HERCULANO GRIPP NETO

Orientador: Anderson Almeida Ferreira

UM MÉTODO PARA IDENTIFICAÇÃO DE UM CONJUNTO REPRESENTATIVO DE CITAÇÕES BIBLIOGRÁFICAS PARA REMOÇÃO DE AMBIGUIDADE DE NOMES DE AUTORES DE ARTIGOS CIENTÍFICOS

Ouro Preto

Janeiro de 2013

Universidade Federal de Ouro Preto
Instituto de Ciências Exatas
Bacharelado em Ciência da Computação

UM MÉTODO PARA IDENTIFICAÇÃO DE UM CONJUNTO REPRESENTATIVO DE CITAÇÕES BIBLIOGRÁFICAS PARA REMOÇÃO DE AMBIGUIDADE DE NOMES DE AUTORES DE ARTIGOS CIENTÍFICOS

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

HERCULANO GRIPP NETO

Ouro Preto
Janeiro de 2013

UNIVERSIDADE FEDERAL DE OURO PRETO

FOLHA DE APROVAÇÃO

Um Método para Identificação de um Conjunto Representativo de Citações Bibliográficas para Remoção de Ambiguidade de Nomes de Autores de Artigos Científicos

HERCULANO GRIPP NETO

Monografia defendida e aprovada pela banca examinadora constituída por:

Dr. Anderson Almeida Ferreira – Orientador
Universidade Federal de Ouro Preto

Dr. David Menotti Gomes
Universidade Federal de Ouro Preto

Dr. Guilherme Tavares de Assis
Universidade Federal de Ouro Preto

Ouro Preto, Janeiro de 2013

Resumo

Problemas de ambiguidade de nomes de autores de artigos científicos são comumente encontrados em repositórios de bibliotecas digitais, devido à grande variedade de fontes de dados utilizadas para a coleta dos dados, à falta de padronização dos mesmos nos meta-dados dos artigos, a abreviações, etc. Assim, é comum encontrar, em bibliotecas digitais, artigos de um mesmo autor com variações do nome deste autor, o que pode levar a tratar cada nome como se fosse de um autor diferente, ou encontrar autores distintos com o mesmo nome em seus artigos. Essa última situação pode levar a tratar todos os artigos como se fossem de um mesmo autor. Vários métodos já foram propostos para tentar resolver este problema. No entanto, nenhum deles o resolve completamente. O objetivo deste trabalho é propor um método capaz de identificar um conjunto de registros de citações representativos presentes em coleções de artigos de um mesmo autor para serem usados no processo de desambiguação de nomes. Registros representativos são aqueles que melhor refletem os trabalhos de um autor, contendo os autores com quem ele trabalhou, os assuntos tratados por este e os veículos de publicação onde difundiu seus trabalhos. Isso visa mostrar que não é necessário usar todos os registros de artigos já inseridos no repositório de uma biblioteca digital no processo de desambiguação incremental, ou seja, para desambiguar os novos registros a serem inseridos no repositório. O método desenvolvido irá auxiliar o método INDi (*Incremental unsupervised Name Disambiguation*) de remoção de ambiguidade de nomes de autores, no processo de identifição de autores. Para avaliar os resultados foram utilizadas duas coleções: a primeira com 361 registros extraídos da BDBComp e a segunda com 41672 registros extraídos da DBLP.

Palavras-chave: Ambiguidade de Nome, Bibliotecas Digitais, Citações Bibliográficas, Split Citation, Mixed Citation, Citação Representativa, Grupo Representativo

Abstract

Ambiguous author name problems are commonly found in digital library repositories due to the variety of data sources used to collect the data, the lack of standardization of metadata, the abbreviations, and so on. Thus, it is common to find articles of the same author with variations of her name or find distinct authors with the same name in your articles. This may lead to treat each name as belong to different authors, or to treat all articles belonging to the same author, respectively. A lot of methods have been proposed to solve this problem. However, none of them completely solves. This work aims to propose a method capable of identifying a set of representative citation records in collections of articles assigned to a given author and use such technique in a disamguation process. Representative records are of those that better represent the work of an author, containing the authors with whom he worked, the subject discussed in her publications and the publication venues of her articles. Such a selection aims to show that, with only some of the records already inserted into a digital library, it is possible to obtain performance close to the use of all records in a incremental disambiguation process. The technique developed was used by INDi (Incremental unsupervised Name Disambiguation) to disambiguate the ambiguous author names. To evaluate the results it was used two collections: The first one contains 361 records extracted from BDBComp and second one contains 41,672 records extracted from DBLP.

Keywords: Ambiguous Name, Digital Library, Bibliographic Citation, Split Citation, Mixed Citation, Representative Citation, Representative Group

Dedico este trabalho primeiramente aos meus pais, José (in memorian) e Valdelice, pelo esforço, dedicação e compreensão, em todos os momentos desta e de outras jornadas.

Ao meu irmão Gustavo, pelo incentivo e apoio.

Agradecimentos

Agradeço aos professores do DECOM, principalmente ao Professor Anderson Almeida Ferreira, pela contribuição, para o desenvolvimento desta monografia e especialmente pela dedicação e paciência apresentados no decorrer de todas atividades.

A UFOP, por proporcionar uma educação gratuita e de qualidade. E a todos que contribuiram direta ou indiretamente para a conclusão deste trabalho.

Sumário

Lista de Figuras

Lista de Tabelas

Lista de Algoritmos

Capítulo 1

Introdução

O problema de ambiguidade de nomes pode ser observado em diversos contextos. Esse problema afeta principalmente os sistemas computacionais que, na maioria das vezes, não conseguem identificá-los e corrigí-los. Alguns exemplos de ambiguidade de nomes são encontrados em nomes de lugares como a cidade Ouro Preto do estado de Minas Gerais e o bairro Ouro Preto localizado na cidade Belo Horizonte em Minas Gerais, em nomes de pessoas como os ex-presidentes dos EUA George W. Bush e George H. W. Bush ou ainda em nomes de veículos de publicação onde, por exemplo, SBBD e Simpósio Brasileiro de Banco de Dados se remetem ao mesmo simpósio.

Este trabalho trata do problema de ambiguidade de nomes de pessoas, mais especificamente do nome de autores em registros de artigos científicos, que comumente são armazenados em repositórios de bibliotecas digitais (DLs), que são sistemas de informação complexos, que são projetados para um público específico, possuem um conjunto grande de objetos digitais e seus meta-dados, várias estruturas organizacionais e fornecem diversos serviços para manter e acessar esses objetos digitais (Gonçalves et al., 2004). As bibliotecas digitais são importantes sistemas de gestão de informação na Internet. As DLs são fontes massivas de informação para diversos segmentos. Atualmente, são de grande relevância em diversas áreas, principalmente a acadêmica e, com a ajuda desta, vem atingindo um alto crescimento. Um artigo científico em um repositório de uma biblioteca digital é representado pelos seus meta-dados que são constituídos por pelo menos os nomes dos autores, o título do trabalho, o título do veículo de publicação e o ano de publicação.

No contexto de bibliotecas digitais, o problema da ambiguidade de nomes de autores pode afetar a busca por artigos de um determinado autor, a análise de padrões de colaboração em redes sociais, a análise de qualidade e de impacto das publicações de um autor ou um grupo de autores, dentre outros.

As seções a seguir descrevem bibliotecas digitais, o problema de ambiguidade de nomes de autores neste contexto, bem como o escopo a ser abordado neste trabalho.

1.1 Ambiguidade de Nomes de Autores em Artigos Científicos

A ambiguidade de nomes de autores em artigos científicos é um problema presente em várias bibliotecas digitais acadêmicas ou bibliotecas digitais de artigos científicos. Segundo Lee et al. (2005), pode-se dividir este problema em dois sub-problemas: *split citation*(SC) e *mixed citation*(MC).

O primeiro ocorre quando há uma variação no modo como o nome de um autor é representado nos registros dos artigos. Nesse caso, publicações de um mesmo autor podem estar divididas como se pertencessem a pessoas distintas. Isso acontece principalmente devido a erros ortográficos, abreviações ou ainda a mudanças na ordem de nome e sobrenomes.

A Figura 1.1 ilustra um caso de SC retirado de uma pesquisa na BDBComp[1] para a autora "Aleksandra Silva". Neste exemplo, foram recuperadas três diferentes representações do nome de uma mesma autora, sendo elas, "Aleksandra Silva", "Aleksandra do Socorro Silva" e "Aleksandra do Socorro da Silva" em que todas as citaçoes se referem a mesma autora.

Figura 1.1: Exemplo de *Split Citation* extraído da BDBComp

O problema MC ocorre quando diferentes autores compartilham o mesmo nome ou a mesma variação de nome. Neste caso, as publicações aparecem como se pertencessem a um mesmo autor. Figura 1.2 mostra um exemplo de MC retirado de uma pesquisa feita na DBLP[2] para o autor "Mohammed Zaki". Nesta figura, há registros de artigos de dois autores distintos com o nome "Mohammed Zaki": o primeiro refere-se a um docente da Universidade de Al-Zhar, na cidade de Nasr, Cairo, Egito; o outro refere-se ao docente do Departamento de Ciência da Computação do Instituto Politécnico Rensselaer nos Estados Unidos.

[1] http://www.lbd.dcc.ufmg.br/bdbcomp/
[2] http://informatic.uni-trier.de/ ley/db

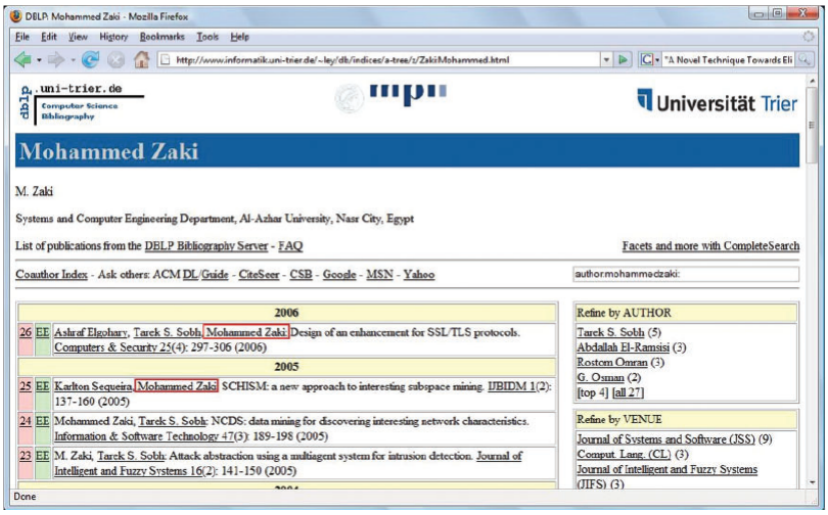

Figura 1.2: Exemplo de *Mixed citation* extraído da DBLP - Fonte: Cota et al. (2010)

1.2 Escopo

Na literatura, existem diversos trabalhos que tratam do problema de remoção de ambiguidade de nomes em uma coleção de registros de artigos científicos contidas em um repositório de uma DL (Ferreira et al., 2012). Alguns deles serão expostos e explicados no Capítulo 2.

No contexto de desambiguação de nome de autores de artigos científicos, há duas maneiras de aplicar um método de remoção de ambiguidade. A primeira, e mais comumente tratada, consiste em desambiguar todos os registros de artigos contidos no repositório de uma DL. Nesta maneira, sempre que uma nova citação é inserida no repositório da DL, o desambiguador de nomes de autores desambigua todos os registros da DL, podendo, eventualmente, corrigir alguns registros incorretamente desambiguados.

A segunda maneira fundamenta-se na inserção de novos registros, levando em consideração que os registros de artigos contidos no repositório da DL estão livres de ambiguidades. Isso quer dizer que os registros a serem desambiguados são apenas aqueles que estão sendo inseridos no repositório da DL.

O escopo deste trabalho não é limitado a somente uma destas abordagens, podendo ser aplicados a ambas, mas o avaliamos apenas na maneira incremental. Este trabalho apresenta um algoritmo para identificar os registros mais representativos de um conjunto de registros, ou seja, um subconjunto do conjunto de registros que pertencem a um determinado autor a. O intuito é utilizar os registros mais representativos de um grupo no processo de desambiguação visando obter um resultado tão bom quanto o obtido usando todos os registros. Isso significa que, em um grupo, um número n de registros serão selecionados como os registros mais representativos e estes serão utilizados para identificar se um novo registro a ser inserido no repositório pertence a este grupo.

1.3 Justificativa

A quantidade de registros de artigos científicos nos repositórios das bibliotecas digitais vem aumentando enormemente e utilizar todos esses registros no processo de desambiguação incremental pode elevar o custo computacional da desambiguação, uma vez que todos os registros poderiam ser comparados ao novo registro. Neste sentido, este trabalho visa propor uma forma de selecionar apenas alguns registros já existentes na DL e usar apenas esses registros na desambiguação de novos registros.

Além disso, no contexto de bibliotecas digitais, o problema da ambiguidade de nomes de autores pode afetar a busca por artigos de um determinado autor, análise de padrões de colaboração em redes sociais, análise de qualidade de impacto das publicações de um autor ou um grupo de autores, dentre outros.

Tratando-se da busca por artigos de um determinado autor a, caso haja ambiguidade no nome deste autor, os resultados da pesquisa poderiam não conter todas as publicações deste autor, ou até, conter publicações que pertençam a outros autores.

Na análise de padrões de colaboração em redes sociais, existem dois problemas: a existência de ligação entre dois colaboradores que não deveriam existir e a inexistência de uma ligação entre colaboradores que deveria existir. Nestes problemas, autores com áreas de atuação totalmente distintas podem ser conectados erroneamente.

A análise de qualidade de impacto das publicações é um dos problemas mais graves, pois a avaliação de um pesquisador está diretamente relacionado com a qualidade de suas publicações. Isso quer dizer que se uma publicação de um autor não constar em sua lista de publicações, devido a algum erro, a avaliação do seu trabalho estará prejudicada.

1.4 Objetivos

1.4.1 Objetivo geral

- Propor um método que identifique os registros mais representativos de um grupo de publicações de autores.

1.4.2 Objetivos específicos

- Fazer uma revisão bibliográfica sobre métodos de remoção de ambiguidade.
- Manter uma coleção de regsitros de artigos científicos livre de ambiguidade usando na desambiguação um menor número de registros.
- Avaliar os resultado do método proposto aplicado ao método INDi.

1.5 Organização do Trabalho

O restante deste trabalho está organizado da seguinte forma: o Capítulo 2 descreve alguns trabalhos publicados na literatura relacionados ao problema de ambiguidade de nomes. O Capítulo 3 trata das definições teóricas usadas neste trabalho. O quarto capítulo descreve estruturas referentes ao problema e o método proposto. O Capítulo 5 descreve e analisa os experimentos. E, finalmente, o Capítulo 6 apresenta as conclusões.

Capítulo 2

Trabalhos Relacionados

Encontram-se na literatura diversos trabalhos sobre o problema de desambiguação de nomes de autores. Os métodos propostos por esses trabalhos possuem formas diferentes para lidar com esse problema. Existem métodos baseados em técnicas de aprendizagem supervisionada, semi-supervisionada ou não supervisionada. Neste capítulo, serão apresentados trabalhos com abordagens diferentes para o problema de desambiguação de nomes.

Han et al. (2004) propõem duas abordagens baseadas em técnicas de aprendizagem supervisionada. A primeira baseia-se no modelo *naïve Bayes*, que é um modelo estatístico generativo frequentemente utilizado em tarefas de classificação e tem como objetivo capturar todos os padrões de publicações nos registros de artigos científicos. A segunda abordagem baseia-se em Máquinas de Vetores de Suporte (*Support Vector Machine - SVM*) que também são bastante utilizados em classificação. As duas abordagens diferenciam-se pelos tipos de exemplos necessários: a primeira necessita somente de exemplos positivos e a segunda necessita de ambos, ou seja, exemplos positivos e negativos para aprender a classificar os registros.

Han et al. (2005) propuseram um método de aprendizagem não supervisionada que utiliza a técnica de agrupamento *K-way Spectral Clustering*, que é baseada em grafo e tem sido aplicada com sucesso na mineração de dados e na análise de *clusters*. O método *K-way Spectral Clustering* encontra os autovalores e autovetores de uma matriz Laplaciana (ou valores singulares e vetores singulares de certos dados da matriz) relacionada com o dado grafo, e constrói *clusters* com base nessas informações. A abordagem utilizada baseia-se em 3 atributos para a desambiguação de nomes que são nomes dos co-autores, título do artigo e título do veículo de publicação.

Huang et al. (2006) apresentaram um *framework* para o problema de desambiguação de nomes em que primeiro utiliza-se um método de blocagem, que cria blocos de registros de autores com nomes semelhantes e então emprega em cada bloco um método de clusterização baseado em densidade, DBSCAN, para realizar a desambiguação. A métrica de distância usada pelo DBSCAN é aprendida usando um algoritmo de aprendizado de *SVM* de seleção ativa, conhecida como LaSVM.

Ferreira et al. (2010) propõem um método híbrido de desambiguação de nomes dividido em duas fases. Na primeira, são obtidos de forma automática os exemplos para compor o conjunto de treino que será utilizado na segunda fase. Na segunda, uma função de desambiguação é inferida usando-se os exemplos. Na fase inicial, os registros são agrupados usando uma heurística baseada em co-autoria. Após a formação desses grupos (*clusteres*), alguns destes *clusteres* são selecionados para a base de treinamento, que é constituída de exemplos retirados desses *clusteres*. Essa fase é chamada de fase não supervisionada. Na segunda fase, os exemplos de treino são usados para inferir uma função de desambiguação de nomes de autores usando características frequentes presentes nas citações (por exemplo, nome de co-autores, título e local da publicação). A função criada é usada para prever o autor das publicações dos grupos não selecionados e assim remover a ambiguidade. Essa é a fase supervisionada.

Xiaoming et al. (2011) sugerem um método de desambiguação de autores baseado em grafos. A abordagem apresentada por esses autores constrói grafos direcionais. Somente um único atributo é utilizado para a remoção da ambiguidade que é co-autoria. O Método consiste em dividir as publicações a serem desambiguadas de forma que cada *cluster* contenha somente as publicações de um mesmo autor. Inicialmente, os *clusters* criados contém registros de autores com nomes ambíguos. Para remover a ambiguidade, é utilizado um *framework* de desambiguação de nomes chamado GHOST (abreviatura para *GrapHical framewOrk for name diSambiguaTion*). Este *framework* cria um grafo de co-autoria a partir dos nomes ambiguos. Neste grafo, cada nó representa um nome de autor distinto (uma instância de cada autor); por exemplo, se dois autores a_1 e a_2 correspondem a um mesmo autor irão ser representados por dois nós diferentes no grafo. As arestas não dirigidas representam a co-autoria. Baseado nisso, é utilizado um algoritmo de caminhamento para definir se dois nomes se referem ao mesmo autor. Se dois nomes se referem ao mesmo autor, eles ficam no mesmo grupo; caso contrário, são colocados em grupos distintos.

Carvalho et al. (2011) apresenta um método que utiliza métricas para identificar se novos registros de citações bibliográficas pertencem a autores com trabalhos já cadastrados ou não na DL. Nos experimentos realizados, na maioria dos casos, os novos autores são identificados corretamente. O primeiro passo destes método, dado uma nova citação a ser inserida no repositório da DL, é realizar uma filtragem no conjunto de *clusteres*, utilizando similaridade entres os nomes dos autores para identificar os clusteres similares. Na segunda etapa, são testadas a similaridade entre co-autores e título do trabalho ou título do veículo de publicação. Caso exista um co-autor em comum e um dos dois últimos atributos citados semelhantes, a citação é considerada do autor do *cluster* selecionado. Caso não, existem ainda duas etapas para identificar o autor da nova citação. Antes destas duas etapas, os limiares de similaridade do título do trabalho e do título do veículo de publicação são incrementados. Na etapa seguinte, é verificado se a lista de co-autores da nova citação está vazia e se esta possui alguns dos dois outros atributos similares. Na última etapa, é verificado se a lista co-autores do *cluster* está

vazia e se os atributos título do trabalho e título do veículo de publicação são similares. Caso, em nenhum destas etapas, um *cluster* seja identificado, a citação é considerada como sendo um novo autor da DL.

Aprendizagem ativa é um importante campo de aprendizagem de máquina e é utilizado em problemas onde a rotulagem dos exemplos de treinamento tem um custo elevado. Bodó et al. (2011) propõem um algoritmo de agrupamento baseado em aprendizagem ativa. O algoritmo de agrupamento usado é o *K-means Spectral Clustering* que seleciona ou gera exemplos representativos do conjunto de dados para serem usados no processo de aprendizagem ativa. Os exemplos selecionados como representativos são os centróides relativos aos grupos gerados pelo método citado. Com os exemplos selecionados é utilizado um algoritmo de aprendizagem de SVM (Support Vector Machines) para aprender o modelo de classificação.

Zhao et al. (2012) apresenta um algoritmo semi-supervisionado de agrupamento de documentos e um novo método para selecionar ativamente níveis de instância de restrição para melhorar o desempenho do agrupamento. O algoritmo semi-supervisionado utilizado é o DBSCAN com restrição (*Cons-DBSCAN*), que incorpora o nível de instâncias de restrição para orientar o processo de agrupamento do DBSCAN. Nível de instância de restrição são tipos de dados supervisionados fornecidos pelo usuário. Neste trabalho existem dois tipos de niveis de instância de restrição: *mustlink* significa que um par de documentos devem pertencer a um mesmo grupo; e *cannot-link* que significa que um par de documentos devem pertencer a grupos distintos. A abordagem de aprendizagem ativa seleciona o nível de instância de restrição mais informativo para o agrupamento de pares de documentos. Existem dois fatores para determinar se os níveis de instâncias de restrição são mais informativos a um par de documentos: conter pelo menos um ponto de cada agrupamento envolvido em outro grupo; e existir restrições que controlem o limite de cada grupo. Os resultados dos experimentos deste trabalho mostram que o *Cons-DBSCAN* com a abordagem de seleção ativa pode melhorar significamente a performance do agrupamento com uma quantidade relativamente pequena rições.

Nguyen e Smeulders (2004) tratam de duas classes de aprendizagem ativa e propõem um modelo formal para a incorporação de agrupamento em apredizagem ativa. O modelo permite selecionar exemplos de treino mais representativos. Na primeira classe é construído um classificador sobre os exemplos representativos. O critério de seleção dá prioridade a dois tipos de seleção: amostras próximas ao limite de classificação e amostras que são representantes dos grupos. Após isso, é utilizada uma estrutura probabilística, chamada modelo de ruído local, para propagar a decisão de classificação para as demais amostras não representativas. Durante o processo de aprendizagem ativa o agrupamento é ajustado usando uma estrátegia chamada *coarsed-to-fine* (em português, de grosso para fino) para equilibrar a vantagem dos grandes aglomerados e da precisão de representação dos dados.

Capítulo 3

Fundamentação Teórica

Neste capítulo, serão apresentados aspectos teóricos relevantes ao trabalho. Dentre eles, as métricas de similaridade entre cadeias de caracteres, as métricas de avaliação e o método utilizado como *baseline*.

3.1 Definições

No Capítulo 1, foram apresentados o problema de ambiguidade de nomes em registros de artigos científicos de bibliotecas digitais, seus sub-problemas (*split citation* e *mixed citation*) e o escopo do trabalho.

Inicialmente, são definidos alguns termos utilizados no decorrer deste trabalho. O primeiro é o termo *registro* que é representado por meio de meta-dados de uma publicação ou artigo científico. Esses meta-dados contém pelo menos os atributos, nome do autor, nome dos co-autores, título do trabalho, título do veículo de publicação e ano em que o trabalho foi publicado.

Grupo é um conjunto de registros que deveriam pertencer a um determinado autor a_x. Entretanto, em um grupo pode haver registros que pertençam a outro autor e foram atribuídos a a_x erroneamente. Por fim, os dois últimos termos são *registro representativo* e *grupo representativo*. O primeiro termo é um registro que pode servir como um modelo das citações presentes em um grupo. O grupo representativo é um sub-grupo de registros de um determinado grupo que reúne os registros representativos deste determinado conjunto.

3.2 Métricas de Similaridade entre Cadeias de Caracteres

Nesta seção, são apresentadas as principais métricas de similaridade entre cadeias de caracteres existentes. No entanto, somente as métricas Distância de Levenshtein, Comparação por Fragmentos e Distância do Cosseno foram utilizadas no trabalho.

Distância de Levenshtein

Distância de Levenshtein (Levenshtein, 1966) foi nomeada em homenagem ao cientista russo Vladimir Levenshtein, que desenvolveu o algoritmo em 1965. É também conhecida pelo nome de distância de edição. O cálculo de similaridade utilizando distância de edição baseia-se no número mínimo de transformações (inserção, exclusão e substituição) necessário para transformar uma cadeia S em outra T. Quanto menor a distância de edição, mais similar são as cadeias. Existem diversas variações deste método: alguns atribuem pesos diferentes para cada operação e outros utilizam métricas diferentes da distância de Levenshtein como a distância de Hamming. A fórmula para similaridade entre cadeias de caracteres é:

$$lev(S,T) = 1 - \frac{ed(S,T)}{min(|S|,|T|)}$$

onde, $ed(S,T)$ calcula o número de transformações necessárias para transformar a cadeia S em T e $min(S,T)$ retorna o tamanho mínimo das cadeias de caracteres.

Comparação por Fragmentos

Segundo French et al. (2000), a similaridade por comparação de fragmentos é uma função de casamento de padrão que, por meio do algoritmo de distância de edição, avalia um a um cada fragmento (palavra) de duas cadeias de caracteres que representam nomes.

Os parâmetros necessários são duas cadeias de caracteres normalizadas, S e T, e um limiar L utilizado para a distância de edição. O resultado retornado é verdadeiro se S e T são similares, e falso caso contrário. O limiar L pode ser um número inteiro e, neste caso, é utilizado diretamente pelo algoritmo, ou um número entre 0 e 1, neste caso, o número máximo de erros permitidos será igual ao tamanho do menor fragmento comparado multiplicado por L.

Segundo Oliveira et al. (2005), a métrica de similaridade comparação por fragmentos foi desenvolvida especialmente para nomes de pessoas e vem sendo usada com sucesso em diversos trabalhos (Oliveira et al., 2005; Cota et al., 2010; Ferreira et al., 2010; Carvalho et al., 2011).

Coeficiente de Jaccard

Segundo Cohen et al. (2003), a métrica de similaridade de Jaccard é baseada em fragmentos. Jaccard considera as cadeias a serem comparadas como palavras separadas por espaços para definir a similaridade. Sua equação é dada a seguir:

$$jaccard = \frac{S \cap T}{S \cup T}$$

onde, S e T são conjuntos de fragmentos obtidos pela quebra em palavras das duas cadeias de entrada S e T, respectivamente. Esta métrica retorna o quociente do número de fragmentos

que representam a intersecção dos conjuntos S e T pelo número de fragmentos que representam a união desses conjuntos.

Distância do Cosseno

Segundo Salton et al. (1975), uma cadeia é uma sequência de palavras. Existe um conjunto finito de palavras que determinam o vocabulário. Logo, em cada elemento de uma cadeia, podem aparecer quaisquer palavras do vocabulário. Sendo assim, é possível criar uma representação vetorial em um espaço Euclidiano multidimensional para cada elemento do conjunto. Cada eixo deste espaço corresponde a uma palavra. A coordenada de um elemento e na direção correspondente a uma palavra p é determinada por duas medidas:

- TF (*term frenquecy*), que corresponde ao número de vezes que uma palavra p aparece em uma cadeia.
- IDF (*inverse document frenquecy*), que corresponde ao peso da palavra de acordo com o inverso da frequência nas cadeias.

Sendo E o conjunto de todas as cadeia de um repositório R e Ep o conjunto de cadeias de E que contêm determinada palavra p, uma forma comum para o cálculo do IDF de p é:

$$w_p = \log\left(1 + \frac{|E|}{|E_p|}\right)$$

Cadeias longas tendem a ser favorecidas por conterem um número maior de palavras diferentes. Com isso, é necessário realizar uma normalização em função do tamanho de cadeias, que é determinada pela fórmula:

$$w_e = \sqrt{\sum w_{e,p}^2}$$

onde $w_{e,p}$ corresponde ao peso das palavras em relação à cadeia e é calculado através da regra $TF * IDF$

$$w_{e,p} = (1 + \log(f_{e,p})) * w_p$$

sendo $f_{e,p}$ o número de ocorrências da palavra p na cadeia e.

A similaridade entre dois elementos é calculada através da medida do cosseno entre suas representações vetoriais. Quanto maior o cosseno, maior a similaridade. Para isto, utilizamos a fórmula:

$$\cos(e_1, e_2) = \frac{1}{w_{e1} * w_{e2}} * \sum_{p \in (e_1 \cap e_2)} (w_{e_1,p} * w_{e_2,p})$$

A medida distância do coseno é utilizada para identificar a similaridade dos atributos título do trabalho e título do veículo de publicação entre os registros a serem inseridos e os grupos

sejam, grupo de autor ou grupo representativo. Cada palavra de um destes atributos é tratado como um termo para o calculo da medida.

3.3 Métricas de Avaliação

Nesta seção, a métrica utilizada na avaliação do método e o método base são descritos.

Para avaliar a eficácia do método proposto para a remoção de ambiguidade de nomes, foi utilizada a métrica k. A seguir, é descrita essa métrica.

Métrica k

A métrica K (Lapidot, 2002) determina o equilíbrio entre duas métricas específicas de agrupamento: pureza média do grupo (PMG) e pureza média do autor (PMA).

A PMG avalia a pureza dos grupos gerados em relação aos grupos de referência desambiguados manualmente, ou seja, verifica se os grupos gerados incluem apenas os registros pertencentes a um mesmo autor (são puros). Assim, se os grupos gerados são puros, o resultado desta métrica será 1. A fórmula para calcular PMG é:

$$PMG = \frac{1}{N} \sum_{i=1}^{q} \sum_{j=1}^{R} \frac{n_{ij}^2}{n_i}$$

onde R é o número de grupos gerados manualmente (grupos de referência), N é o número total de registros na coleção da DL, q é o número de grupos automaticamente gerados pelo método, n_{ij} é o número total de registros do grupo i gerado automaticamente pertencente ao grupo j gerado manualmente, e n_i é o número total de itens do grupo i gerado automaticamente.

A PMA avalia a fragmentação dos grupos gerados automaticamente em relação aos grupos de referência. Se houver uma baixa proporção de grupos fragmentados, o resultado estará mais próximo a 1. Seus valores variam entre 0 e 1. A fórmula para calcular PMA é

$$PMA = \frac{1}{N} \sum_{j=1}^{R} \sum_{i=1}^{q} \frac{n_{ij}^2}{n_j}$$

onde R é o número de grupos gerados manualmente (grupos referência), N é o número total de registros na coleção da DL, q é o número de grupos automaticamente gerados pelo método, n_{ij} é o número total de registros do grupo i gerado automaticamente pertencente ao grupo $j$gerados manualmente, e n_j é o número total de itens do grupo j gerado manualmente. A métrica K consiste na média geométrica entre as duas métricas anteriores. Ela combina a avaliação de ambas, pureza e a fragmentação dos grupos, gerados pelo método.

A fórmula para calcular K é:

$$K = \sqrt{PMC * PMA}$$

3.3.1 Baseline

Como o problema de remoção de ambiguidade de nomes já vem sendo estudado há anos, existem diversos trabalhos relacionados na literatura. Será utilizado um trabalho como base de comparação para o método desenvolvido. O método a ser utilizado para esta comparação é o método INDi (Carvalho et al., 2011). E se baseia em uma abordagem incremental do problema de desambiguação de nomes de autores. O método INDi foi escolhido como *baseline* pois, por se tratar de um métoto incremental, este considera apenas os novos registros no processo de desambiguação e os resultados obtidos são bons. Além disso, por desambiguar apenas os novos registro o método diminui o tempo computacional do processo de desambiguação.

Método INDi

O método INDi, *Incremental Unsupervised Name Disambiguation in Clenad Digital Libraries*, visa determinar se novos registros inseridos na DL pertencem a autores com trabalhos já cadastrados ou se tratam de novos autores da DL. Desta forma, evita que seja necessário desambiguar toda a biblioteca digital a cada inserção de novos registros. Este método prioriza a atribuição correta de um trabalho a um autor; em caso de dúvida, o trabalho é atribuido a um novo autor. Em contrapartida, o problema de divisão de publicações (*split citation*) de um autor pode aumentar.

Resumidamente, o INDi busca desambiguar os novos registros procurando um autor presente na biblioteca digital que possui registros similares. Para definir esta similaridade, o registro a ser inderido na DL deve possuir o nome do autor similar ao do grupo comparado, pelo menos um co-autor em comum com os registros deste grupo e o título do trabalho ou do veículo de publicação similares ao dos registros presentes no grupo. No caso de um registro ou de um grupo não possuir nenhum co-autor, são utilizados somente os atributos autor, título do trabalho e do veículo de publicação.

Antes que o procedimento de desambiguação ocorra, um pré-processamente é realizado nos atributos título do trabalho e do veículo de públicação dos registros a serem inseridos. Primeiro são removidas todas as pontuações e *stop words*[1]. Na sequência, o algoritmo de Potter (Porter, 1980) é utilizado para reduzir uma palavra ao seu radical. Posteriormente, cada citação a ser inserida na DL deverá passar pelos mesmos passos que os citados abaixo.

O processo de desambiguação de nomes pode ser divido em 3 etapas. Antes da primeira etapa, são selecionados os grupos cujo os nomes dos autores são similares ao do autor do registro a ser inserido. Os grupos selecionados também passam pelo mesmo preprocessamento que a nova publicação. Cada um dos grupos similares são selecionados para realizarem as 3 etapas de desambiguação até que um grupo seja identificado como o autor do registro a ser inserido.

[1] *Stop words* são palavras que aparecem com muita frequência em textos de uma determinada língua, tais como, artigos, preposições, dentre outros.

A primeira etapa verifica se existe ao menos um co-autor em comum do registro com o grupo selecionado, além de verificar se o título da publicação ou do véiculo de publicação são similares ao do grupo. Para a verificação de similaridade dos dois últimos atributos, é utilizado uma métrica de similaridade de cadeias de caracteres, com parâmetros definidos por α_{title} e α_{venue}, que são limiares de similaridade utilizados por essas métricas. Caso um grupo com essas características seja encontrado, as demais etapas são ignoradas e o processo prossegue para a próximo registro a ser inserido.

Antes do início da próxima etapa os valores de α_{title} e α_{venue} são incrementados pelo valor δ. Na etapa 2, são verificadas se a novo registro não possue co-autor. Caso seja falso o processo segue para a etapa 3. Caso verdadeiro, são testadas as similaridades do título ou do veículo de publicação com os novos parâmetros de similaridade. Se um dos dois forem similiares o grupo foi encontrado; caso contrário, o registro pertence a um novo autor da DL.

Na etapa 3, é verificada se a lista de co-autores do grupo similar está vazia e se o título ou veículo de publicação são similares aos da registro, conforme os parâmetros atualizados após o fim da etapa 1. Sendo essas condições satisfeitas, o grupo é encontrado; caso contrário, a registro pertence a um novo autor.

Essas etapas são repetidas para cada novo registro de citação inserido na DL.

Capítulo 4

Método Proposto

O objetivo do método de remoção de ambiguidade de nomes de autores é manter uma coleção de registros de artigos científicos livre de ambiguidade. O desafio é, dado registros a serem inseridos nesta coleção, identificar de forma única os seus autores. Para isso, é necessário identificar se os autores a que se referem os nomes dos registros a serem inseridos já possuem trabalhos cadastrados na biblioteca digital: se já possuem, esses registros devem ser atribuídos a esses autores já existentes; caso contrário, devem ser atribuídos a um novo autor da DL.

Indicar se um registro a ser inserido em uma DL se refere a autores com trabalhos já cadastrados nessa DL, requer iniciamente uma comparação dos nomes dos autores deste registro com os nomes dos autores que possuem trabalhos cadastrados nesta DL. O modo como essa comparação é feita, influencia diretamente na eficência e na eficácia do método de desambiguação.

Como dito no Capítulo 1, a proposta deste trabalho é criar um método que identifique os registros mais representativos de um determinado grupo, para serem utilizados no processo de desambiguação de novos registros a serem inseridos no repositório de uma DL. A intenção é adaptar um método de desambiguação para que utilize somente os registros mais representativos de cada autor no processo de desambiguação. Neste trabalho, é utilizado como base de comparação o método de desambiguação INDi, explicado em detalhes na Seção 3.3.1.

Antes de apresentar o método proposto, algumas perguntas devem ser analisadas. A primeira é: como identificar se um registro é representativo a um determinado grupo ? Inicialmente, pode-se entender que o registro que possui a maior quantidade de informações em seus meta-dados representa melhor um determinado autor a_1.

Entretanto, o propósito do método é criar um sub-grupo de registros representativos a fim de utilizá-los no processo de desambiguação de nomes. Assim, a pergunta anterior deve ser reformulada para: como criar ou identificar um grupo de registros representativos em um grupo de registros de um autor ? Considerando os registros de um grupo, quanto maior a heterogenia entre os registros dentro do sub-grupo representativo, maior será a representação do sub-grupo perante ao grupo. Diante deste quadro, vamos analisar separadamente cada

atributo de uma citação para definir as características que diferenciam duas citações de um mesmo autor.

Nas seções a seguir serão apresentados a análise dos atributos, o método proposto e a forma de utilização do método.

4.1 Análise dos Atributos

Conforme mencionado anteriormente, um registro r é representado por seus meta-dados, que são constituídos por, pelo menos, nomes de autores, título do trabalho, veículo de publicação e ano em que o trabalho foi publicado.

Tabela 4.1: Registros retirados da BDBComp - Autora "Aleksandra do Soccorro da Silva"

Id	Autor	Título	Veículo	Co-autores	Ano
r_1	aleksandra do socorro da silva	development of an intelligent tutoring system based on agents in the context of a virtual classroom adapted	xii Brazilian symposium on computer science education	a hernandez dominguez	2001
r_2	aleksandra do socorro silva	an architecture for developing interactive learning environments based on agents, components and framework	xiv Brazilian symposium on computer science education	s brito, e favero, a hernandez-dominguez, o tavares ,c frances	2003
r_3	aleksandra do socorro silva	love: learning environment multi-paradigmati	xiv Brazilian symposium on computer science education	m harb, s brito, e favero, o tavares ,c frances	2003
r_4	aleksandra silva	afm: an assistant-tutor agent-based modeling tool for object-oriented	xv Brazilian symposium on computer science education	a hernandez dominguez,b silva	2004

A Tabela 4.1 alguns registros retirados da BDBComp em uma pesquisa pela autora "Aleksandra do Socorro Silva". Todos registros presentes na tabela pertencem a mesma autora e serão utilizados para demonstrar como registros de um mesmo autor podem se diferenciar.

4.1.1 Nome do Autor

Na pesquisa realizada, quatro registros bibliográficos da autora foram recuperados e é possível identificar três diferentes formas de grafia de seu nome. O nome do autor é um fator fundamental no processo de desambiguação, pois ele é usado como um primeiro filtro, para selecionar quais são os possíveis grupos que um novo registro r_n pode pertencer. Logo, os registros representativos do grupo necessitam conter a maior variação possível da grafia do nome de um autor.

Como este atributo é primordial para uma seleção correta dos grupos, outros aspectos devem ser levados em conta, tais como, o número de vezes que determinada variação do nome ocorre no grupo e a quantidade de termos no atributo (i.e., nome e sobrenomes). Este último aspecto será útil na aplicação de métricas de similaridade entre cadeias de caracteres empregadas em diversas partes do algoritmo. Essas métricas foram explicadas na Seção 3.2.

Assim, um registro pode se diferenciar de outro através do nome autor, pela quantidade de termos e pelo número de aparições dentro do grupo. Levando em consideração somente este atributo, pode-se selecionar dois registros como candidatos ao grupo de representativo, que são os registros r_1, que possui o maior número de termos, e o r_2 ou r_3 devido ao número de ocorrências no grupo.

4.1.2 Título do Trabalho

O título de um trabalho é um atributo tão importante para um artigo científico, quanto os nomes de autor e co-autores. Por meio dele, é possível identificar a área de atuação dos pesquisadores e traçar um perfil de suas publicações. Entretanto, a tarefa de comparar dois títulos de registros diferentes e confirmar se estes tratam do mesmo problema ou de alguma área correlacionada é muito complexo, já que dois títulos podem não possuir nenhum termo lexicalmente semelhante, e mesmo assim, tratarem de uma área afim ou até de um problema comum. Existe ainda um problema, em que não há nenhum termo em comum entre os dois títulos, mas ,mesmo assim, tratam do mesmo assunto. Este trabalho usa apenas os termos presentes no título, sem tentar inferir se títulos diferentes remetem a um mesmo assunto. Logo, para dizer se dois títulos possuem algo em comum, é preciso comparar termo a termo usando métricas de similaridade entre cadeias de caracteres.

Para definir se um registro é mais representativo do que outro usando os títulos dos trabalhos, deve-se observar a quantidade de termos em cada título. O título que possui o maior número de termos será, provavelmente, o registro mais represetantivo dentro de um grupo. Quanto maior a quantidade de termos em um título provavelmente maior será a sua diversidade. Na Seção 4.2, são apresentados e explicados os processos preliminares que ocorrem no título antes de que este seja utilizado no processo de desambiguação nomes.

4.1.3 Título do Veículo de Publicação

O título do veículo de publicação de um trabalho é um atributo, na maioria das vezes, correlacionado ao título do trabalho. No entanto, as informações contidas neste são mais genéricas do que as relacionadas ao título. Deste modo, a tarefa de traçar um perfil de publicação de um autor pode se tornar um pouco mais simples. Normalmente, um autor publica seus trabalhos em determinados periódicos ligados a uma área especifica e alguns destes são os principais e mais representativos.

Como dito anteriormente, os registros apresentados na Tabela 4.1 são de um mesmo autor. Analisando o atributo título do veículo de publicação, observa-se que o autor citado sempre publica no mesmo simpósio, *"Brazilian Symposium on Computer Science Education"*, com a única diferença sendo a edição deste. No entanto, em algumas DLs o título do veículo de publicação é armazenado abreviado. Assim, a utilização do título do veículo de publicação no processo de desambiguação não é muito satisfatória, pois dois veículos de publicação diferentes podem possuir a mesma forma abreviada.

4.1.4 Nomes dos Co-autores

O atributo nome dos co-autores é um dos mais importantes e mais utilizados no processo de desambiguação. No Capítulo 2, pode-se verificar que todos os trabalhos citados que tratam do problema de desambiguação de nomes utilizam este atributo com parte fundamental do procedimento de identificar os autores.

Assim, como é comum um autor publicar trabalhos sempre em uma mesma área, é usual um autor desenvolver projetos com as mesmas pessoas. Logo, é habitual que os co-autores de um trabalho estejam presentes em outro trabalho do mesmo autor, como ocorre no registro r_1 com os demais r_2 e r_4. O co-autor de r_1, *"a hernandez dominguez"*, aparece também como co-autor nos demais registros.

No entanto, como a intenção é identificar os registros mais representativos, a idéia de escolher registros que possuem co-autores em comum pode não ser interessante. Seria mais eficiente escolher os registros que possuem maior número de co-autores e que a maior parte destes não estejam presentes nos demais registros representativos. Quanto mais co-autores não comuns, melhor será a abrangência dos registros do grupo relacionado pela ótica dos nomes de co-autores.

4.1.5 Ano de Publicação

O ano de publicação de um trabalho não é muito utilizado, na literatura, no processo de desambiguação de nomes de autores. Entretanto, pode vir a ser útil no processo de seleção de registros representativos. Suponha que dois registros, r_x e r_y, pertençam a um mesmo autor e que eles não possuam nada em comum, em termos de título de trabalho e título do veículo

de publicação, sobrando somente alguns co-autores em comum. O atributo ano de publicação poderia ser útil para identificar novos registros a serem inseridos pois, de acordo com o ano, um registro tem a possibilidade de conter atributos, como título e veículo de publicação diferentes dos usuais deste autor. Isso poderia ocorrer onde as diferenças entre os anos é maior, uma vez que um autor em suas primeiras publicações podem tratar problemas ou se relacionar a áreas diferentes das atuais ou em suas últimas publicações.

4.2 Método

Conforme dito no escopo do trabalho, o método de identificação das citações representativas de um grupo é aplicável a diversos processos de desambiguação de nomes presentes na literatura. Neste trabalho foi utilizado como *baseline* o método INDi (Carvalho et al., 2011), apresentado na Seção 3.3.1.

Na seção anterior, os atributos de um registro foram analisados com o intuito de reconhecer as características que diferenciam um registro de outro e tornam um conjunto de registros representativos diante de um grupo. É evidente que, quanto maior a variedade de informações presentes nos registros representativos mais heterogêneo será o grupo representativo e melhor retratará o grupo. No entanto, deve haver um número máximo de registros representativos (**maxCit**) dentro do grupo, pois a idéia é utilizar esse grupo como base para comparar e identificar os novos registros, evitando assim o uso de todos os registros do grupo para isso. O ideal seria que o próprio algoritmo reconhecesse automaticamente a quantidade necessária de registros para representar determinado grupo. Grupos diferentes exigem quantidades diferentes de registros representativos, visto que esse número não está ligado somente ao número de registros presentes no grupo mas também na variedade de informações presentes nestes.

Como a quantidade de registros representativos devem respeitar um limite **maxCit**, é necessário um método para escolha de registro e outro para a remoção entre registros representativos. Suponha que o conjunto representativo de um grupo está completo, com número maxCit de registros, e que um novo registro é inserido e este novo registro representa melhor o grupo do que um registro que já pertence ao grupo representativo. Neste caso, um registro do grupo representativo deverá ser removido para a adição do novo registro. Nas subseções a seguir são apresentados o método de escolha de registros representativos e o método de remoção de registros do grupo de registros representativos.

4.2.1 Escolha de registro representativo

O método de escolha de registro representativo consiste em analisar os registros existentes no grupo representativo comparando-os com o novo registro inserido no grupo.

O algoritmo de escolha de registro representativo é chamado após um registro ser inserido em um grupo. Este recebe como entrada o registro que foi inserido, o grupo ao qual ele

pertence, os parâmetros de similaridade $\alpha_{coAutores}$, $\alpha_{veiculo}$, α_{titulo}, que são os limiares de similaridade entre cadeias de caracteres.

Ele retorna **verdadeiro** se o registro deve ser incluído no grupo representativo e **falso** caso contrário. Caso o registro deva ser considerado como um registro representativo e o número de registros do grupo representativo atingiu o limite **maxCit**, a função para remoção de registro representativo, verifica se o registro inserido é mais representativo do que os registros pertencentes ao grupo representativo.

Na Tabela 4.3, é apresentado um exemplo de seleção de um registro como representativo. Os registros r_1 e r_2 já pertencem ao conjunto de registros representativos do grupo de registros de artigos do autor *"joel da silva"*. O registro r_3 foi também atribuído ao autor acima citado e foi selecionado pelo algoritmo de escolha de registro representativo.

Analisando os atributos de r_3 verifica-se que este possui um co-autor,*"r fonseca"*, que não pertence a nenhuma das listas de co-autores dos registros presentes no conjunto representativo (r_1 e r_2), por este motivo foi selecionado para compor o conjunto de registros representativos do grupo.

Tabela 4.3: Escolha de Registro Representativo

Id	Autor	Título	Veículo	Co-autores	Ano
r_3	joel da silva	Geographical querying data warehouses with geomdql	xxii Brazilian Symposium database	**r fonseca** :r fidalgo:v times	2007
r_1	joel silva	gmla: xml schema for the exchange and integration of multi-date Geographical	xix Brazilian Symposium database	r fidalgo:v times:f souza:a salgado	2003
r_2	joel da silva	An investigation of models for estimating the effort in project management software	v Brazilian Symposium on Geoinformatics	r fidalgo:v times:f sousa:r barros	2006

O Algoritmo 4.2.1 apresenta o pseudo-código da escolha de registros representativos.

Algoritmo 4.2.1: Escolha de Registro Representativo

Input: Registro *reg*; Grupo *grupo*; Parâmetros de similaridade $\alpha_{coAutores}$, α_{titulo} e $\alpha_{veiculo}$;

Output: *verdadeiro* se o Registro *reg* for representativo e *falso* caso contrário;

```
1  repre ← falso;
2  grupoRepre ← GetGrupoRepresentativo(grupo);
3  if grupoRepre está vazio then
4      return verdadeiro;
5  if ¬CoautoresSimilar(reg,grupoRepre,α_coAutores) then
6      repre ← verdadeiro;
7  else
8      if ¬TituloSimilar(reg,grupoRepre,α_titulo) ou
       ¬VeiculoSimilar(reg,grupoRepre,α_veiculo) then
9          repre ← verdadeiro;
10 if repre é verdadeiro then
11     if NumCits(grupoRepre) < maxCit then
12         return verdadeiro;
13     else
14         return removeReg(reg,grupo);
15 return repre;
```

Nas linhas 1 e 2, as variáveis **repre**, **grupoRepre** são inicializadas. A primeira se refere a representatividade do registro inserido em relação ao grupo representativo, a segunda se refere aos registros que pertencem ao grupo representativo do grupo fornecido como entrada da função. A função **GetGrupoRepresentativo** retorna a lista de registros representativos referentes ao **grupo** passado como parâmetro. Na linha 3, é verificado se o grupo representativo está vazio; caso esteja, o registro inserido no grupo deve obrigatóriamente ser inserido no grupo representativo e a funcão de escolha de registro representativo retorna **verdadeiro**.

Na linha 5, é testada se a lista de co-autores do registro não é similar a lista de co-autores do grupo represetantivo. A lista de co-autores do grupo representativo possui todos co-autores, de todos registros cadastrados neste grupo, sem a presença de duplicatas. Para definir se estas duas listas não são similares, é utilizada uma métrica de similaridade entre cadeias de caracteres, chamada Comparação por Fragmentos. A função **¬CoautoresSimilar** verifica se existe no minímo um número $\alpha_{coAutores}$ de co-autores presentes no registro inserido e não presentes nos registros do conjunto representativo. Se as lista não forem similares o registro é considerado representativo; caso contrário não e segue-se para a próxima etapa.

Caso não seja possível identificar se um registro é representativo utilizando o atributo co-

autor são usados os atributos título do trabalho ou título do veículo de publicação. Assim, a intenção é verificar se alguns destes dois atributos do registro inserido são similares ao dos registros do conjunto representativo. Na linha 8, as funções **TituloSimilar** e **VeiculoSimilar** verificam se os atributos título do trabalho e título do veículo de publicação são respectivamente similares, utilizando a métrica distância do cosseno. Os parâmetros $\alpha_{titulo} e \alpha_{veiculo}$ são respectivamente os limiares de similaridade dos atributos título do trabalho e título do veículo de publicação. Se um destes atributos do registro **reg** não for similar ao dos registros do conjunto representativo o novo registro é considerado representativo.

Por fim nas linhas 12 a 16, caso o registro seja representativo, é testado se o grupo representativo já atingiu seu limite **maxCit** de registros. Caso sim, é chamada a função para remoção de registro representativo. Esta investiga se algum registro pertencente ao grupo representativo deve ser substituído pelo novo registro considerado representativo.

4.2.2 Remoção entre registros menos representativos

O método de remoção de registro representativo é chamado sempre que um registro é definido como representativo pelo algoritmo 4.2.1 e o grupo representativo já possui o número máximo de registros permitidos para aquele grupo. O intuito deste método é encontrar um registro no conjunto representativo semelhante ao registro escolhido para se juntar a este conjunto.

Diferente do método acima, este usa como base somente os atributos nome do autor,dos co-autores e título do trabalho. Para substituir um registro presente no grupo representativo pelo novo registro inserido, a função utiliza o nome do autor para filtrar os registros. Com registros pertencentes ao grupo devidamente filtrados pelo nome autor, existem duas maneiras de identificar qual registro deve permanecer no grupo representativo, a saber: na primeira, são testados se as listas de co-autores dos dois registros são similares (**CoautoresSimilar(reg,r,**$\alpha_{coAutores}$**)**), e qual deles tem o maior número de co-autores **NumeroCoautores(reg,r)**. Se o novo registro tiver a lista de co-autores similar ao do registro selecionado e possuir maior número de co-autores, ele será definido como um registro representativo e o registro selecionado será removido do grupo representativo; na segunda, também são testados se os co-autores são similares e qual deles tem o maior número de termos no atributo título do trabalho (**NumTermosTitulo(reg,r)**). Se as duas condições forem satisfeitas o novo registro é definido como representativo e o registro selecionado será removido do grupo representativo.

A Tabela 4.5 apresenta um exemplo de remoção entre registros menos representativos. No exemplo o registro $r3$ foi inserido no grupo da autora *"lyrene fernandes da silva"* e foi escolhido como um registro representativo pelo Algoritmo 4.2.1. No entanto, o conjunto de registros representativos está completo, sendo necessário remover um registro do conjunto para inserir r_3. Para isso é verificado se $r3$ melhor representa o grupo do autor em que ele foi inserido do que um registro presente no conjunto representativo. Analisando os registros representativos

verifica-se que r_3 é mais representativo do que r_2. Este registro possui o nome autor similar, um co-autor em comum e um número menor de co-autores do que r_3.

Tabela 4.5: Remoção entre registros menos representativos

Id	Autor	Título	Veículo	Co-autores	Ano
r_3	lyrene fernandes da silva	integration of features for cross-modeling requirements	workshop on requirements engineering	c felicissimo:k breitman:**j leite**	2005
r_1	lyrene silva	a meta-model for specifying software architectures layered	xvii Brazilian Symposium of Software Engineering	m sayao:j leite:k breitman	2001
r_2	lyrene fernandes da silva	generation of ontologies subsidized by engineering requirements	workshop on requirements engineering	**j leite**	2003

O Algoritmo 4.2.2 recebe como parâmetro o novo registro considerado como representativo (**reg**), o grupo ao qual ele pertence (**grupo**), e os limiares de similaridade do autor e dos co-autores (respectivamente α_{autor} e $\alpha_{coAutores}$). Ele retorna **verdadeiro** se algum registro semelhante foi identificado e removido, e **falso** caso contrário.

Algoritmo 4.2.2: Remoção entre Registros menos Representativos

Input: Registro *reg*; Grupo *grupo*; Parâmetros de similaridade α_{autor}, $\alpha_{coAutores}$;
Output: **verdadeiro** se o registro *reg* representar melhor o grupo do que algum registro pertencente ao grupo representativo e **falso** caso contrário;

```
1  remove ← falso;
2  grupoRepre ← GetGrupoRepresentativo(grupo);
3  for r ∈ grupoRepre do
4      if AutorSimilar(reg,r,α_autor) then
5          if CoautoresSimilar(reg,r,α_coAutores) and NumeroCoAutores(reg,r) then
6              removeReg(r);
7              return verdadeiro;
8          else
9              if CoautoresSimilar(reg,r,α_coAutores) and NumeroTermosTitulo(reg,r) then
10                 removeReg(r);
11                 return verdadeiro;
12 return remove;
```

Nas linhas 16 e 17, as variáveis **remove** e **grupoRepre** são inicializados. A primeira se refere ao fato de um registro do grupo representativo ter sido removido, a segunda se refere aos registros que pertencem ao grupo representativo do grupo fornecido como entrada da função. A função **GetGrupoRepresentativo** assim como no Algoritmo 4.2.1 retorna a lista de registros representativos do **grupo** passado como parâmetro.

Na linha 18, são percorridos todos os registros do grupo representativo e filtrados pelo atributo autor com o limiar de similaridade definido por α_{autor}. Essa filtragem é feita utilizando métricas de similaridade entre cadeias de caracteres. Assim como na função de escolha de registro representativo, é utilizada a métrica de comparação por fragmentos. Após a filtragem, existem duas maneiras de identificar se um registro que pertence ao grupo representativo deve ser removido; a primeira maneira, na linha 20, são testadas se as listas de co-autores são semelhantes, também utilizando a métrica de similaridade comparação por fragmentos, para que as listas de co-autores dos dois registros sejam similares estas devem possui no máximo $\alpha_{coAutores}$ não comuns, e se o novo registro possui mais co-autores do que o registro selecionado. Sendo assim, o registro selecionado é removido (linha 21) e a função retorna **verdadeiro**; a segunda maneira, na linha 24, também são testadas se as listas de co-autores são semelhantes, da mesma forma, e se o novo registro possui mais termos no seu atributo título do trabalho do que o registro selecionado. Se essas duas condições forem satisfeitas, o registro selecionado é removido e a função retorna **verdadeiro**.

Caso nenhum registro seja selecionado por meio da similaridade dos autores ou, se forem selecionados registros e nenhuma das duas maneiras para identificar um registro forem satisfeitas, a função retorna **falso**. Assim, nenhum registro do grupo representativo será removido e o novo registro não será inserido no grupo representativo.

4.3 Utilização do Método

Nesta seção, serão apresentados o formato de entrada dos dados, o formato de saída dos dados e a forma de utilização do método proposto neste capítulo.

4.3.1 Formato de Entrada

Os registros de entrada das coleções são mantidos em arquivos de texto, separados de acordo com o atributo ano de publicação. Todos os arquivos que contém os registros de uma mesma coleção devem estar em uma mesma pasta com o nome da coleção. Os nomes dos arquivos que contém os registros são padronizados, e o formato dos mesmos é: "Base_XXXX.txt", onde XXXX é ano em que os registros contidos neste arquivo foram publicados. Agora que foi apresentada a forma como os registros são distribuidos, será apresentado o formato do registro. Segue abaixo o formato dos registros bibliográficos:

Id Registro<>Id Grupo Ambiguo_Id Autor<>Lista de co-autores<>Título do trabalho<>Veículo de publicação<>Nome do autor<>Ano de publicação

O Campo **Id Registro** é o número identificador do registro; cada registro possui um único identificador. O segundo campo, **Id Grupo Ambiguo_Id Autor**, são dois identificadores que significam respectivamente: o identificador do grupo ambiguo ao qual o registro pertence e o identificador do autor dentro deste grupo ambiguo. O campo **Lista de co-autores** fornece a lista dos co-autores deste registro, os co-autores são separados pelo caracter **:**. O **Título do trabalho** armazena o título do trabalho associado ao registro. O quinto campo é o título do veículo de publicação do artigo ao qual o registro faz referência. O penúltimo é o nome do autor e o último é o ano publicação do mesmo.

4.3.2 Formato de Saída

Os dados de saída do método podem ser divididos em dois tipos: os resultados das métricas avaliativas e os grupos de autores gerados pelo método, ambos armazenados em arquivo texto. As métricas avaliativas são armazenadas em arquivo texto com o nome "Nome_Coleção_Metricas.txt". Neste, são salvos os resultados das métricas para cada experimento realizado. Cada linha do arquivo contém os dados referentes a um experimento.

Segue abaixo o formato de saída dos resultados das métricas avaliativas:

Número_de_Anos_Carregados Métrica_PMC Métrica_PMA Métrica_K Número_de_Registros_Representativos

Os grupos de autores gerados pelo método são separados de acordo com o experimento, pois os resultados de um experimento para outro podem ser diferentes e, consequentemente, os grupos de autores gerados também. Para a saída destes dados, o método cria uma pasta nomeada como "Log_Nome_da_Coleção". Dentro desta pasta, é criada uma pasta para cada experimento, armazenando os arquivos relativos aos grupos de autores gerados. Por exemplo, ao se processar 3 experimentos sobre uma coleção de nome "DL_1", serão criadas quatro pastas: uma para a coleção e, dentro desta, outras 3 para os experimentos.

Dentro de cada pasta referente a um experimento são salvos os grupos de autores gerados. Estes são separados de acordo com o autor; para cada autor, existe um arquivo texto com suas publicações. O nome dos arquivos criados são no formato "Nome_Autor_ID_Grupo.txt", mesmo que dois autores diferentes tenham nomes iguais suas publicações não serão salvas no mesmo arquivo, devido ao **ID_Grupo** presente no nome do arquivo. Dentro de cada arquivo são armazenados o identificador do grupo, o nome do autor, os registros das publicações e os registros representativos associados aquele grupo. Segue abaixo um exemplo deste tipo de arquivo.

Id do Grupo: 163 Nome Representativo: marcelino pereira dos santos silva

---

Quantidade de Registros no Grupo = 2

302<>163<>0<>marcelino silva<>[f feitosa, g camara, a monteiro, t koschitzki]<>[deploy, process, improv, requir, engin, informat, compani, formula]<>[vi, brazilian, symposium, geoinformat]<>2004
303<>163<>1<>marcelino pereira dos santos silva<>[j robin]<>[spatial, measur, residenti, segreg]<>[xvii, brazilian, symposium, artifici, intellig]<>2004

Quantidade de Registros Representativos = 1

302<>163<>0<>marcelino silva<>[f feitosa, g camara, a monteiro, t koschitzki]<>[deploy, process, improv, requir, engin, informat, compani, formula]<>[vi, brazilian, symposium, geoinformat]<>2004

4.3.3 Forma de Utilização do Método

O método proposto foi desenvolvido em Java, na plataforma Linux Ubuntu. Para utilizar o método, basta executar o arquivo "mono.jar" e definir os seis parâmetros. Os parâmetros a serem definidos são a similaridade do título do trabalho, a similaridade do título do veículo de publicação, o valor incremental, estes se referem ao processo de desambiguação. Os demais são referentes a escolha dos registros representativos, que são eles: a quantidade de co-autores em comum, a similaridade do título do trabalho e a similaridade do título do veículo de publicação. Segue abaixo duas imagens da aplicação do método.

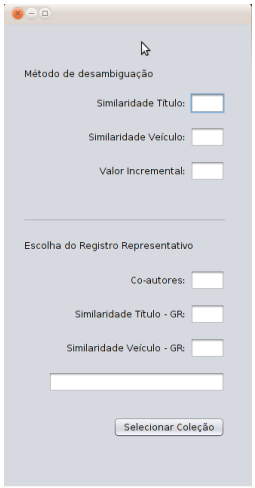

Figura 4.1: Tela 1

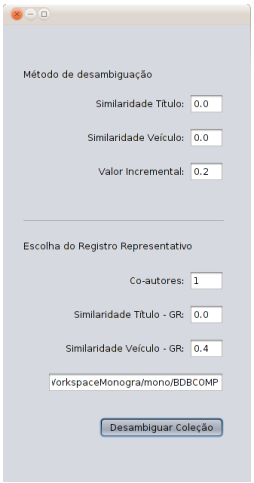

Figura 4.2: Tela 2

Na Figura 4.1, os parâmetros estão definidos e basta clicar no botão “Selecionar Coleção” e direcionar o caminho onde a coleção no formato apresentado na Seção 4.3.1, se encontra. Com a coleção selecionada, basta clicar no botão “Desambiguar Base”, da Figura 4.2.

Capítulo 5

Avaliação Experimental

Este capítulo apresenta a forma como os experimentos foram conduzidos, os resultados e a análise dos resultados. Os experimentos foram realizados sobre duas coleções, BDBComp, cujos os dados foram extraídos da biblioteca digital BDBComp e Kisti, cujos os dados foram extraídos da biblioteca digital DBLP, utilizando o método proposto neste trabalho, apresentado no Capítulo 4, e o *baseline* definido na Seção 3.3.1.

Nas seções a seguir são apresentados, as coleções, os modelos dos experimentos, os resultados e suas análises.

5.1 Coleções

Nesta seção, serão apresentadas as duas coleções utilizadas nos experimentos dos dois métodos.

Coleção BDBComp

Esta coleção possue cerca de 361 registros extraídos da biblioteca digital BDBComp[1]. Neste conjunto de registros extraídos desta DL, existem 184 autores distintos e aproximadamente 2 registros por autor. Esta coleção é difícil de ser desambiguada, porque possui muitos autores com apenas um registro. Ela contém os 10 maiores grupos ambíguos, ou seja, os 10 maiores grupos de autores, apresentados na Tabela 5.1 começando com a mesma inicial do primeiro nome e terminando com o mesmo último sobrenome, encontrados na BDBComp dos anos 1987 a 2007. Atualmente, esta DL conta com cerca 11072 trabalhos publicados.

[1]Bibioteca Digital mantida pelo Laboratório de Banco de Dados da UFMG - http://www.lbd.dcc.ufmg.br/bdbcomp/

Tabela 5.1: Coleção BDBComp - Fonte: Carvalho et al. (2011)

Grupo Ambíguo	Número de Registros	Número de Autores distintos
A. Oliveira	52	16
A. Silva	64	32
F. Silva	26	20
J. Oliveira	48	18
J. Silva	36	17
J. Souza	35	11
L. Silva	33	18
M. Silva	21	16
R. Santos	20	16
R. Silva	26	20

Coleção KISTI

A coleção Kisti foi construída através da reunião de informações de diversas fontes. O objeto era construir uma base de testes com registros corrigidos a partir de informações investigadas em diversas bibliotecas digitais como: ArXiV, CiteSeer, CS BiBTeX, DBLP e NCSTRL. Os dados eram selecionados da DBLP[2], considerando as correções que se baseavam nas demais fontes. Foram selecionados registros de 1947 à 2007, dos 1000 autores mais frequentes. O conjunto de testes consiste em 41673 registros com ocorrência de 881 grupos de autores e 6921 autores. A coleção Kisti é conhecida como **KISTI-AD-E-01-TestSet**, que significa a primeira versão (01) do conjunto de teste, para o inglês (E -*English*), para desambiguação de autores (AD - *Author Dismbiguation*); foi criada pelo Instituto Coreano de Ciência e Tecnologia da Informação (Kang et al., 2011).

5.2 Configuração dos Experimentos

Como dito no início deste capítulo, os registros são separados em arquivos por ano. Dispondo desta característica, os experimentos foram conduzidos para analisar a eficiência dos métodos de acordo com o passar dos anos. Os experimentos consistem em definir duas situações em uma DL: a primeira em que a DL não possui nenhum registro cadastrado e todos os registros devem ser inseridos na etapa de desambiguação; a segunda em que a DL já possui registros cadastrados, livres de ambiguidade e com os grupos de autores devidamente classificados, restando apenas a tarefa de desambiguar os novos registros a serem inseridos.

Nesta última situação citada, foram realizados experimentos com diversas quantidades de registros, sendo adicionados a DL de acordo com o ano em uma ordem cronológica. Na Tabela 5.2, estão apresentadas as configurações dos experimentos para essa coleção. Na primeira

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

situação, a DL não possui nenhum registro de citação cadastrado; logo, todos os registros são inseridos juntos no processo de desambiguação. Esta consiste apenas no experimento 1, em que todos os 361 registros deverão ser inseridos. A segunda situação, compreende os experimentos de 2 à 19, nestes a quantidade de registros cadastrados na DL aumentam gradualmente de acordo com ano de publicação. Este fato, pode ser verificado nas colunas 2 e 3, que repectivamente, mostram a quantidade registros cadastrados e os anos destes registros.

Tabela 5.2: Configurações dos Experimentos - BDBComp

Experimento	Registros cadastrados	Anos dos Registros	Registros a serem cadastrados
Experimento 1	0	...	361
Experimento 2	2	1987	359
Experimento 3	5	1987 à 1989	356
Experimento 4	7	1987 à 1991	354
Experimento 5	10	1987 à 1992	351
Experimento 6	13	1987 à 1993	348
Experimento 7	14	1987 à 1994	347
Experimento 8	19	1987 à 1995	342
Experimento 9	28	1987 à 1996	333
Experimento 10	37	1987 à 1997	324
Experimento 11	54	1987 à 1998	307
Experimento 12	69	1987 à 1999	292
Experimento 13	91	1987 à 2000	270
Experimento 14	125	1987 à 2001	236
Experimento 15	168	1987 à 2002	193
Experimento 16	202	1987 à 2003	159
Experimento 17	272	1987 à 2004	89
Experimento 18	310	1987 à 2005	51
Experimento 19	343	1987 à 2006	18

Como dito, a cada experimento, o número de registros cadatrados na DL aumenta. Para ilustrar o estado da DL em um destes experimentos, é utilizado o Experimento 6 (vide Tabela 5.3), que possui 13 registros cadastrados, classificados em 11 grupos.

Tabela 5.3: Configuração de uma DL - Experimento 6 (Base BDBComp)

Grupo	Autor	Título	Véiculo de Publicação	Co-Autores	Ano
Grupo 1	jorge luiz e silva	communication protocol in a distributed system based on a centralized parallel bus	v Brazilian Symposium of computer networks	c kirner	1987
Grupo 2	r santos	packet switch system compact	xi Brazilian Symposium of computer graphics and image processing	t ohashi, t yoshida, t ejima	1987
Grupo 3	antonio mauro barbosa de oliveira	Earthworm Plus, a local network for didactic purposes	vii Brazilian Symposium of computer networks	j filho, j jr, m vieira	1989
	a mauro b oliveira	minhonix-a distributed system for teaching	ix Brazilian Symposium of computer networks	j peyrin	1991
	a mauro b oliveira	a computation platform for administration of ibcns	x Brazilian Symposium of computer networks	j sousa,m penna,j junior	1992

Continua na próxima página

Grupo	Autor	Título	Véiculo de Publicação	Co-Autores	Ano
Grupo 4	jose palazzo m de oliveira	journal of theoretical and applied informatics - Volume 1	journal of theoretical and applied computer science - volume 1	l lamb	1989
Grupo 5	r j f santos	Packet communication in ISDN pilot experience	v Brazilian Symposium of computer networks	j leite,c klemtz,m mandel, a mantovani, s cintra	1989
Grupo 6	marcos silva	specification of a system of supervision and management of data communication equipment	vi Brazilian Symposium on Geoinformatics	a monteiro,j medeiros	1991
Grupo 7	fernando antonio marques da silva	system for manipulation of images via electronic mail	x Brazilian Symposium of computer networks	p rodrigues	1992
Grupo 8	r m da silva	a prototype server modem over a network tcp / ip	xi Brazilian Symposium of computer graphics and image processing	s wu	1992

Continua na próxima página

Grupo	Autor	Título	Véiculo de Publicação	Co-Autores	Ano
Grupo 9	ana cristina b da silva	implementation of new agents to manage networks of computers	xi Brazilian Symposium of computer networks	c westephall	1993
Grupo 10	flavio m de silva	sisdi-osi, didactic system for the OSI model	xi Brazilian Symposium of computer networks	c fujito,e garcia	1993
Grupo 11	jose n souza	managing heterogeneous networks, integrator based approach	vii workshop testing and fault tolerance	r correia,a lages,l pirmez,l granville,e duarte-jr,r andrade	1993

Com relação aos parâmetros dos métodos, foram utilizados os valores que produziram os melhores resultados experimentalmente. Esses valores estão na Tabela 5.4, para cada coleção e método.

Tabela 5.4: Parâmetros dos Métodos

Base	INDi			Grupo Representativo		
	α_{Title}	α_{Venue}	δ	$\alpha_{coAuthors}$	α_{title}	α_{venue}
BDBComp	0.0	0.2	0.2	1	0.0	0.4
Kisti	0.0	0.0	0.2	1	0.0	0.4

Lembrando que os parâmetros α_{Title} e α_{Venue} do método INDi representam os limiares de similaridade entre os títulos dos trabalhos e dos títulos dos veículos de publicação, respectivamente, e δ é o valor incremental utilizado em uma etapa do processo de desambiguação.

No método Grupo Representativo, os parâmetros $\alpha_{coAuthors}$, α_{title} e α_{venue} significam respectivamente o número de co-autores do registro a ser inserido no grupo representativo que não pertençam aos registros do grupo representativo. O segundo, é a similaridade entre título do registro a ser inserido no grupo representativo e os títulos de trabalhos dos registros que pertecem ao grupo representativo. O último, é a similaridade entre o título do veículo de publicação do registro a ser inserido no grupo representativo e os títulos de veículos de publicação dos registros que pertencem ao grupo representativo.

5.3 Análise dos Experimentos

Nesta seção, são analisados os resultados dos experimentos, realizando comparações entre a utilização de registros representativos e com a utilização de todos os registros, no processo de desambiguação, sobre a mesma coleção. Os gráficos a seguir apresentam os valores das métricas de avaliação (métricas PMG, PMA e K) e a quantidade de registros representativos, selecionados pelo método proposto neste trabalho, em decorrer do aumento de registros na DL, aumento este baseado nos anos cadastrados na DL.

As Figuras 5.1 e 5.2 apresentam respectivamente os resultados dos métodos, Grupo Representativo (GR) e INDi aplicados sobre a coleção BDBComp. Neste, verifica-se que os valores de PMG, PMA e K aumentam de acordo com a quantidade de anos carregados na DL e ,consequentemente, o aumento na quantidade de registros livres de ambiguidade contidos nesta.

Nos experimentos realizados sobre esta base, pode ser verificado que os resultados das métricas citadas acima são semelhantes para os dois métodos. A exceção na semelhança dos resultados das métricas é visto entre 8 e 15 anos carregados. Os valores de PMA, pureza média do autor, sofrem um crescimento mais acentuado no método INDi e, com isso alavancando o crescimento de K, já que os valores de PMG, pureza média do grupo, permanecem semelhantes

nos dois métodos. Isto, acontece pois, como já visto, K é a média geométrica entre PMG e PMA. No método Grupo Representativo, a partir dos 15 anos carregados, os aumentos dos valores de PMA crescem nitidamente. A Tabela 5.5 compara os resultados da métrica k nos dois métodos, relacionando a porcentagem de registros representativos de GR com a dimuição da eficiência do processo de desambiguação.

Tabela 5.5: Comparativo Métrica K - BDBComp

Experimento	K - GR	K - INDi	Inferior (%)	Registros Representativos (%)
Experimento 1	0,872182177	0,873425365	0,14	80,33
Experimento 2	0,870747405	0,873425365	0,31	80,61
Experimento 3	0,871311215	0,873425365	0,24	80,61
Experimento 4	0,869994382	0,874579251	0,52	80,06
Experimento 5	0,872668902	0,875774995	0,35	80,06
Experimento 6	0,869448788	0,875774995	0,72	80,06
Experimento 7	0,872777828	0,879044456	0,71	79,50
Experimento 8	0,871893942	0,880390217	0,97	79,22
Experimento 9	0,875248602	0,882912434	0,87	79,22
Experimento 10	0,875544684	0,890831028	1,72	78,95
Experimento 11	0,885234777	0,901927204	1,85	77,84
Experimento 12	0,889424587	0,903461546	1,55	77,29
Experimento 13	0,896742331	0,920464085	2,58	75,62
Experimento 14	0,899319291	0,936288917	**3,95**	73,41
Experimento 15	0,905102476	0,941117773	3,83	72,30
Experimento 16	0,91716551	0,949233723	3,38	69,53
Experimento 17	0,951208693	0,973793744	2,32	64,27
Experimento 18	0,978925046	0,988839634	1,00	61,22
Experimento 19	0,99182119	0,994212244	0,24	58,17

Analisando todos experimentos dos dois métodos de remoção de ambiguidade de nomes de autores sobre a base BDBComp, pode-se comprovar uma leve diferença nos resultados de algumas de suas métricas, sendo elas PMA e K, na qual, o método INDi possui melhores resultados. A maior diferença entre os valores da métrica foi detectado no experimento 14, que pode ser visto na Tabela 5.5. Neste, usando 73,41% dos registros como representativo, obteve-se um resultado apenas 3,95% inferior ao uso de todos os registros.

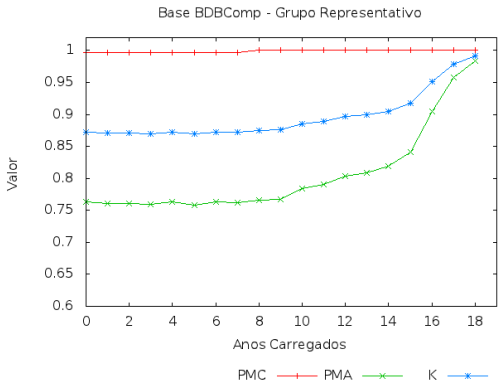

Figura 5.1: Base BDBComp - GR

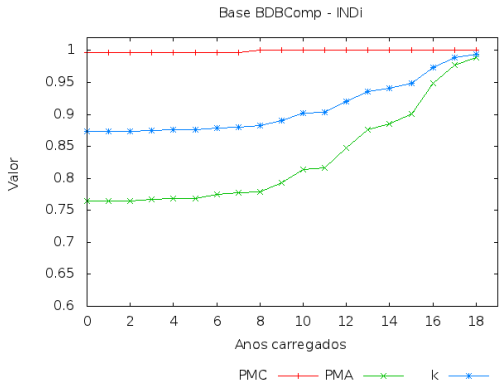

Figura 5.2: Base BDBComp - INDi

As Figuras 5.3 e 5.4 apresentam respectivamente os resultados dos métodos Grupo Representativo e INDi aplicados sobre a coleção Kisti. Analisando os dois gráficos, é nítido que entre os 0 e 40 anos carregados na DL, os resultados das métricas possuem pequena variação; em alguns casos, os valores oscilam entre crescentes e decrescentes.

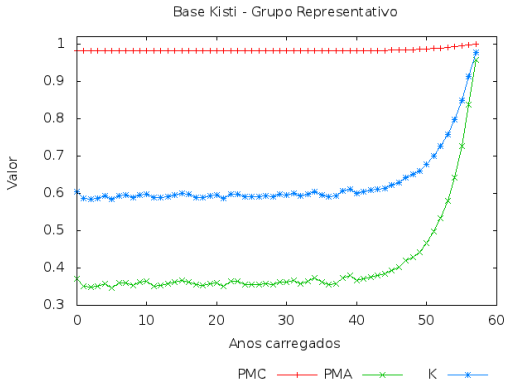

Figura 5.3: Base Kisti - GR

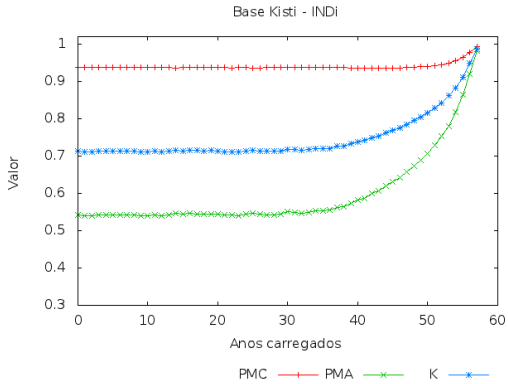

Figura 5.4: Base Kisti - INDi

No método Grupo Representativo, os valores das métricas PMA e K possuem pequenas oscilação até os 40 anos carregados na DL. A partir deste, os valores das métricas citadas crescem rapidamente. Os valores da métrica PMG permanecem praticamente constantes durante todos os experimentos.

O método INDi possui os valores de PMA e K bem superiores ao do Grupo Representativo (GR); no entanto, assim como ocorreu no GR, essas métricas possuem pequenas oscilações até os 40 anos carregados na DL e, a partir deste, os valores também crescem rapidamente. Os valores de PMG de INDi são levemente inferiores ao de GR e também permanecem praticamente constantes durante os experimentos. Na Tabela 5.6 é apresentado um comparativo dos resultados da métrica k nos métodos, aplicados sobre a coleção Kisti.

Tabela 5.6: Comparativo Métrica K - Kisti

Experimento	K - GR	K - INDi	Inferior (%)	Registros Representativos (%)
Experimento 1	0,603774863	0,713121787	15,33	61,46
Experimento 2	0,586941182	0,711959742	17,56	61,45
Experimento 3	0,584706405	0,711959742	17,87	61,46
Experimento 4	0,586817629	0,712591431	17,65	61,49
Experimento 5	0,5927251	0,712591431	16,82	61,62
Experimento 6	0,584360017	0,712591431	18,00	61,51
Experimento 7	0,594303802	0,712591431	16,60	61,64
Experimento 8	0,594892825	0,712591431	16,52	61,54
Experimento 9	0,589228669	0,712591431	17,31	61,42
Experimento 10	0,596655723	0,711719633	16,17	61,51
Experimento 11	0,598083365	0,711719633	15,97	61,54
Experimento 12	0,587885638	0,713267698	17,58	61,51
Experimento 13	0,589660804	0,711913824	17,17	61,47
Experimento 14	0,592164022	0,713433452	17,00	61,60
Experimento 15	0,596409692	0,714820578	16,57	61,58
Experimento 16	0,600121108	0,714160572	15,97	61,67
Experimento 17	0,596696159	0,715380325	16,59	61,47
Experimento 18	0,589944777	0,714799222	17,47	61,49
Experimento 19	0,588477181	0,713827989	17,56	61,50
Experimento 20	0,59324179	0,714689924	16,99	61,58
Experimento 21	0,594532236	0,714403725	16,78	61,60
Experimento 22	0,587381277	0,712149617	17,52	61,51
Experimento 23	0,597662282	0,711889149	16,05	61,61
Experimento 24	0,597882121	0,711494186	15,97	61,48
Experimento 25	0,591262826	0,713771842	17,16	61,59
Experimento 26	0,591419614	0,715884066	17,39	61,55
Experimento 27	0,590427578	0,714319499	17,34	61,53
Experimento 28	0,592628525	0,712459052	16,82	61,44
Experimento 29	0,591132249	0,713245284	17,12	61,41
Experimento 30	0,597273375	0,714126186	16,36	61,28
Experimento 31	0,595668927	0,718071746	17,05	61,42
Experimento 32	0,59963125	0,717626583	16,44	61,35
Experimento 33	0,592923712	0,716154378	17,21	61,23

Continua na próxima página

Experimento	K - GR	K - INDi	Inferior (%)	Registros Representativos (%)
Experimento 34	0,597701456	0,71769059	16,72	61,23
Experimento 35	0,604687007	0,720649663	16,09	61,10
Experimento 36	0,596429579	0,719469376	17,10	60,99
Experimento 37	0,591199857	0,720958598	18,00	60,99
Experimento 38	0,593475471	0,725710455	18,22	60,72
Experimento 39	0,606036749	0,727367951	16,68	60,68
Experimento 40	0,611630149	0,732664247	16,52	60,49
Experimento 41	0,600256872	0,738459142	18,71	60,10
Experimento 42	0,604549941	0,741163835	18,43	60,05
Experimento 43	0,608202404	0,749961259	18,90	59,54
Experimento 44	0,611445752	0,7538967	18,90	58,85
Experimento 45	0,614201454	0,761234131	**19,32**	58,18
Experimento 46	0,6218401	0,768714753	19,11	57,50
Experimento 47	0,628940416	0,776004224	18,95	56,40
Experimento 48	0,642546339	0,785137439	18,16	55,46
Experimento 49	0,650212558	0,794534985	18,16	53,84
Experimento 50	0,66012398	0,804231324	17,92	52,08
Experimento 51	0,678400292	0,815374178	16,80	50,17
Experimento 52	0,70056901	0,828098188	15,40	48,04
Experimento 53	0,726469914	0,842976203	13,82	45,74
Experimento 54	0,757790049	0,861266624	12,01	42,47
Experimento 55	0,798372486	0,882965404	9,58	38,54
Experimento 56	0,849650375	0,912063956	6,84	33,02
Experimento 57	0,913886311	0,948368898	3,64	26,36
Experimento 58	0,978003258	0,987688656	0,98	19,16

Analisando todos os experimentos dos dois métodos sobre a coleção Kisti, é evidente que os resultados dos experimentos do método INDi são bem superiores ao do Grupo Representativo. Nos dois métodos, as métricas sofreram pouco crescimento até os 40 anos carregados na DL. Isso se deve a quantidade de registros já pertencentes a DL. Na tabela 6.1 de experimentos da coleção Kisti, pode-se verificar que a maior parte dos registros da coleção não pertencem a DL, e serão inseridos no processo de desambiguação, pois quanto maior o número de registros já pertencentes a DL melhor para ambos os métodos. A maior diferença entre os valores da métrica K foi detectada no experimento 45, que pode ser visto na Tabela 5.6. Neste, usando 58,18% dos registros como representativo obteve-se um resultado 19,32% inferior ao uso de todos os registros.

As Figuras 5.5 e 5.6 apresentam respectivamente a quantidade de registros representativos

das coleções BDBComp e Kisti de acordo com quantidade de anos carregados em cada DL. Em ambos os gráficos, é verificado que a quantidade de registros representativos decresce de acordo com o aumento dos anos carregados na DL. Isso ocorre pois o PMA aumenta de acordo com o crescimento do número de registros na DL e o PMC tem um crescimento sútil. Estes dois fatores contribuem para que a quantidade de grupos de autores diminua e, com isso, a quantidade de registros representativos também diminua. No entanto, mesmo com essa diminuição o número de registros representativos na coleção BDBComp é alto se comparado com o total de registros presentes nesta. Isso se deve ao fato desta coleção possuir uma média baixa de registros por autor (média de 2 registros por autor). Na coleção Kisti, a quantidade de registros representativos no últimos experimentos é menor se comparada a quantidade total de registros, contrapondo o que ocorre na coleção BDBComp.

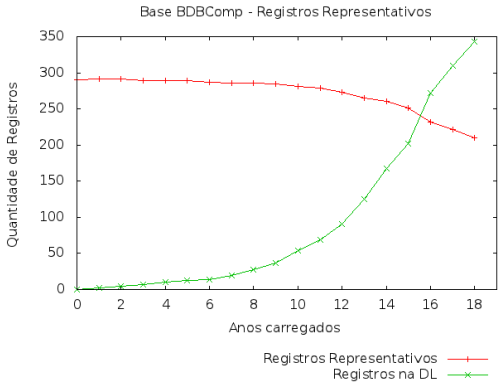

Figura 5.5: Base BDBComp - Registros Representativos

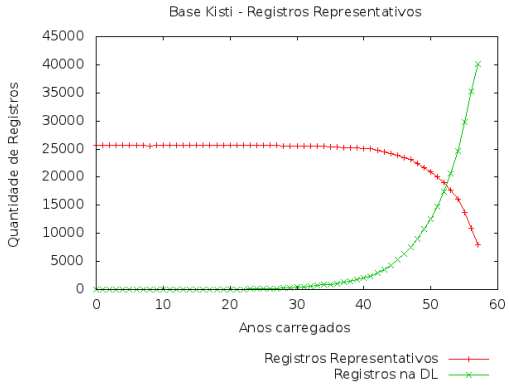

Figura 5.6: Base Kisti - Registros Representativos

Um comparativo diferente do apresentado, é relacionar o número de registros representativos com a pureza média dos autores (PMA) de acordo com o número de anos carregados na DL. Conforme o PMA cresce, a quantidade de registros representativos decresce, pois o PMA crescente significa que a fragmentação dos registros dos autores está diminuindo e o número de registros dos autores aumenta. Logo, o número de registros para representar um autor é menor do que para representar n autores, visto que o número de registros representativos está atrelado ao número de registros de um autor.

Capítulo 6

Conclusões

Neste trabalho, foi proposto um método para identificação de um conjunto de registros representativos em grupos de registros. A finalidade é diminuir o número de registros utilizados no processo de remoção de ambiguidade de nomes de autores, obtendo resultados competitivos quando comparados ao uso de todos os registros. Diversos métodos podem ser adaptados para utilizar este conjunto de registros representativos no processo de identificação dos autores. Neste trabalho, adaptamos o método INDi.

O objeto desta técnica é tornar o processo de identificação de um autor em uma biblioteca digital mais sucinto, utilizando somente os registros mais pertinentes a este propósito. Foram utilizadas duas coleções para avaliar o método. Em ambas, a redução do número de registros no processo de desambiguação causou uma pequena dimuição da eficácia do método. Na coleção extraída da BDBComp, usando 73,41% dos registros como representativos, obteve-se um resultado apenas 3,95% inferior ao uso de todos os registros. Na coleção extraída da DBLP, usando apenas 58,18% dos registros como representativos obteve-se um resultado 19,32% inferior ao uso de todos os registros. Na primeira coleção, a redução da eficácia foi pequena, mas a redução da quantidade de registros representativos utilizados no processo não foi muito significativo. Na segunda, a redução da quantidade de registros foi significativa, no entanto isso afetou a eficácia do método. A diferença na quantidade de registros utilizados nas duas coleções se deve principalmente a diferença nas médias de registros por autor.

Como trabalho futuro, pretende-se investigar outras maneiras de selecionar registros representativos e aplicá-los a outros processos de desambiguação, bem como realizar experimentos com outras bases.

Apêndice

A tabela abaixo apresenta a configuração dos experimentos realizados sobre a coleção Kisti. Uma tabela semelhante sobre a coleção BDBComp é apresentada no capítulo 5.

Tabela 6.1: Experimentos - Base Kisti

Experimento	Citações cadastradas	Anos das Citações	Citações a serem cadastradas
Experimento 1	0	...	41672
Experimento 2	1	1947	41671
Experimento 3	2	1947 à 1948	41670
Experimento 4	3	1947 à 1949	41669
Experimento 5	6	1947 à 1950	41666
Experimento 6	7	1947 à 1952	41665
Experimento 7	8	1947 à 1953	41664
Experimento 8	9	1947 à 1954	41663
Experimento 9	10	1947 à 1955	41662
Experimento 10	13	1947 à 1957	41659
Experimento 11	15	1947 à 1958	41657
Experimento 12	16	1947 à 1960	41656
Experimento 13	17	1947 à 1961	41655
Experimento 14	19	1947 à 1962	41653
Experimento 15	23	1947 à 1963	41649
Experimento 16	25	1947 à 1964	41647
Experimento 17	27	1947 à 1965	41645
Experimento 18	28	1947 à 1966	41644
Experimento 19	33	1947 à 1967	41639
Experimento 20	39	1947 à 1968	41633
Experimento 21	44	1947 à 1969	41628
Experimento 22	55	1947 à 1970	41617
Experimento 23	71	1947 à 1971	41601

Continua na próxima página

Experimento	Citações cadastradas	Anos das Citações	Citações a serem cadastradas
Experimento 24	94	1947 à 1972	41578
Experimento 25	113	1947 à 1973	41559
Experimento 26	134	1947 à 1974	41538
Experimento 27	165	1947 à 1975	41507
Experimento 28	209	1947 à 1976	41463
Experimento 29	260	1947 à 1977	41412
Experimento 30	328	1947 à 1978	41344
Experimento 31	396	1947 à 1979	41276
Experimento 32	467	1947 à 1980	41205
Experimento 33	574	1947 à 1981	41098
Experimento 34	684	1947 à 1982	40988
Experimento 35	813	1947 à 1983	40859
Experimento 36	945	1947 à 1984	40727
Experimento 37	1094	1947 à 1985	40578
Experimento 38	1287	1947 à 1986	40385
Experimento 39	1469	1947 à 1987	40203
Experimento 40	1719	1947 à 1988	39953
Experimento 41	2042	1947 à 1989	39630
Experimento 42	2425	1947 à 1990	39247
Experimento 43	2939	1947 à 1991	38733
Experimento 44	3517	1947 à 1992	38155
Experimento 45	4325	1947 à 1993	37347
Experimento 46	5277	1947 à 1994	36395
Experimento 47	6325	1947 à 1995	35347
Experimento 48	7516	1947 à 1996	34156
Experimento 49	8965	1947 à 1997	32707
Experimento 50	10728	1947 à 1998	30944
Experimento 51	12582	1947 à 1999	29090
Experimento 52	14810	1947 à 2000	26862
Experimento 53	17350	1947 à 2001	24322
Experimento 54	20647	1947 à 2002	21025
Experimento 55	24637	1947 à 2003	17035
Experimento 56	29732	1947 à 2004	11940
Experimento 57	35261	1947 à 2005	6411
Experimento 58	40154	1947 à 2006	1518

Referências Bibliográficas

Bodó, Z.; Minier, Z. e Csató, L. (2011). Active learning with clustering. *Journal of Machine Learning Research - Proceedings Track*, 16:127–139.

Carvalho, A. P.; Ferreira, A. A.; Laender, A. H. F. e Gonçalves, M. A. (2011). Incremental unsupervised name disambiguation in cleaned digital libraries. *Journal of Information and Data Management*, 2(3):289–304.

Cohen, W. W.; Ravikumar, P. D. e Fienberg, S. E. (2003). A comparison of string distance metrics for name-matching tasks. In *Information Integration on the Web*, pp. 73–78, Acapulco, Mexico.

Cota, R. G.; Ferreira, A. A.; Nascimento, C.; Gonçalves, M. A. e Laender, A. H. F. (2010). An unsupervised heuristic-based hierarchical method for name disambiguation in bibliographic citations. *Journal of the American Society for Information Science and Technology*, 61(9):1853–1870.

Ferreira, A. A.; Gonçalves, M. A. e Laender, A. H. F. (2012). A brief survey of automatic methods for author name disambiguation. *SIGMOD Record*, 41(2):15–26.

Ferreira, A. A.; Veloso, A.; Gonçalves, M. A. e Laender, A. H. F. (2010). Effective self-training author name disambiguation in scholarly digital libraries. In *Proceedings of the 10th annual joint conference on Digital libraries*, JCDL '10, pp. 39–48, Gold Coast, Queensland, Australia.

French, J. C.; Powell, A. L. e Schulman, E. (2000). Using clustering strategies for creating authority files. *Journal of the American Society for Information Science and Technology*, 51(8):774–786.

Gonçalves, M. A.; Fox, E. A.; Watson, L. T. e Kipp, N. A. (2004). Streams, structures, spaces, scenarios, societies (5s): A formal model for digital libraries. *ACM Transactions on Information Systems*, 22(2):270–312.

Han, H.; Giles, L.; Zha, H.; Li, C. e Tsioutsiouliklis, K. (2004). Two supervised learning approaches for name disambiguation in author citations. In *Joint Conference on Digital Libraries*, pp. 296–305, Tucson, AZ, USA. ACM.

Han, H.; Zha, H. e Giles, C. L. (2005). Name disambiguation in author citations using a k-way spectral clustering method. In *Joint Conference on Digital Libraries*, JCDL '05, pp. 334–343, Denver, CO, USA. ACM.

Huang, J.; Ertekin, S. e Giles, C. L. (2006). Efficient name disambiguation for large-scale databases. In *Proceedings of the 10th European conference on Principle and Practice of Knowledge Discovery in Databases*, PKDD'06, pp. 536–544, Berlin, Heidelberg. Springer-Verlag.

Kang, I.-S.; Kim, P.; Lee, S.; Jung, H. e You, B.-J. (2011). Construction of a large-scale test set for author disambiguation. *Information Processing Management*, 47(3):452–465.

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

Lee, D.; On, B.-W.; Kang, J. e 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*, IQIS '05, pp. 69–76, Baltimore, Maryland, USA. ACM.

Levenshtein, V. I. (1966). Binary codes capable of correcting deletions, insertions, and reversals. Technical Report 8.

Nguyen, H. T. e Smeulders, A. (2004). Active learning using pre-clustering. In *Proceedings of the twenty-first international conference on Machine learning*, ICML '04, pp. 623–630, Banff, Alberta, Canada. ACM.

Oliveira, J. W. A.; Laender, A. H. F. e Gonçalves, M. A. (2005). Remoção de ambiguidades na identificação de autoria de objetos bibliográficos. In *SBBD*, pp. 205–219, UberlÃndia, MG, Brazil.

Porter, M. F. (1980). An Algorithm for Suffix Stripping. *Program*, 14(3):130–137.

Salton, G.; Wong, A. e Yang, C. S. (1975). A vector space model for automatic indexing. *Communications of the ACM*, 18:613–620.

Xiaoming, F.; Jianyong, W.; Xu, P.; Lizhu, Z. e Bing, L. (2011). On graph-based name disambiguation. *ACM Journal Data and Information Quality*, 2:10:1–10:23.

Zhao, W.; He, Q.; Ma, H. e Shi, Z. (2012). Effective semi-supervised document clustering via active learning with instance-level constraints. *Knowledge and Information Systems*, 30(3):569–587.