a ordem de processamento da lista de registros pudesse afetar o resultado final. Para avaliar esse impacto, nos resultados desta seção são apresentadas médias de dez execuções de cada experimento. Em cada execução, a lista de registros é ordenada aleatoriamente. Entretanto, o baixo desvio padrão encontrado em todos os experimentos mostra que a ordenação da lista não é um fator significativo no resultado final. Para encontrar os melhores limiares para comparação de títulos e nomes de veículos de publicação foram testados valores na faixa de 0 a 1, utilizando intervalos de tamanho 0,1. Os resultados dos experimentos levam em consideração os valores de 0,3 encontrados para ambos os limiares.

No primeiro experimento, foi avaliada a estratégia de separar a lista de registros de autoria de entrada em uma lista de nomes de autores curtos e outra com os demais nomes na primeira fase do método . A primeira linha da Tabela 3.2 mostra os resultados obtidos com a estratégia de separação da lista de registros de autoria (nomes não curtos primeiro - NNCP); a segunda, mostra os valores obtidos com a execução do método sem a separação da lista (SSL). Além de conseguir um melhor valor para a medida K, os grupos gerados pela primeira estratégia possuem uma grande pureza interna. Isso pode ser explicado pelo fato de que nomes curtos carregam menos informação e utilizar esses nomes no início do processamento pode levar à geração de grupos de baixa qualidade.

Tabela 3.2. Comparação das estratégias NNCP x SSL

	Grupos	*PMG*	*PMA*	*K*
NNCP	462 ± 20	0,857 ± 0,008	0,651 ± 0,020	0,747 ± 0,010
SSL	381 ± 11	0,702 ± 0,015	0,705 ± 0,023	0,703 ± 0,014

O segundo experimento compara o uso do coeficiente de Jaccard (CJ) e do algoritmo de comparação por fragmentos (CF) para determinação da similaridade entre nomes de autores. Para a primeira estratégia, foram testados valores de limiares variando de 0 a 1 em intervalos de 0,1. O melhor valor da medida K foi encontrado utilizando o limiar de 0,5 (primeira linha da Tabela 3.3). Avaliando os resultados da Tabela 3.3, pode ser visto que a diferença entre as estratégias não é significativa. Contudo, o maior valor de PMG indica que é recomendado utilizar o algoritmo de comparação por fragmentos, uma vez que, embora o número de grupos seja maior, o objetivo da primeira fase é gerar grupos de grande qualidade interna. Por outro lado, o elevado valor de PMG nas duas estratégias mostra que a heurística utilizada para comparar nomes de co-autores apresenta bom resultado, ocorrendo poucos casos em que co-autores com nomes compatíveis correspondem a diferentes pessoas no mundo real.

Tabela 3.3. Comparação de estratégias para comparação de nomes de co-autores

	Grupos	*PMG*	*PMA*	*K*
CJ	410 ± 6	0,829 ± 0,005	0,679 ± 0,020	0,750 ± 0,013
CF	462 ± 20	0,857 ± 0,008	0,651 ± 0,020	0,747 ± 0,010

A Tabela 3.4 mostra a comparação dos resultados obtidos pelo método HHC e pelos melhores valores obtidos por Oliveira (2005). Neste caso, utilizamos a comparação direta por dispormos da mesma coleção de teste e utilizarmos as mesmas medidas de

avaliação. Pode ser visto que nosso método apresenta um valor 12% superior para a medida K. A nova forma de combinar as evidências (utilizando dados dos grupos ao invés de comparação par-a-par dos registros de autoria) e a fase de fusão de grupos reduzem o efeito de fragmentação do método proposto por Oliveira (2005) sem acarretar em grande perda de qualidade interna dos grupos.

Tabela 3.4. Comparação de estratégias

	PMG	*PMA*	*K*
Oliveira [2005]	0,8803	0,5059	0,6673
HHC	0,8574	0,6517	0,7475

Tabela 3.5. Comparação por classe de nome ambíguo da coleção DBLP (Medida K).

Classe	*SVM*	*Kmeans*	*HHC*
agupta	0,6407 ± 0,05	0,6552 ± 0,04	0,6558 ± 0,04
akumar	0,1848 ± 0,06	0,2215 ± 0,07	0,6486 ± 0,05
cchen	0,1430 ± 0,03	0,2075 ± 0,03	0,4693 ± 0,03
djohnson	0,6181 ± 0,04	0,5774 ± 0,05	0,6760 ± 0,07
jmartin	0,5686 ± 0,08	0,7342 ± 0,04	0,8769 ± 0,04
jrobinson	0,5388 ± 0,05	0,5971 ± 0,04	0,6760 ± 0,03
jsmith	0,1431 ± 0,07	0,1641 ± 0,08	0,7020 ± 0,04
ktanaka	0,6858 ± 0,07	0,6490 ± 0,04	0,7391 ± 0,03
mbrown	0,6233 ± 0,07	0,6645 ± 0,05	0,8238 ± 0,02
mjones	0,6136 ± 0,05	0,6127 ± 0,05	0,7553 ± 0,04
mmiller	0,7686 ± 0,10	0,7259 ± 0,01	0,8198 ± 0,09

O último experimento visa comparar a qualidade da fase de agrupamento hierárquico com outras estratégias, uma que utiliza o método baseado em *K-means* e a outra que utiliza o método baseado em SVM. Para avaliar melhor a capacidade de resolução de ambigüidade entre nomes, a coleção foi dividida em classes de nomes de autores e cada classe foi avaliada isoladamente. Como o método baseado em SVM requer treinamento, os grupos iniciais gerados na primeira etapa foram repartidos aleatoriamente, com 50% dos grupos sendo utilizados para treino do classificador. Para que os resultados sejam diretamente comparáveis, cada experimento levou em consideração apenas os grupos utilizados para testar o classificador. Como pode ser visto pelos resultados apresentados nas Tabelas 3.5 e 3.6, na média, o método HHC apresenta resultado 15,5% superior ao K-means para a medida K e 21% superior ao SVM. Além disso, os grupos

gerados pelo método HHC possuem qualidade interna bastante superior às linhas de base.

Tabela 3.6. Média das medidas PMG e PMA

	PMG	*PMA*
HHC	0,6519 ± 0,27	0,5820 ± 0,22
SVM	0,3821 ± 0,19	0,6840 ± 0,28
K-means	0,5180 ± 0,22	0,5467 ± 0,20

3.5 Exemplos de Falhas

```
<registro_autoria>
    <autor>Chuen Liang Chen</autor>
    <posicao>1</posicao>
    <co_autores>
        <co_autor>Bing Feng Wang</co_autor>
        <co_autor>Gen Huey Chen</co_autor>
    </co_autores>
    <titulo>Simple Approach Implementing Multiplication
    Small Tables.<titulo>
    <veiculo_publicacao>Inf. Process. Lett.</veiculo_publicacao>
</registro_autoria>
<registro_autoria>
    <autor>C. Chen</autor>
    <posicao>1</posicao>
    <co_autores>
        <co_autor>Yang Xiao</co_autor>
        <co_autor>B. Wang</co_autor>
    </co_autores>
    <titulo>Bandwidth Degradation QoS Adaptive Multimedia
    Wireless Mobile Networks.<titulo>
    <veiculo_publicacao>Computer Communications Elsevier
    Science</veiculo_publicacao>
</registro_autoria>
```

Figura 3.1. Registros de autoria do primeiro exemplo de falha do método HHC

Nesta seção são apresentados alguns exemplos onde o método HHC falha. O primeiro exemplo é um caso de falha na fase de geração de grupos iniciais. Dois autores de uma mesma classe (*cchen*) são considerados como sendo um mesmo autor. O primeiro registro da Figura 3.1, possui o co-author Bing Feng Wang, enquanto no

segundo aparece o co-autor B. Wang. Ao utilizar o algoritmo de comparação por fragmentos para comparação de nomes de co-autores, os dois registros de citação tiveram um co-autor em comum e, conseqüentemente, foram atribuídos a um mesmo autor.

```
<registro_autoria>
    <autor>A. Gupta</autor>
    <posicao>1</posicao>
    <co_autores>
        <co_autor>Y. Yang</co_autor>
        <co_autor>A. Bagchi</co_autor>
        <co_autor>A. Ray</co_autor>
    </co_autores>
    <titulo>Declarative Specification Gene Regulatory
    Networks Query Evaluation Pathsys.<titulo>
    <veiculo_publicacao>International Conference
    Data Engineering</veiculo_publicacao>
</registro_autoria>
<registro_autoria>
    <autor>A. Gupta</autor>
    <posicao>1</posicao>
    <co_autores>
        <co_autor>Manish Bhide</co_autor>
        <co_autor>Mukesh Mohania</co_autor>
    </co_autores>
    <titulo>Towards Bringing Database Management Task
    Realm It Non Experts.<titulo>
    <veiculo_publicacao>International Conference Data
    Engineering</veiculo_publicacao>
</registro_autoria>
```

Figura 3.2. Registros de autoria do segundo exemplo de falha do método HHC

O exemplo da Figura 3.2 apresenta dois registros de grupos que foram erroneamente fundidos na classe de autores *agupta*. Os dois grupos foram fundidos, pois ambos possuiam registros de autoria relacionados a publicações na International Conference on Data Engineering (ICDE). Porém, o primeiro grupo era relativo ao autor Amarnath Gupta, enquanto o segundo, era relativo a Ajay Gupta (nos registros de autoria os nomes apareceram na forma abreviada A. Gupta).

Em relação ao título das publicações, os registros de autoria da Figura 3.3 do autor Ajay Gupta foram agrupados com os registros do autor Arobinda Gupta (cujo nome também aparecia na forma A. Gupta), pois ambos publicaram trabalhos na área de balanceamento de carga em sistemas distribuidos, conforme determinado pelos títulos de suas publicações.

```
<registro_autoria>
    <autor>A. Gupta</autor>
    <posicao>1</posicao>
    <co_autores>
        <co_autor>Chris Nolt</co_autor>
        <co_autor>Jaswinder Pal Singh</co_autor>
        <co_autor>John L Hennessy</co_autor>
        <co_autor>Takashi Totsuka</co_autor>
    </co_autores>
    <titulo>Load Balancing Locality Adaptive Hierarchical
    N Body Methods Barnes Hut Fast Multipole Rasiosity.<titulo>
    <veiculo_publicacao>J Parallel Distrib Comput</veiculo_publicacao>
</registro_autoria>
<registro_autoria>
    <autor>Ajay Gupta</autor>
    <posicao>1</posicao>
    <co_autores>
        <co_autor>Elise De Doncker</co_autor>
        <co_autor>Patricia Ealy</co_autor>
    </co_autores>
    <titulo>Two Methods Load Balanced Distributed Adaptive
    Integration.<titulo>
    <veiculo_publicacao>European International Conference High
    Performance Computing And Networking Europe</veiculo_publicacao>
</registro_autoria>
```

Figura 3.3. Registros de autoria do terceiro exemplo de falha do método HHC

Por último, a Figura 3.4 mostra dois registros de autoria de um mesmo autor (Amar Gupta, professor do Eller College of Management, University of Arizona) que ficaram em grupos separados porque o primeiro continha registros de autoria de publicações na área de integração de banco de dados enquanto o outro continha registros relacionados a publicações na área de reconhecimento de caracteres.

```
<registro_autoria>
    <autor>Amar Gupta</autor>
    <posicao>1</posicao>
    <co_autores>
        <co_autor>B. E. Prasad</co_autor>
        <co_autor>M. P. Reddy</co_autor>
        <co_autor>P. G. Reddy</co_autor>
    </co_autores>
    <titulo>methodology Integration Heterogeneous Databases.<titulo>
    <veiculo_publicacao>IEEE Trans. Knowl. Data. Eng.</veiculo_publicacao>
</registro_autoria>
<registro_autoria>
    <autor>A. Gupta</autor>
    <posicao>1</posicao>
    <co_autores>
        <co_autor>A. Liu</co_autor>
        <co_autor>M. V. Nagendraprasad</co_autor>
        <co_autor>Patrick Shen Pei Wang</co_autor>
        <co_autor>S. Ayyadurai</co_autor>
    </co_autores>
    <titulo>Integrated Architecture Recognition Totally
    Unconstrained Handwritten Numerals.<titulo>
    <veiculo_publicacao>International Journal of Pattern
    Recognition and Artificial Intelligence</veiculo_publicacao>
</registro_autoria>
```

Figura 3.4. Registros de autoria do quarto exemplo de falha do método HHC

Capítulo 4

Uma Ferramenta para Resolução de Ambigüidade em Bibliotecas Digitais

Neste capítulo é apresentada uma ferramenta baseada no método HHC que tem por objetivo auxiliar os administradores de bibliotecas digitais na tarefa de resolução de ambigüidade entre nomes de autores catalogados nos respectivos repositórios. A ferramenta extrai os registros conforme definidos pelo protocolo OAI-PMH, realiza o processo de resolução de ambigüidade entre os nomes dos autores e gera um arquivo de saída no mesmo formato, atribuindo identificadores para os autores das citações, visando estabelecer uma correspondência entre um autor no mundo real e um nome de um autor em uma citação. Conseqüentemente, dado um autor de uma citação cujo identificador seja A, é possível obter todas as demais citações desse mesmo autor em uma coleção. Para isso, basta selecionar entre as citações aquelas que possuirem um autor com o mesmo identificador A.

Para atribuir esses identificadores, a ferramenta utiliza o identificador de grupo definido automaticamente pelo método HHC. Assim, ao obter a lista de grupos gerada pela execução do método HHC sobre os registros de autoria, a ferramenta percorre cada grupo dessa lista. Para cada registro de um grupo, ela recupera o autor do registro de citação respectivo e atribui a ele o identificador do grupo.

Com o arquivo de resultado, os administradores podem atualizar os repositórios de suas bibliotecas digitais, utilizando para isso funções já disponíveis ou ferramentas desenvolvidas especificamente para essa finalidade.

4.1 O Protocolo OAI-PMH

O protocolo OAI-PMH[1], *Open Archives Initiative Protocol for Metadata Harvesting*, é um protocolo para transferência de metadados entre bibliotecas digitais. Ele define um arcabouço para permitir a interoperabilidade entre fontes de dados independentemente de suas arquiteturas.

O protocolo OAI-PMH possui duas classes de participantes:

- Provedores de dados que fazem uso do protocolo OAI-PMH para exportar seus metadados.
- Provedores de serviços que utilizam os metadados colhidos através do protocolo OAI- PMH como base para construir serviços de valor agregado.

O protocolo é baseado na colheita de metadados, portanto a interoperabilidade entre as fontes de dados não é direta, ou seja, um provedor de serviços deve fazer regularmente a colheita de metadados dos provedores de dados para depois disponibilizar o conteúdo obtido.

A comunicação entre o servidor do provedor de dados e o programa de colheita do provedor de serviços para a transferência de metadados é unidirecional e funciona da seguinte forma. O provedor de serviços faz requisições ao provedor de dados, que responde enviando os metadados solicitados. As requisições do provedor de serviço são feitas usando o protocolo HTTP, através de comandos CGI codificados por meio dos métodos GET ou POST. As requisições são respondidas pelo provedor de dados com o envio de dados das respostas ou metadados dos documentos armazenados, codificados em XML. O processo de colheita baseado no protocolo OAI-PMH é ilustrado na Figura 4.1. Para a realização da colheita dos metadados, o protocolo define seis tipos de requisição que um provedor de serviços pode enviar a um provedor de dados para coletar qualquer tipo de dado do repositório: Identify, ListSets, ListMetadataFormats, ListIdentifiers, ListRecords e GetRecord.

O protocolo OAI-PMH utiliza o padrão Dublin Core[2] e estabelece o Dublin Core Metadata Element Set como o conjunto mínimo de metadados a ser suportado pelos provedores de dados em resposta a uma requisição de um provedor de serviços. No entanto, o protocolo OAI-PMH contempla o conceito de múltiplos conjuntos de metadados, permitindo que as comunidades exponham seus metadados nos formatos que são específicos a suas aplicações e domínios. O protocolo não coloca assim nenhuma limitação à natureza de tais conjuntos de metadados, com exceção daquela que

[1] http://www.openarchives.org/OAI/openarchivesprotocol.html
[2] http://dublincore.org

os registros dos metadados sejam estruturados no formato XML e que possuam um esquema correspondente para sua validação.

Na ferramenta desenvolvida, foi implementado um componente provedor de serviços OAI-PMH para extrair os registros de citação de uma biblioteca digital. O provedor utiliza apenas a requisição do tipo ListRecords, uma vez que o objetivo é obter todos os registros de um repositório. Para isso, é feita uma requisição inicial e após receber a reposta, o componente verifica se o provedor de dados devolveu um *Resumption Token*, o que, segundo o protocolo OAI-PMH, indica que há mais registros a serem extraídos. Caso positivo, uma nova requisição é feita, passando o *Resumption Token* recebido na resposta anterior. O processo é repetido até que seja recebida uma resposta sem *Resumption Token*.

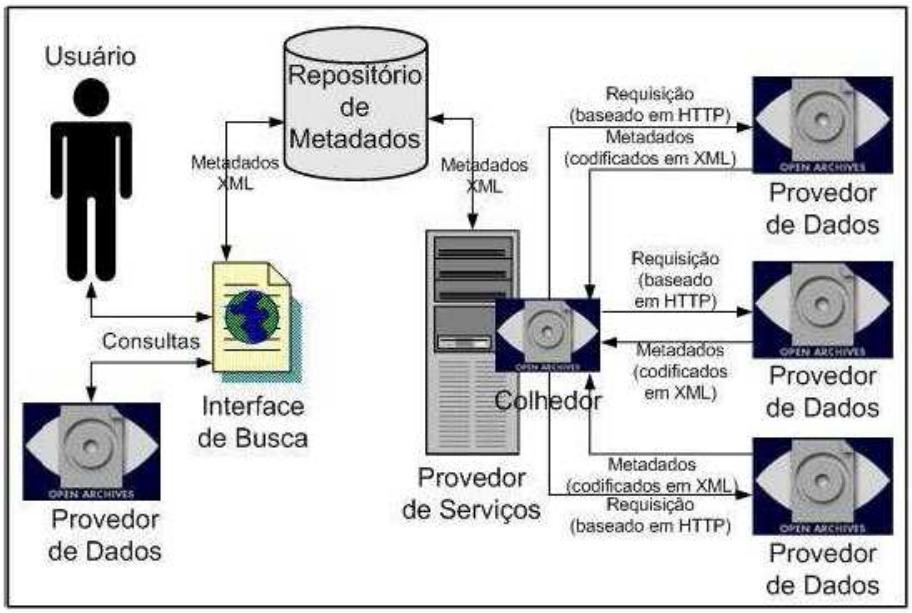

Figura 4.1. Processo de colheita baseado no protocolo OAI-PMH.

4.2 Dublin Core Metadata Element Set

O formato Dublin Core foi criado originalmente com o nome de Dublin Core Metadata em março de 1995 em um evento sobre metadados realizado em Dublin, Ohio. O objetivo principal desse formato foi identificar e definir um conjunto mínimo de

elementos capazes de descrever em larga escala recursos disponíveis na Internet. O formato Dublin Core inclui dois níveis: Simples e Qualificado. O Dublin Core Simples compreende quinze elementos: Title, Creator, Subject, Description, Publisher, Contributor, Date, Type, Format, Identifier, Source, Language, Relation, Coverage, Rights. Cada elemento é opcional e pode ser repetido. A maioria dos elementos possui um conjunto limitado de qualificadores. O formato Dublin Core é adotado mundialmente e tem sido utilizado em vários projetos que buscam um entendimento entre diferentes comunidades de usuários. Na ferramenta desenvolvida, foi incorporado um *parser* de registros no formato Dublin Core. Para utilizá-la em bibliotecas digitais que possuem registros catalogados em outros formatos de metadados, novos *parsers* devem ser incorporados à ferramenta.

4.3 Implementação

A ferramenta foi implementada na linguagem Java seguindo o paradigma de orientação a objetos. As classes implementadas possuem as seguintes funções:

- RegistroCitacao - Representa os registros de citação da biblioteca digital no formato Dublin Core;
- RegistroAutoria - Representa cada registro de autoria;
- Grupo - Representa a lista de trabalhos de um determinado autor. Contém a lista de registros de autoria e variáveis que armazenam os dados agregados de um autor: listas de termos (veículo e título), lista de co-autores;
- Colecao - Contém dados acerca da coleção de citações: lista dos registros de citação e frequência de termos;
- DublinCoreParser - Realiza a análise sintática dos registros de metadados e gera os registros de citação da coleção;
- Extrator - Implementa o protocolo de colheita OAI-PMH [Lagoze & de Sompel, 2001] para extrair os registros de uma biblioteca digital;
- GeraGrupos - Inclui as funções para a geração de grupos iniciais e para o agrupamento hierárquico. É a classe principal da ferramenta;
- Similaridade - Implementa as funções de similaridades implementadas: Distância de Levenshtein, Função do Cosseno, Coeficiente de Jaccard, Sort Winkler e o algoritmo comparação por fragmentos;

- Stemmer - Implementa o algoritmo de Porter [Porter, 1997];
- Util - Inclui várias funcões auxiliares para tratamento de cadeias de caracteres, ordenação e conversão de estruturas necessárias para tratamento de documentos XML.

4.4 Utilização da Ferramenta

A ferramenta possui uma tela com duas abas. A primeira (Figura 4.2) é usada para ativar o extrator de registros OAI-PMH e a segunda (Figura 4.3) para ativar o módulo principal. Esta divisão foi feita para isolar os processos de obtenção e processamento das citações. Caso um usuário já tenha a lista de registros de uma biblioteca no seu diretório de arquivos local, ele pode executar diretamente o módulo da aba Principal.

Figura 4.2. Tela do extrator de registros.

Para configuração do extrator OAI-PMH, são necessários os seguintes parâmetros (ver Figura 4.2):

- Endereço base OAI-PMH: deve ser configurado com a URL base do provedor OAI-PMH da biblioteca digital.
- Prefixo de metadados: define o prefixo de metadados passado diretamente como parâmetro nas requisições OAI-PMH. Este parâmetro é obrigatório para o verbo ListRecords do OAI-PMH e, caso não seja especificado, será utilizado o valor metadataPrefix=oai_dc.
- Diretório de saída: define o diretório de saída do extrator de registros de citações.

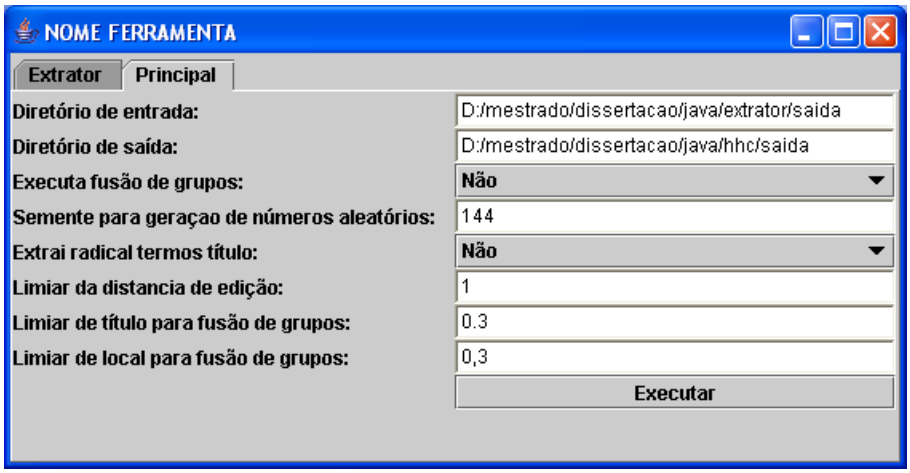

Figura 4.3. Tela principal da ferramenta.

Para configuração do módulo principal são necessários os seguintes parâmetros (ver Figura 4.3):

- Diretório de entrada: diretório de onde serão carregados os registros de citação.
- Diretório de saída: diretório onde será gravado o resultado da execução.
- Executa fusão de grupos: define se a fase de agrupamento hierárquico deve ser executada.
- Semente para geração de números aleatórios: define o valor inteiro inicial da semente para geração de números aleatórios.
- Extrai radical dos termos do título: define se os radicais dos termos dos locais devem ser extraídos. Deve ser utilizado apenas em bibliotecas em que os títulos das publicações estejam em Inglês.
- Limiar da distância de edição: define o valor (double) do limiar da distância de edição tolerado para que dois termos sejam considerados o mesmo.
- Limiar de título para fusão de grupos: define o valor (double) do limiar de similaridade de título para fusão de dois grupos.
- Limiar de nome de veículo para fusão de grupos: define o valor (double) do limiar de similaridade de veículo para fusão de dois grupos.

4.5 Validação da Ferramenta Utilizando a BDBComp

Para validar a implementação da ferramenta, utilizamos o repositório da Biblioteca Digital Brasileira de Computação (BDBComp). A BDBComp surgiu como um projeto do Laboratório de Bancos de Dados do DCC/UFMG com o objetivo de disponibilizar na Web informações bibliográficas referentes a trabalhos da comunidade brasileira de computação, suprindo a carência de um acervo brasileiro centralizado e *on-line* na área e permitindo que pesquisadores disseminassem seus trabalhos para toda a comunidade [Laender et al., 2004]. A BDBComp foi construída a partir de componentes que seguem o padrão OAI e adota o formato Dublin Core para seus metadados. Em dezembro de 2007, quando foi realizada a validação da ferramenta, ela possuia 5277 registros de metadados de trabalhos apresentados, principalmente, em eventos promovidos pela Sociedade Brasileira de Computação. O seu conteúdo vem sendo catalogado pelos administradores a partir dos dados obtidos de várias fontes. Além disso, a BDBComp possui um serviço de auto-arquivamento [Silva et al., 2007]. Dessa forma, esse ambiente serve como um cenário interessante para validação da ferramenta.

Inicialmente, o conteúdo da BDBComp foi extraído através de seu componente OAI-PMH Provider[3]. A extração feita através desse componente, além dos registros de citações de publicações, obteve também os registros de citação que identificam as coleções catalogadas, que no caso são as ocorrências dos eventos. Após a coleta, foi executado o processo de resolução de ambigüidade dos 5277 registros de citações extraídos. A partir deles, foram gerados 14856 registros de autoria, uma média de aproximadamente 2,8 registros de autoria por registro de citação. Apenas 1055 registros de autoria possuiam o nome autor no formato curto. Ao término da execução, a ferramenta definiu 8058 grupos, o que corresponderia a 8058 autores catalogados. De acordo com os grupos obtidos, pode ser visto que cerca de 73% dos autores possuem apenas um trabalho na BDBComp. A Tabela 4.1 mostra a quantidade de grupos encontrados para uma determinada quantidade de registros de autoria.

Os grupos gerados que possuiam acima de 20 de registros de autoria (32 grupos) foram individualmente analisados para verificação da qualidade interna obtida. Foi verificado se o autor de cada registro de autoria era o mesmo dos demais registros do grupo. Para isso, foram utilizados os dados do currículo Lattes e da página dos autores na Web. Em alguns casos, os artigos não eram encontrados, mas analisando o registro de autoria, o currículo Lattes ou através de busca na Web, encontrou-se a informação

[3]http://www.lbd.dcc.ufmg.br/cgi-bin/bdbcomp/oai2/oai.pl

Tabela 4.1. Número de grupos por quantidade de registros de autoria

Número de registros	*Número de grupos*	*Número de registros*	*Número de grupos*
49	1	19	4
43	1	18	8
42	1	17	6
41	1	16	5
39	1	15	19
35	1	14	8
33	1	13	12
32	1	12	16
31	1	11	22
29	2	10	24
28	4	9	31
27	1	8	46
26	4	7	72
25	1	6	77
24	1	5	137
23	2	4	193
22	2	3	433
21	2	2	1038
20	4	1	5875

necessária para concluir se o registro de autoria foi atribuído corretamente ou não. Como exemplo, havia registros de citação referentes a publicações que não estavam no currículo Lattes de um pesquisador, mas na lista de alunos orientados havia o nome de um co-autor com o título da dissertação relacionado ao título do registro de autoria.

Para esses 32 grupos, também foi verificado se havia registros de autoria isolados em outros grupos. Do total de 907 registros manualmente inspecionados, foram encontrados 21 erros (aproximadamente 2,3%) seja por registros fragmentados, seja por registros atribuídos ao autor errado. O baixo número de erros mostra que a ferramenta desenvolvida com base no método HHC apresenta bons resultados para a resolução de ambigüidade em bibliotecas reais.

Os erros encontrados podem ser classificados em três tipos: erro na catalogação dos dados, dados incompletos e falha das medidas de similaridade (ou do algoritmo de comparação por fragmentos).

No primeiro caso, verificou-se, por exemplo, que o autor José Marcos Silva Nogueira teve um registro isolado, devido a um erro na catalogação na BDBComp. No registro, dentro da mesma *tag creator* havia o nome de outro autor. A ferramenta

considerou que se tratava de um autor e criou um grupo apenas para esse registro de autoria.

No segundo caso, erro devido à falta de dados, o autor Carlos José Pereira de Lucena teve dois grupos criados isolados porque os registros de citação respectivos não continham o veículo de publicação e nesses casos também não havia co-autores em comum entre esses registros de autoria e os 28 registros que formaram o grupo principal do autor. Neste caso, a falta de uma evidência adicional além da similaridade textual dos nomes fez com que o método HHC não agrupasse os dois trabalhos.

Como exemplo de erro das medidas de similaridade, podemos citar o caso do autor Paulo Roberto Freire Cunha, que teve um registro de autoria erroneamente atribuído a ele. O autor do registro era outro pesquisador com o nome Paulo Cunha Filho.

Considerando os grupos analisados, podemos concluir que, embora a ferramenta não consiga resolver todos os casos de ambigüidade e até introduza alguns erros, agrupando citações de autores diferentes, existem ganhos significativos na sua utilização. A BDBComp poderia se beneficiar da ferramenta desenvolvida para recatalogar seu conteúdo, atribuindo um identificador interno para representar cada autor. Isso possibilitaria, por exemplo, agrupar os resultados de uma busca por nome de autor ou padronizar a exibição do nome de um autor nas suas citações.

Capítulo 5

Conclusões

5.1 Revisão do Trabalho

Citações bibliográficas desempenham um importante papel em muitos sistemas relacionados ao mundo acadêmico. Usuários desses sistemas geralmente utilizam citações bibliográficas para encontrar informação de interesse e pesquisadores as utilizam para estimar o impacto de um artigo. Adicionalmente, quando o conteúdo de diferentes bibliotecas digitais é integrado, elas servem como identificadores dos documentos associados. Portanto, as citações bibliográficas devem ser mantidas o mais consistente e atualizadas possível. Isto não é uma tarefa fácil, considerando os vários tipos de erro que podem ocorrer, entre os quais, podemos citar erros na catalogação dos dados, variedade de formatos, nomes de autores ambíguos, abreviação dos nomes de veículos de publicação [Lee et al., 2007].

Nesta dissertação, foi apresentado um método de agrupamento hierárquico baseado em heurísticas (HHC) para tratar o problema de resolução de ambigüidade entre nomes de autores em citações bibliográficas. O método baseia-se em várias heurísticas que exploram os componentes das citações (nomes de co-autores, título do trabalho, nome do veículo de publicação) para fundir sucessivamente grupos de citações de autores com nomes compatíveis.

Experimentos realizados com uma base de dados extraída da DBLP mostram ganhos acima de 12% sobre um método anterior que utiliza o mesmo algoritmo de identificação de padrões na resolução de ambigüidade entre nomes [Oliveira, 2005] mas não utiliza agrupamento hierárquico. Comparado a uma estratégia supervisionada baseada no classificador SVM, o método a supera em 21%. Em relação a uma estratégia baseada na utilização de um algoritmo de agrupamento não-supervisionado (K-Means), o ganho é de 15,5%.

Por fim, foi apresentada uma ferramenta para auxiliar os administradores de coleções de bibliotecas digitais na tarefa de remoção de ambigüidade entre nome de autores que utiliza o método HHC. A ferramenta extrai as citações de uma coleção através do protocolo OAI-PMH [Open Archives Initiative, 2006], executa o método HHC e ao fim da execução atribui um identificador único a cada autor. A ferramenta foi avaliada utilizando o repositório da BDBComp e os resultados obtidos mostraram a efetividade do método em um caso real. Entretanto, tendo em vista ser a BDBComp uma biblioteca digital relativamente pequena e restrita a uma comunidade específica, é necessário ainda estudar a escalabilidade do método. Esse e outros tópicos são abordados na próxima seção.

Os exemplos de falhas discutidos na Seção 4.5 mostram a dificuldade para se resolver ambigüidade entre nomes de autores sem qualquer supervisão humana. Se um método exigir uma grande similaridade entre os componentes das citações para agrupá-las, após o processamento muitas citações ainda podem ficar isoladas em grupos menores. Isso ocorre devido à dinâmica dos autores em relação às suas atividades de pesquisa, por exemplo, publicando trabalhos sobre diversos temas, interagindo com vários grupos de pesquisa e contribuindo em vários veículos de publicação. Por outro lado, se um método exigir um baixo limiar de similaridade, a ocorrência de falsos positivos aumenta consideravelmente, causando impacto na qualidade dos grupos gerados.

Considerando os grupos analisados no estudo de caso da BDBComp, podemos concluir que, embora a ferramenta desenvolvida não consiga resolver todos os casos de ambigüidade e até introduza alguns erros existem ganhos significativos na sua utilização. A BDBComp poderia se beneficiar da ferramenta desenvolvida para recatalogar seu conteúdo, possibilitando, por exemplo, agrupar os resultados de uma busca por nome de autor ou padronizar a exibição do nome de um autor nas suas citações.

5.2 Trabalhos Futuros

Os resultados obtidos nesta dissertação abrem várias perspectivas para trabalhos futuros, tanto na melhoria do método HHC, quanto na sua aplicação em vários contextos. O método HHC trabalha com três fontes de evidência (nomes dos co-autores, título do trabalho e nome do veículo de publicação), pois essas são as evidências disponíveis em citações bibliográficas. Entretanto, ele pode ser estendido para trabalhar com outras evidências, que podem ser obtidas, por exemplo, de cabeçalhos de publicações, rede de colaboração ou buscas na Web. Com a obtenção de novas evidências, novas formas podem ser testadas para combiná-las. Por exemplo, pode-se utilizar programação

genética para encontrar boas combinações. Além disso, melhorias das evidências podem ser feitas, por exemplo, criando arquivos de autoridade para nomes dos veículos de publicação.

Como mencionado anteriormente, é necessário fazer um estudo detalhado sobre a performance do método para verificar a viabilidade de sua aplicação em grandes bibliotecas digitais como a DBLP.

Finalmente, deve ser avaliada a capacidade do método HHC de resolver ambiguidade entre nomes de autores de citações de outras áreas além da Ciência da Computação de modo a ser adaptado para outros contextos. Dentre eles, podemos sugerir o estudo do método HHC em aplicações de conteúdo geográfico, onde poderia-se utilizar o método de casamento aproximado de cadeias de caracteres proposto em [Davis & de Salles, 2007] em tarefas como remoção de ambiguidade na recuperação de informação geográfica e no casamento de endereços, por exemplo, em *gazetteers*.

Referências Bibliográficas

Baeza-Yates, R. & Ribeiro-Neto, B. (1999). *Modern Information Retrieval*. Addison Wesley,, New York.

Bekkerman, R. & McCallum, A. (2005). Disambiguating web appearances of people in a social network. In *Proceedings of the 14th International Conference on World Wide Web*, pp. 463--470, Chiba, Japan. ACM.

Chang, C.-C. & Lin, C.-J. (2001). *LIBSVM: a library for support vector machines*. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

Cota, R. G.; Gonçalves, M. A. & Laender, A. H. F. (2007). A heuristic-based hierarchical clustering method for author name disambiguation in digital libraries. In *Proceedings of the 22nd Brazilian Symposium on Databases*, pp. 20--34, João Pessoa, PB.

Davis, C. A. & de Salles, E. (2007). Approximate string matching for geographic names and personal names. In *Proceedings of the IX Brazilian Symposium on GeoInformatics*, pp. 49–60, Campos do Jordão, SP.

Han, H.; Giles, L.; Zha, H.; Li, C. & Tsioutsiouliklis, K. (2004). Two supervised learning approaches for name disambiguation in author citations. In *Proceedings of the 4th ACM/IEEE-CS Joint Conference on Digital Libraries*, pp. 296--305, Tuscon, AZ, USA. ACM Press.

Han, H.; Xu, W.; Zha, H. & Giles, C. L. (2005a). A hierarchical naive bayes mixture model for name disambiguation in author citations. In *Proceedings of the 2005 ACM Symposium on Applied Computing*, pp. 1065--1069, Santa Fe, New Mexico. ACM Press.

Han, H.; Zha, H. & Giles, C. L. (2005b). Name disambiguation in author citations using a k-way spectral clustering method. In *Proceedings of the 5th ACM/IEEE-CS Joint Conference on Digital Libraries*, pp. 334--343, Denver, CO, USA. ACM Press.

Huang, J.; Ertekin, S. & Giles, C. L. (2006). Efficient name disambiguation for large-scale databases. In *Proceedings of the 10th European Conference on Principles and Practice of Knowledge Discovery in Databases*, pp. 536--544, Berlin, Germany.

Laender, A. H. F.; Gonçalves, M. A. & Roberto, P. A. (2004). BDBComp: building a digital library for the Brazilian computer science community. In *Proceedings of the 4th ACM/IEEE-CS Joint Conference on Digital Libraries*, pp. 23–24, Tucson, AZ, USA. ACM Press.

Lagoze, C. & de Sompel, H. V. (2001). The Open Archives Initiative: building a low-barrier interoperability framework. In *Proceedings of the 1st ACM/IEEE-CS Joint Conference on Digital Libraries*, pp. 54--62, Roanoke, Virginia, USA. ACM Press.

Lamport, L. (1994). *LATEX (2nd ed.): A Document Preparation System: User's Guide and Reference Manual*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.

Lapidot, I. (2002). Self-organizing-maps with bic for speaker clustering. Technical Report 02-60, IDIAP Research Institute, Martigny, Switzerland.

Lee, D.; Kang, J.; Mitra, P.; Giles, C. L. & On, B.-W. (2007). Are your citations clean? *Communications of the ACM*, 50(12):33--38.

Lee, D.; On, B.-W.; Kang, J. & Park, S. (2005). Effective and scalable solutions for mixed and split citation problems in digital libraries. In *Proceedings of the 2nd International Workshop on Information Quality in Information Systems*, pp. 69--76, Baltimore, Maryland. ACM Press.

Levenshtein, V. I. (1966). Binary codes capable of correcting deletions, insertions, and reversals. *Soviet Physics Doklady*, 10(8):707--710.

Malin, B. (2005). Unsupervised name disambiguation via social network similarity. In *Proceedings of the Workshop on Link Analysis, Counterterrorism, and Security, in conjunction with the SIAM International Conference on Data Mining*, pp. 93--102, Newport Beach, CA. ACM Press.

McRae-Spencer, D. M. & Shadbolt, N. R. (2006). Also by the same author: Aktiveauthor, a citation graph approach to name disambiguation. In *Proceedings of the 6th ACM/IEEE-CS Joint Conference on Digital Libraries*, pp. 53--54, Chapel Hill, NC, USA. ACM Press.

Oliveira, J. W. A. (2005). Uma Estratégia para Remoção de Ambiguidades na Identificação de Autoria de Objetos Bibliográficos. Master's thesis, Departamento de Ciência da Computação, UFMG, Belo Horizonte.

On, B.-W.; Elmacioglu, E.; Lee, D.; Kang, J. & Pei, J. (2006). An effective approach to entity resolution problem using quasi-clique and its application to digital libraries. In *Proceedings of the 6th ACM/IEEE-CS Joint Conference on Digital Libraries*, pp. 51--52, Chapel Hill, NC, USA. ACM Press.

On, B.-W.; Lee, D.; Kang, J. & Mitra, P. (2005). Comparative study of name disambiguation problem using a scalable blocking-based framework. In *Proceedings of the 5th ACM/IEEE-CS Joint Conference on Digital Libraries*, pp. 344--353, Denver, CO, USA. ACM Press.

Open Archives Initiative (2006). The Open Archives Initiative Protocol for Metadata Harvesting (OAI-PMH) - v2.0. On-line. Disponível em: `http://www.openarchives.org/OAI/openarchivesprotocol.html`.

Porter, M. F. (1997). An algorithm for suffix stripping. *Readings in information retrieval*, pp. 313--316.

Silva, L. V.; Gonçalves, M. A. & Laender, A. H. F. (2007). Evaluating a digital library self-archiving service: The BDBComp user case study. *Inf. Process. Manage.*, 43(4):1103--1120.

Song, Y.; Huang, J.; Councill, I. G.; Li, J. & Giles, C. L. (2007). Efficient topic-based unsupervised name disambiguation. In *Proceedings of the 7th ACM/IEEE-CS Joint Conference on Digital Libraries*, pp. 342–351.

Vapnik, V. (1998). *Statistical Learning Theory*. Wiley, New York.

Vu, Q. M.; Masada, T.; Takasu, A. & Adachi, J. (2007). Using a knowledge base to disambiguate personal name in web search results. In *Proceedings of the 2007 ACM Symposium on Applied Computing*, pp. 839--843, Seoul, Korea. ACM.

Witten, I. H. & Frank, E. (2005). *Data Mining: Practical machine learning tools and techniques, 2nd Edition*. Morgan Kaufmann, San Francisco. MIT Electrical Engineering and Computer Science Series.