# HEURÍSTICAS PARA DESAMBIGUAÇÃO INCREMENTAL DE NOMES DE AUTORES EM REFERÊNCIAS BIBLIOGRÁFICAS

ALAN FILIPE SANTANA

# HEURÍSTICAS PARA DESAMBIGUAÇÃO INCREMENTAL DE NOMES DE AUTORES EM REFERÊNCIAS BIBLIOGRÁFICAS

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

Orientador: Marcos André Gonçalves

Belo Horizonte

Abril de 2015

Santana, Alan Filipe

S232a Heurísticas para desambiguação incremental de nomes de autores em referências bibliográficas / Alan Filipe Santana. — Belo Horizonte, 2015
xviii, 73 f. : il. ; 29cm

Dissertação (mestrado) — Universidade Federal de Minas Gerais - Departamento de Ciência da Computação
Orientador: Marcos André Gonçalves

1. Computação - Teses. 2. Heurísticas - Teses. 3. Referências bibliográficas - Teses. 4. Bibliotecas digitais - Teses. 5. Ambiguidade - Teses. I. Orientador. II. Título.

CDU 519.6*73(043)

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

# FOLHA DE APROVAÇÃO

Heurísticas para desambiguação incremental de nomes de autores em referências bibliográficas

**ALAN FILIPE SANTANA**

Dissertação defendida e aprovada pela banca examinadora constituída pelos Senhores:

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

PROF. ANDERSON ALMEIDA FERREIRA
Departamento de Computação - UFOP

PROF. CLODOVEU AUGUSTO DAVIS JÚNIOR
Departamento de Ciência da Computação - UFMG

PROF. RICARDO DA SILVA TORRES
Instituto de Computação - UNICAMP

Belo Horizonte, 02 de julho de 2015.

# Agradecimentos

Primeiramente, gostaria de agradecer à minha família por todo apoio, amor e confiança dado durante essa e todas as etapas da minha vida.

À minha companheira, Evanise, por todo carinho e compreensão dados durante meu mestrado e graduação.

Ao meu orientador, Marcos André Gonçalves, pela orientação, paciência e ensinamentos que foram essenciais para o desenvolvimento deste trabalho.

Aos professores Anderson Ferreira e Alberto Laender pela colaboração durante o desenvolvimento dos trabalhos. A todos os professores do departamento pelos conhecimentos adquiridos.

À Universidade Federal de Minas Gerais e ao Departamento de Ciência da Computação, pela oportunidade de realização do curso de Pós-Graduação.

À CAPES, pelo apoio financeiro concedido durante o primeiro ano do mestrado, sem o qual não seria possível o ingresso à UFMG. A todos do Laboratório de Banco de Dados, pela oportunidade de fazer parte desta incrível e acolhedora equipe de pesquisa.

Aos meus colegas de trabalho e à Universidade Federal de Lavras pelo suporte dado para que fosse possível a conclusão deste curso.

# Resumo

A ambiguidade de nomes de autores em referências bibliográficas é um dos principais problemas que afetam a qualidade dos serviços oferecidos pelas bibliotecas digitais. Nos últimos anos, inúmeros métodos automáticos de desambiguação de nomes em referências bibliográficas foram propostos, baseados em diferentes abordagens supervisionadas e não supervisionadas. Entretanto, poucos foram desenvolvidos com o objetivo de permitir a desambiguação das referências no momento em que elas são incluídas no repositório de uma biblioteca digital. Em um cenário real, estas soluções são potencialmente mais práticas e eficientes, uma vez que evitam a necessidade de reprocessar todo repositório sempre que novas citações são incluídas no repositório. Neste trabalho, é proposto um novo método de desambiguação incremental baseado em heurísticas, capaz de criar e atualizar automaticamente um conjunto de treinamento utilizado para determinar os autores de cada referência. Este método foi avaliado em diferentes cenários de aplicação e comparado com várias soluções encontradas na literatura. Na avaliação experimental, foram obtidos ganhos significativos em todas as coleções utilizadas em relação aos melhores *baselines* supervisionados e não-supervisionados. Também foram realizados experimentos a fim de demonstrar a praticidade e eficiência do método ao realizar a desambiguação de coleções de forma incremental.

**Palavras-chave:** Heurísticas, Ambiguidade de Nomes, Referências Bibliográficas, Bibliotecas Digitais.

# Abstract

The ambiguity of author name in bibliographic references is one of the main problems affecting the quality of services offered by digital libraries. In recent years, numerous methods for automatic name disambiguation have been proposed, based on different supervised and unsupervised approaches. However, just a few have been developed in order to allow the disambiguation at the time that citations are incorporated into the digital library. In a real situation, these solutions are potentially more efficient and practical, since they avoid the need to reprocess the entire repository whenever new citations are included in the database. This paper proposes a new incremental disambiguation method based on heuristics, able to automatically create and update a training set used to determine the author of each reference. This method was evaluated in different application scenarios and compared with several solutions found in the literature. In the experimental evaluation, our solution has achieved significant gains in all collections when compared with the best supervised and unsupervised baselines. We also performed experiments to demonstrate the practicability and efficiency of the method when used in a incremental way.

**Keywords:** Heuristics, Name Ambiguity, Bibliographic References, Digital Libraries.

# Lista de Figuras

# Lista de Tabelas

# Sumário

# Capítulo 1

# Introdução

Nos últimos anos é possível observar um aumento exponencial da produção global de publicações científicas [Bornmann & Mutz, 2014]. Com o desenvolvimento da *World Wide Web* (WWW), grande parte destas publicações ficam acessíveis através da Internet. Nesse cenário, Bibliotecas Digitais (BD), como DBLP[1], CiteSeer[2], MEDLINE[3] e BDBComp[4], possuem grande importância ao prover serviços que facilitam o acesso a publicações relevantes pela comunidade acadêmica, além de possibilitar pesquisas e análises relacionadas com redes de colaboração, tendências, cobertura de tópicos e impacto de publicações.

Bibliotecas Digitais são consideradas complexos sistemas de informação que reúnem uma rica coleção de metadados organizados em uma estrutura interna a fim de suportar serviços como indexação, navegação, busca e personalização [Gonçalves et al., 2004]. A qualidade dos serviços e conteúdos oferecidos por uma BD é central para o sucesso do sistema [Laender et al., 2008]. Neste contexto, a ambiguidade de nomes de autores é um problema que prejudica a qualidade dos serviços de recuperação de informação bibliográfica fornecidos pelas BDs. Este problema ocorre quando dois ou mais autores compartilham um mesmo nome (nomes homônimos) ou quando um autor utiliza diferentes nomes em suas referências bibliográficas (nomes sinônimos). A Tabela 1.1 mostra exemplos de referências homônimas e sinônimas encontradas na DBLP. Nas duas primeiras linhas são apresentadas citações que possuem a referência *A. K. Gupta*, entretanto a primeira refere-se ao pesquisador *Amit Kumar Gupta* enquanto que a segunda refere-se ao pesquisador *Arun Kumar Gupta*. Já as citações apresentadas nas três últimas linhas possuem referências distintas a um mesmo autor chamado

[1]http://dblp.uni-trier.de/ (acesso em: 27 de julho de 2015).
[2]http://citeseer.ist.psu.edu/ (acesso em: 27 de julho de 2015).
[3]http://www.ncbi.nlm.nih.gov/pubmed/ (acesso em: 27 de julho de 2015).
[4]http://www.lbd.dcc.ufmg.br/bdbcomp/ (acesso em: 27 de julho de 2015).

Tabela 1.1: Exemplos de referências ambíguas encontrados na DBLP

| **Problema** | **Nome ambíguo** | **Referência bibliográfica** |
|---|---|---|
| Nomes homônimos | A. K. Gupta | A. K. Gupta, K. Nagaraj, T. R. Viswanathan: *A Two-Stage ADC Architecture With VCO-Based Second Stage.* IEEE Trans. on Circuits and Systems 58-II(11): 734-738 (2011) |
| | A. K. Gupta | Dancy Kurian, R. S. Jayasree, M. Wilscy, A. K. Gupta: *White Matter Hyperintensity segmentation using multiple stage FCM.* A2CWiC 2010: 36 |
| Nomes sinônimos | John A. Robinson | John A. Robinson: *Efficient Gaussian filtering using Cascaded Prefix Sums.* ICIP 2012: 117-120 |
| | J. A. Robinson | J. A. Robinson: *Adaptive prediction trees for image compression.* IEEE Transactions on Image Processing 15(8): 2131-2145 (2006) |
| | John Robinson | Mark Blythe, John Robinson, David Frohlich: *Interaction design and the critics: what to make of the "weegie".* NordiCHI 2008: 53-62 |

*John Allen Jobinson.*

A ambiguidade de nomes também impacta consideravelmente análises bibliométricas [Strotmann & Zhao, 2012], integração de bancos de dados [Kalashnikov & Mehrotra, 2006] e análise de redes sociais [Strotmann et al., 2009; Fegley & Torvik, 2013]. Nas BDs, as causas deste problema incluem a forma dinâmica e descentralizada de obtenção de conteúdo, falta de padronização de formatos e processos compartilhados pelas fontes dos metadados [Laender et al., 2008], mudanças de nomes, múltiplas transliterações de nomes não romanos e erros tipográficos.

Os desafios ao lidar com este problema tem levado ao desenvolvimento de inúmeros métodos de desambiguação [Klaas, 2007; Smalheiser & Torvik, 2009; Ferreira et al., 2012b]. Dentre as estratégias possíveis, existem atribuições manuais, desenvolvimento de identificadores únicos globais e métodos automáticos. A primeira estratégia possui um alto custo por exigir grande esforço humano e pode ser representado, por exemplo, pelo trabalho realizado por bibliotecários [Scoville et al., 2003]. A segunda estratégia consiste em construir um registro global de pesquisadores e associá-los a identificadores únicos. Exemplos de serviços de associação de identificadores a pesquisadores incluem ResearcherID[5], Open Researcher and Contributor ID (ORCID)[6] e, mais recentemente, Researcher Name Resolver (RNR)[7] [Kurakawa et al., 2014]. Embora estas estratégias exijam menos esforço humano, elas ainda dependem da colaboração voluntária e ativa de pesquisadores e autores, o que pode ser improvável de se conseguir dentro de um

[5]http://www.researcherid.com (acesso em: 27 de julho de 2015).
[6]http://orcid.org (acesso em: 27 de julho de 2015).
[7]http://rns.nii.ac.jp/ (acesso em: 27 de julho de 2015).

intervalo de tempo curto e em âmbito global [Smalheiser & Torvik, 2009]. Portanto, nos últimos anos, uma grande atenção tem sido dada ao desenvolvimento de métodos automáticos de desambiguação para serem aplicados em BDs [Ferreira et al., 2012b].

Métodos automáticos de desambiguação normalmente tentam resolver o problema agrupando referências de um mesmo autor baseado em medidas de similaridades entre seus atributos ou, diretamente, associando uma citação a um determinado autor. Ambos os tipos de métodos podem seguir abordagens supervisionadas ou não-supervisionadas. Historicamente, métodos supervisionados têm, empiricamente, produzidos os melhores resultados [Ferreira et al., 2012b]. Entretanto, ao depender de dados de treinamento, esses métodos podem não ser adequados em situações reais nas quais novos nomes ambíguos, ausentes no conjunto de treinamento, aparecem todo o tempo, além de desconsiderar as mudanças que comumente ocorrem nos perfis de publicação dos autores. Mesmos os métodos de desambiguação baseados em técnicas tradicionais de clusterização não são práticos ao considerar as características de uma BD real [Carvalho et al., 2011].

A fim de lidar com os desafios relativos a evolução das BDs, alguns métodos utilizaram heurísticas, combinando-as com abordagens supervisionadas [Veloso et al., 2012; Ferreira et al., 2014] ou com o objetivo de se obter métodos incrementais de desambiguação [Carvalho et al., 2011; Esperidião et al., 2014]. Estes últimos visam desambiguar as referências no momento em que elas são inseridas na BD, considerando um conjunto de regras ou heurísticas para identificar a presença de um novo autor e a fragmentação de grupos de citações (ou *clusters*), portanto, são potencialmente mais eficientes. Recentemente, também foram propostos métodos baseados em *relevance feedback* que, a partir de um pequeno esforço do administrador (ou usuários) de uma BD, são capazes de aumentar significativamente a qualidade do desambiguador [Ferreira et al., 2012c; Godoi et al., 2013; Li et al., 2014].

Neste trabalho, é proposto um novo método incremental de desambiguação, que combina diversas heurísticas específicas do domínio, a fim de obter uma técnica eficiente e efetiva. O método proposto utiliza novas funções de similaridade para identificar a ocorrência de novos autores e *clusters* fragmentados que representam um mesmo autor. Para aumentar a praticidade do método em diferentes cenários, foram desenvolvidos procedimentos para a definição automática dos valores dos seus parâmetros com ou sem um conjunto de treinamento. Também foi avaliada a utilização de características baseadas na coocorrência de palavras, de forma similar ao realizado por Figueiredo et al. [2011] na tarefa de classificação de texto. A seguir são detalhados os objetivos e as contribuições do trabalho e a organização deste texto.

## 1.1 Objetivos

O principal objetivo deste trabalho é o desenvolvimento de um novo método de desambiguação, tendo em vista os desafios em aberto listados no estudo de Ferreira et al. [2012b], relacionados com as seguintes características: a presença de poucas informações nas citações (ou referências bibliográficas), tolerância a erros, eficiência (baixa complexidade de tempo), tolerância a mudanças nos perfis de publicação dos autores, inclusão de novos autores e desambiguação incremental. Para isto, os seguintes objetivos específicos foram vislumbrados:

- Desenvolver uma nova função de similaridade entre autores e referências considerando os atributos mais comumente encontrados nas citações bibliográficas.
- Utilizar heurísticas para possibilitar a identificação de novos autores no momento em que uma nova citação é incluída no repositório de uma BD.
- Utilizar heurísticas para identificar e corrigir possíveis erros de classificação gerados durante o processo de desambiguação.
- Utilizar heurísticas para identificar *clusters* fragmentados, ou seja, grupos de referências que pertencem a um mesmo autor.
- Avaliar a utilização de características baseadas em coocorrência de termos no cálculo da similaridade entre autores e referências.
- Comparar o método proposto com métodos supervisionados no estado-da-arte.
- Comparar o método proposto com métodos não-supervisionados no estado-da-arte.
- Avaliar o método proposto em cenários simulando a evolução de bibliotecas digitais durante um período de tempo.

## 1.2 Principais Contribuições

As principais contribuições deste trabalho são:

- O desenvolvimento de um novo método de desambiguação incremental altamente eficiente e efetivo, capaz de ser utilizado de forma supervisionada ou não.

- Análise de uma estratégia de incorporação de características baseadas em co-ocorrência de palavras na resolução da tarefa de desambiguação de nomes de autores.

- Avaliação do método proposto utilizando coleções reais extraídas a partir da DBLP e BDBComp e comparação dos resultados com vários métodos supervisionados e não-supervisionados encontrados na literatura.

- Avaliação do método proposto em cenários que simulam a evolução de bibliotecas digitais durante um determinado período de tempo, utilizando coleções reais e sintéticas, e comparação dos resultados com dois métodos incrementais encontrados na literatura.

Uma parte dos trabalhos desenvolvidos foi publicada e apresentada na *Joint Conference on Digital Libraries* (JCDL) [Santana et al., 2014]. Uma versão estendida desse trabalho foi convidada e aceita para publicação, após nova revisão, no periódico *International Journal on Digital Libraries* (IJDL) [Santana et al., 2015] numa edição especial dos melhores artigos da conferência. Além dessas publicações, um novo artigo abordando a versão mais recente do método proposto foi submetido ao periódico *Journal of the Association for Information Science and Technology* (JASIST) e se encontra sob avaliação.

## 1.3 Organização do Trabalho

Este documento está organizado da seguinte forma: no Capítulo 2, é apresentada uma definição formal da tarefa de desambiguação de nomes em referências bibliográficas, seguido de um resumo dos principais trabalhos relacionados. O Capítulo 3 descreve o método proposto, os procedimentos que podem ser utilizados para estimar seus parâmetros e a análise de complexidade de tempo do seu algoritmo. O Capítulo 4 detalha a metodologia de avaliação utilizada e discute os resultados alcançados. Por fim, o Capítulo 5 conclui este trabalho incluindo perspectivas para pesquisas futuras.

# Capítulo 2

# Revisão Bibliográfica

Este capítulo apresenta um resumo dos principais trabalhos relacionados encontrados na literatura. Na Seção 2.1, é apresentada uma formalização do problema de desambiguação de nomes em referências bibliográficas, seguida de uma breve revisão bibliográfica, considerando a taxonomia proposta por Ferreira et al. [2012b], além da discussão de duas novas categorias de métodos: baseadas em *relevance feedback* e métodos incrementais. Por fim, a Seção 2.2 apresenta a estratégia de extração de características baseadas em coocorrências de palavras, utilizada na tarefa de classificação de texto, proposta por Figueiredo et al. [2011], que pode ser utilizada também na tarefa de desambiguação de nomes de autores em referência bibliográficas.

## 2.1 Desambiguação de Nomes de Autores

Uma referência bibliográfica, ou citação, consiste em um conjunto de atributos como autor, coautores, título, local de publicação e ano, que descreve uma obra escrita. Citações são componentes essenciais das BDs, uma vez que permitem o acesso às publicações que identificam. A tarefa de desambiguação de nomes de autores em citações pode ser formulada da seguinte forma: seja $\mathcal{C} = \{c_1, c_2, ..., c_k\}$ um conjunto de citações. Cada citação $c_i$ contém um conjunto de atributos que inclui no mínimo uma lista de nomes de autores, um título e um local de publicação. Cada nome de autor encontrado nas citações é uma referência a um autor real. Dada um nome considerado ambíguo em relação a pelo menos uma referência de cada citação $c_i$ em $\mathcal{C}$ e um conjunto de autores $\mathcal{A} = \{a_1, a_2, ..., a_n\}$ relacionados a essas referências, o objetivo consiste em particionar o conjunto $\mathcal{C}$ em $n$ subconjuntos de forma que cada partição contenha todas, e apenas, as citações de um determinado autor $a_i$.

As abordagens tradicionais de desambiguação de nomes de autores seguem duas

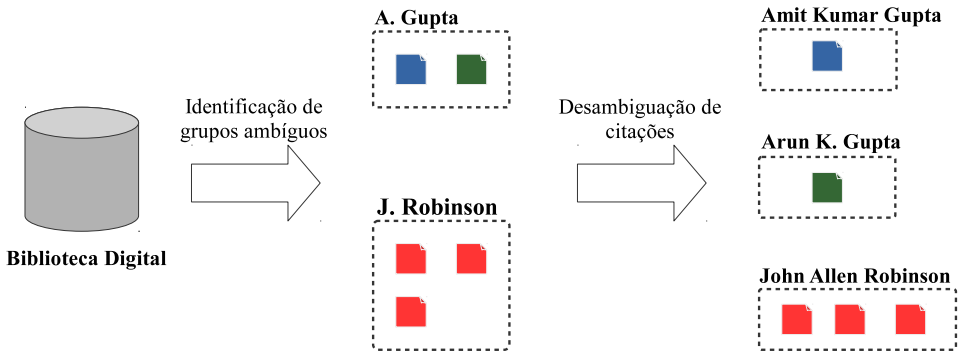



Figura 2.1: Ilustração do processo de desambiguação de referências bibliográficas.

etapas: definição de grupos ambíguos e desambiguação das citações. Na primeira etapa, as citações que compartilham referências consideradas ambíguas, segundo algum critério ou conjunto de critérios, são agrupadas. Esta etapa, conhecida como *blocagem*, também tem o objetivo de tornar o processo de desambiguação escalável e robusto ao ser aplicado em uma biblioteca digital. Exemplos de métodos de blocagem podem ser encontrados no trabalho de On et al. [2005]. Um critério comumente utilizado nesta etapa é agrupar citações que possuem referências com a mesma inicial do primeiro nome e o mesmo último nome. A segunda etapa consiste na aplicação de um método de desambiguação em cada grupo ambíguo. A Figura 2.1 ilustra as etapas de desambiguação.

### 2.1.1 Classificação dos Métodos Automáticos de Desambiguação

De acordo com Ferreira et al. [2012b], os métodos automáticos de desambiguação de nomes de autores podem ser agrupados conforme o principal tipo de abordagem explorada e também conforme o tipo de evidência utilizada. Conforme a abordagem explorada, os métodos podem ser classificados como *agrupamento de autores*, que tentam agrupar referências a um mesmo autor por meio de uma técnica de clusterização e usando alguma função de similaridade entre os atributos da citação. Nesses métodos, a função de similaridade pode ser predefinida, aprendida a partir de um método de aprendizagem de máquina ou extraída a partir do relacionamento entre autores e coautores. Ou podem ser classificados como *atribuição a autores*, quando cada referência é diretamente atribuída a um dado autor a partir da construção de um modelo que o represente. Esse modelo pode ser obtido, por exemplo, por meio de uma técnica de aprendizado supervisionado ou a partir de uma técnica de clusterização baseada em

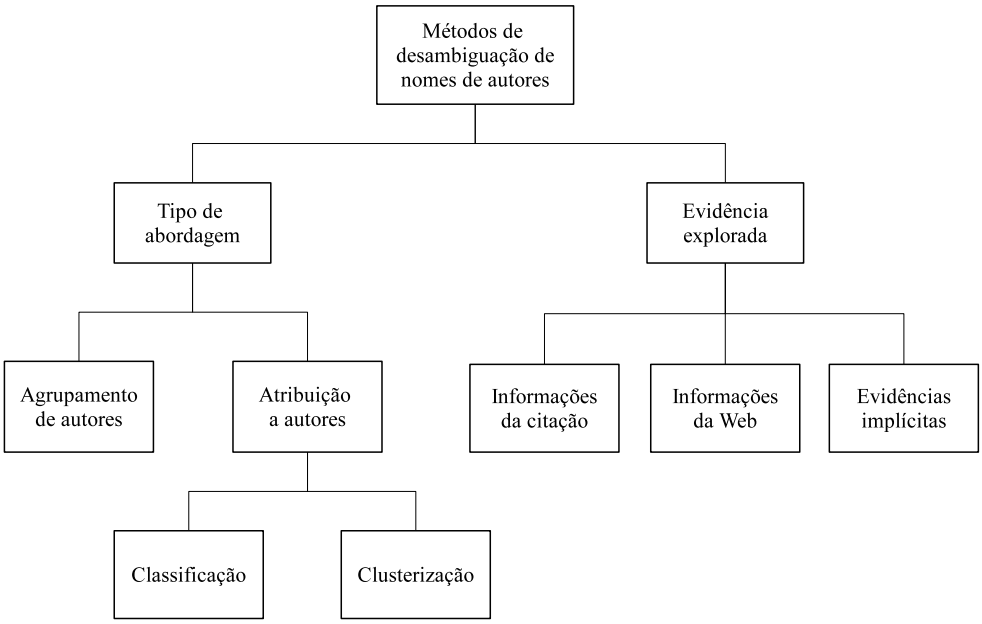



Figura 2.2: Taxonomia proposta por Ferreira et al. [2012b].

modelos. Em ambos os grupos de métodos, existem estratégias supervisionadas e não supervisionadas. Nas estratégias supervisionadas, são utilizados conjuntos de exemplos de citações cujas autorias são conhecidas para criar uma função de desambiguação ou definir uma função de similaridade entre citações. Nesses casos, os métodos geralmente são muito efetivos, entretanto, a necessidade de aquisição de exemplos de treino devido à natureza dinâmica das BDs, pode inviabilizar sua utilização.

Os métodos de desambiguação também podem ser classificados quanto ao tipo de evidência explorada. A maior parte dos trabalhos encontrados na literatura consideram apenas informações normalmente presentes na citação, como coautores, título e local de publicação. Informações adicionais podem ser obtidas a partir de buscas na Web ou por meio de evidências implícitas inferidas através dos elementos das citações como, por exemplo, a estimativa da distribuição de tópicos de uma citação utilizando *Latent Direchlet Allocation* (LDA) [Blei et al., 2003]. A Figura 2.2 resume a taxonomia proposta por Ferreira et al. [2012b]. É importante observar que as categorias definidas não são completamente disjuntas, ou seja, existem métodos que utilizam abordagens mistas ou exploram mais de um tipo de evidência.

Nas próximas seções, são apresentados os principais trabalhos relacionados baseados em agrupamento de autores e em atribuição a autores. Além disso, são apresentadas duas novas categorias de trabalhos recentes: métodos que exploram *relevance feedback* e métodos incrementais.

### 2.1.2 Abordagens Baseadas em Atribuição a Autores

Os métodos baseados em atribuição a autores tentam atribuir para cada citação um rótulo que representa um autor utilizando técnicas baseadas em aprendizado de máquina [Han et al., 2004; Veloso et al., 2012; Godoi et al., 2013], ou métricas de similaridade entre autores e citações [Lee et al., 2005]; ou tentam associá-los a um *cluster* de citações conforme algum modelo matemático utilizado para caracterizar um determinado autor [Han et al., 2005; Bhattacharya & Getoor, 2006; Tang et al., 2012].

Han et al. [2004] propuseram dois métodos supervisionados para desambiguar citações: um baseado no modelo probabilístico *Naive Bayes* e outro no classificador *Support Vector Machine* (SVM) [Mitchell, 1997]. Em ambos os métodos foram considerados os atributos coautores, título e local de publicação das citações. No modelo probabilístico, a probabilidade condicional da ocorrência de um autor dada a lista de coautores foi decomposta em seis outras probabilidades, considerando diferentes eventos, como a probabilidade de um autor escrever um artigo sozinho, ou com coautores conhecidos ou não. Já no método baseado no classificador SVM foi utilizada a representação vetorial das citações, na qual cada palavra, ou nome de autor, representa uma dimensão. Nessa representação, as características (dimensões do vetor) foram ponderadas conforme a frequência dos termos nas citações. Os métodos foram avaliados utilizando coleções obtidas a partir de listas de publicações coletadas da Web e a partir da DBLP. Na coleção obtida a partir da Web foi alcançada a taxa média de acerto de 94,5% utilizando SVM, enquanto na coleção da DBLP a maior taxa média de acerto foi de 73%, utilizando o modelo baseado no *Naive Bayes*.

Em [Han et al., 2005] foi proposto um método de clusterização baseado em um modelo probabilístico que assume que cada citação é gerada por uma mistura de $k$ componentes, que por sua vez representam os autores. Cada componente foi modelado com um modelo hierárquico baseado no *Naive Bayes* utilizando uma decomposição das probabilidades condicionais similar ao realizado em Han et al. [2004]. Os parâmetros do modelo de mistura foram estimados utilizando o algoritmo de maximização de expectativa [Dempster et al., 1977]. Esse método também foi avaliado utilizando grupos ambíguos coletados a partir da Web e da DBLP. Os resultados foram comparados com o método de clusterização *K-means*. Na maior coleção, a taxa média de acerto obtida foi de 54,1% com o método proposto, contra 34,8% com o baseline escolhido.

Lee et al. [2005] desenvolveram métodos distintos para lidar com os problemas causados pelas referências homônimas e sinônimas. Para reduzir a complexidade dos métodos, inicialmente foi utilizada uma técnica de blocagem baseada em um algoritmo de amostragem proposto por Gravano et al. [2003]. Para identificar homônimos foi

utilizada uma função de similaridade entre as representações vetoriais de um autor e uma citação, que consiste na soma ponderada das similaridades entre os atributos da citação. Para o cálculo das similares foi utilizada a distância cosseno juntamente com a ponderação dos termos *Term Frequency-Inverse Document Frequency* (TF-IDF) que são métricas padrões utilizadas em tarefas de Recuperação de Informação [Baeza-Yates & Ribeiro-Neto, 2011]. Para a identificação de sinônimos foram comparados os nomes dos autores e coautores utilizando diferentes abordagens: dois métodos supervisionados (SVM e *Naive Bayes*) e cinco métricas de similaridade usadas para comparar textos (Jaccard, TF-IDF, Jaro, Jaro−Winkler e distância cosseno). Para avaliar o método de identificação de homônimos foram utilizadas coleções extraídas a partir da DBLP e EconPapers e considerada a porcentagem de *falsas* citações ranqueadas abaixo de uma porcentagem $p$ das citações. Foi alcançada uma taxa de identificação média de 80% com $p$ igual a 20%. Para avaliar os métodos de identificação de sinônimos foram utilizadas coleções extraídas também da BioMed e e-Print. Os melhores resultados em termos de taxa de acerto, considerando os cinco primeiros candidatos a sinônimos, foram obtidos utilizando as métricas TF-IDF, Jaccard e a distância cosseno (com precisão de até 93%).

Bhattacharya & Getoor [2006] introduziram um modelo probabilístico baseado em LDA para modelar grupos colaborativos e explicar as relações de coautoria. Desta forma, os autores consideram que cada citação é gerada através da escolha dos autores a partir de uma ou mais distribuições dos grupos de colaboração. Para isso, inicialmente é selecionada uma distribuição que determina a probabilidade de cada grupo ter um autor específico escolhido para escrever o artigo. Posteriormente, utilizando essa distribuição, os autores e uma variação dos seus nomes são escolhidos para a citação. Nesse trabalho, também foi proposta uma estratégia de amostragem para estimar o número de autores dadas as referências de um conjunto de citações. Para a avaliação do método, foram utilizadas coleções obtidas a partir da CiteSeer e da arXiv. Os resultados em termos da métrica pF1 (Seção 4.3.2) foram 99% na coleção da CiteSeer e 98% na coleção da arXiv.

Tang et al. [2012] desenvolveram um *framework* probabilístico baseado no modelo de Campos Aleatórios de Markov [Kindermann & Snell, 1980], no qual são exploradas informações obtidas a partir dos atributos de um artigo (lista de autores, título, ano, local de publicação, resumo e lista de referências) e informações baseadas nos relacionamentos entre os artigos (mesmas coautorias, mesmos locais de publicação e citações entre os artigos). As evidências são modeladas como funções de características e então incorporadas em um modelo markoviano utilizado para estimar os pesos das funções e para associar as citações aos seus autores. Nesse trabalho, também foi proposta uma

estratégia para estimar o número de autores utilizando o critério de informação bayesiano [Kass & Wasserman, 1995]. A avaliação do *framework* foi realizada utilizando citações extraídas a partir do sistema ArnetMiner[1] considerando a métrica pF1. Os resultados alcançados foram 88%, quando utilizado o número correto de autores, e 80% quando este número foi estimado.

Veloso et al. [2012] criaram o método *Self-Training Lazy Associative Name Disambiguation* (SLAND) que utiliza um classificador baseado em regras de associação definidas a partir dos termos das citações. As regras são geradas no momento da classificação de uma referência. Desta forma o método é capaz de produzir funções de desambiguação específicas para cada citação a ser desambiguada. O método também utiliza uma métrica de confiança para incluir novos exemplos no treino e é capaz de identificar novos autores considerando um número mínimo de regras de associação geradas durante a classificação. O trabalho reportou resultados superiores ao SVM e ao Naive Bayes em coleções extraídas a partir da DBLP e BDBComp. Considerando a métrica *micro F1*, os valores médios alcançados foram de 91% e 45% em nas coleções DBLP e BDBComp, respectivamente.

No trabalho de Ferreira et al. [2012d], foi proposta uma estratégia para selecionar as citações mais informativas em um conjunto de teste, considerando o número de regras de associação projetadas a partir de um conjunto de treino com a utilização do método SLAND. Dessa forma, os autores mostraram que é possível manter a efetividade do método SLAND ao se utilizar como treino apenas as citações selecionadas, reduzindo em até 71% o tamanho do conjunto de treinamento. Já em [Ferreira et al., 2010, 2014], foi proposta uma estratégia para a criação automática de um conjunto de treinamento para o método SLAND baseada em duas etapas: (1) extração de *clusters* puros, e (2) remoção de *clusters* fragmentados. A primeira etapa ocorre a partir da exploração das relações de coautorias. Especificamente, duas citações são colocadas em um mesmo *cluster* se as referências forem compatíveis (utilizando um algoritmo especializado para comparação de nomes) e ambas compartilharem pelo menos um coautor que não tenha um nome popular, ou pelo menos dois coautores. A segunda etapa consiste na ordenação dos *clusters* em ordem decrescente de tamanho e então, iterativamente, são selecionados os maiores *clusters* diferentes dos *clusters* já selecionados. Nesse trabalho, os autores mostraram que utilizando apenas uma função especializada para comparação de nomes ao definir os *clusters* similares, é possível obter uma baixa taxa de fragmentação dos *clusters* do conjunto de treinamento. Uma extensão do trabalho de Ferreira et al. [2010] considerando a utilização de *relevance*

[1]http://arnetminer.org (acesso em: 27 de julho de 2015).

*feedback* foi proposta em [Ferreira et al., 2012c]. Em um trabalho similar a este último, mas utilizando um classificador diferente, foi proposto por Godoi et al. [2013]. Um resumo sobre estes dois últimos trabalhos é apresentado na Seção 2.1.4.

### 2.1.3 Abordagens Baseadas em Agrupamento de Autores

Os métodos baseados em agrupamento de autores exploram a similaridade entre as citações com o objetivo de agrupá-las utilizando alguma técnica de clusterização. A função de similaridade pode ser baseada em funções existentes considerando o tipo de cada atributo [Cota et al., 2010; Wu et al., 2014; Liu et al., 2015], aprendida através de um método de aprendizado de máquina [Huang et al., 2006; Liu et al., 2014] ou extraída a partir do relacionamento entre autores e coautores [Fan et al., 2011; Shin et al., 2014].

Em [Huang et al., 2006] foi proposto um *framework* para desambiguação de citações composto do classificador online LASVM [Bordes et al., 2005] e do método de clusterização DBSCAN [Ester et al., 1996]. Nesse *framework*, inicialmente são identificados os grupos ambíguos (blocagem) e extraídos vetores de similaridade entre cada par de citações dentro desses grupos. Os vetores de similaridade são obtidos utilizando funções específicas para cada atributo da citação, incluindo a similaridade Jaccard para endereços e afiliações e a distância de edição para e-mails e URLs. Esses vetores são então utilizados para inferir uma função de distância induzida através de uma técnica de aprendizado ativo realizada pelo LASVM. Por último, os valores de distância entre cada par de citações são utilizados pelo método de clusterização DBSCAN a fim de lidar com o problema da transitividade das similaridades, ou seja, quando em um grupo de três citações a similaridade entre dois pares é alta, mas a do terceiro par é baixa ou nula. A avaliação do método foi realizada utilizando dez grupos ambíguos extraídos a partir do CiteSeer. Em termos da métrica pF1, foi obtida uma média de 90%.

No trabalho de Cota et al. [2010], foi desenvolvido o método de clusterização hierárquico *Heuristic-based Hierarchical Clustering* (HHC), composto de duas fases: na primeira, são agrupadas referências de autores que compartilham pelo menos um nome de coautor similar. Na segunda etapa os *clusters* identificados que possuem um determinado nível de similaridade entre os atributos de título e local de publicação são unidos. O procedimento se repete até que não seja possível realizar mais fusões. Na primeira etapa, inicialmente é considerada a lista das citações cujas referências ambíguas não estão no formato curto (composta apenas pela inicial do primeiro nome e último nome) e, posteriormente, o restante das citações. Na segunda etapa, as similaridades entre os títulos e o local de publicação são calculadas utilizando a distância cosseno en-

tre as representações vetoriais destes atributos, nas quais os termos foram ponderados utilizando TF-IDF. O método proposto explora a observação segundo a qual raramente dois autores com nomes similares compartilham coautores. Na avaliação experimental, foram utilizados grupos ambíguos coletados a partir da DBLP. O resultado alcançado considerando a métrica K (Seção 4.3.1) foi de 74%.

Fan et al. [2011] desenvolveram o *framework* chamado *GrapHical FramewOrk for name disambiguaTion* (GHOST), que utiliza apenas a lista de autores das publicações para desambiguar referências com nomes idênticos. Este *framework* consiste em cinco etapas: (1) representação do conjunto de referências como um grafo $G = \{V, E\}$, no qual cada nó $v \in V$ representa um autor distinto em uma certa publicação e cada aresta (não direcionada) uma relação de coautoria; (2) seleção de caminhos mínimos entre cada par de vértices que representam nomes ambíguos, nesta etapa são desconsiderados caminhos redundantes; (3) cálculo da matriz de similaridade utilizando o conjunto dos caminhos mínimos selecionados entre cada par de vértices; (4) clusterização das referências utilizando o algoritmo *Affinity Propagation* [Frey & Dueck, 2007]; (5) melhoria do desempenho do método com a incorporação de *user feedback*. Para a avaliação do framework foram utilizados grupos ambíguos obtidos a partir da DBLP e MEDLINE. Considerando a métrica pF1, foram obtidos os valores médios de 86% e 98% nas coleções originadas da DBLP e MEDLINE respectivamente, sem a utilização de *user feedback*. Com a incorporação de *relevance feedback* em um grupo ambíguo da DBLP, os autores reportaram uma possível melhoria de até 54%.

Liu et al. [2014] desenvolveram um sistema para desambiguar as citações da MEDLINE baseado no cálculo da similaridade entre citações e clusterização aglomerativa. Nesse sistema, é utilizado o classificador de Huber [Zhang, 2004] para definir funções de ponderação para cada atributo da citação e criar uma pontuação total para representar as similaridades de cada par de citações. Estas similaridades são transformadas em valores de probabilidades utilizando o algoritmo proposto por Zadrozny & Elkan [2002]. Posteriormente, é utilizado um método proposto pelos autores para corrigir as violações de transitividade e então as citações são agrupadas através de uma técnica de clusterização aglomerativa. Os autores também utilizaram funções de similaridades específicas para os atributos encontrados na MEDLINE baseadas nas somas dos pesos *Inverse Document Frequency* (IDF) dos termos de cada atributo. Para usar o classificador de Huber, os autores criaram automaticamente um conjunto de treinamento a partir dos menores grupos ambíguos extraídos da MEDLINE, utilizando um pequeno conjunto de regras para identificar as referências compatíveis. Em termos da métrica pF1, o trabalho reportou o valor médio de 93% considerando uma coleção composta de 40 grupos ambíguos referentes a pesquisadores altamente citados na MEDLINE.

Em [Wu et al., 2014] foi proposto um algoritmo de clusterização aglomerativa hierárquico baseado na teria de Dempster–Shafer (TDS) [Shafer, 1976]. Nesse método, a TDS é utilizada para combinar características de diferentes tipos, extraídas a partir das correlações existentes entre as publicações (relações de coautorias, similaridade de conteúdo, similaridade das afiliações, similaridade do local de publicação, citações e correlação baseada nas páginas Web). Os autores também criaram um algoritmo para determinar uma condição de convergência ótima para o método de clusterização. O método foi avaliado utilizando uma coleção com 100 grupos ambíguos extraídos a partir do sistema ArnetMiner. O valor médio obtido utilizando métrica pF1 foi de 84%.

Shin et al. [2014] criaram o *framework* chamado *Graph Framework for Author Disambiguation* (GFAD) que utiliza um modelo de grafo construído a partir das relações de coautorias para desambiguar as citações bibliográficas. Neste grafo, cada autor é representado por um vértice que contém um nome (referência a este autor) e a lista de títulos das publicações deste autor. As relações de coautoria, representadas por arestas, são utilizadas para identificar círculos sociais. As referências homônimas são identificadas a partir da localização dos vértices que possuem múltiplos círculos sociais que não se sobrepõem. As informações contidas nestes vértices são divididas pelo GFAD para representar diferentes autores. Para resolver referências sinônimas, o *framework* utiliza uma métrica de similaridade para identificar nomes compatíveis e então une os vértices relacionados que estão conectados direta ou indiretamente. Os autores também propõem utilizar a similaridade cosseno sobre a lista de títulos para desambiguar citações que possuem apenas um autor. O método foi avaliado em uma coleção extraída a partir da DBLP e outra do sistema ArnetMiner. Os resultados obtidos em termos da métrica pF1 foram de 72% e 78% para as coleções obtidas a partir da DBLP e da ArnetMiner, respectivamente.

Liu et al. [2015] desenvolveram um método de clusterização múltipla baseado em três etapas: (1) clusterização das publicações que compartilham pelo menos um coautor, (2) clusterização a partir das informações contidas no título e (3) clusterização a partir das relações latentes existentes entre os locais de publicação. Na segunda etapa, foi utilizada a similaridade cosseno entre os termos dos títulos de cada par de *clusters* para determinar quais pertencem ao mesmo autor. Na terceira etapa, os autores desenvolveram um modelo para extrair relações entre os locais de publicação baseado em análise semântica latente [Deerwester et al., 1990] utilizando fatoração de matrizes não negativas. Para as duas últimas etapas são necessárias a utilização de limites de similaridades que determinam quais fragmentos serão fundidos. Durante a aplicação do método foi usada uma função para ajustar estes limites de acordo com o número de publicações da coleção. Na avaliação experimental foi utilizada uma

coleção extraída a partir da DBLP composta de nove grupos ambíguos. O resultado médio alcançado em termos de pF1 foi de 79%.

### 2.1.4 Estratégias Baseadas em *Relevance Feedback*

Nos sistemas de recuperação de informação é comum nas tarefas de busca a utilização da resposta do usuário para realizar refinamentos de uma consulta e, desta forma, melhorar a qualidade do sistema [Baeza-Yates & Ribeiro-Neto, 2011]. Para isto os resultados obtidos em uma dada consulta são avaliados como relevantes ou não pelo usuário que fez a consulta. Esta informação é dada como um *feedback* ao sistema, que então modifica a consulta original (utilizando, por exemplo, termos pertencentes aos documentos indicados). A partir da consulta modificada, novos documentos são recuperados e o processo continua até que o usuário esteja satisfeito ou desista. Esta estratégia, conhecida como *relevance feedback*, tem sido adaptada em outros problemas, inclusive para melhorar o processo de desambiguação [Wang et al., 2011; Fan et al., 2011; Ferreira et al., 2012c; Godoi et al., 2013; Li et al., 2014].

A incorporação de *relevance feedback* em um sistema de desambiguação implica a resolução de duas questões: (1) dado os resultados iniciais de um algoritmo de desambiguação, quais resultados devem ser selecionados para consultar o usuário? E (2), quando o *feedback* for fornecido, como ele pode ser explorado para melhorar o modelo de desambiguação? Após serem definidas as soluções destas questões, um processo contínuo de melhoria do desambiguador pode ser estabelecido a partir da interação do usuário da BD, como é ilustrado na Figura 2.3. A seguir são apresentadas algumas abordagens de incorporação de *relevance feedback* em técnicas distintas de desambiguação de nomes de autores.

Ferreira et al. [2012c] propuseram incorporar o *feedback* do usuário para melhorar o desempenho do método *Self-training Author Name Disambiguation* (SAND) [Ferreira et al., 2010]. Para a seleção das citações, foi utilizada uma métrica de confiança equivalente à razão entre as duas maiores estimativas de probabilidades de uma referência $r_i$ pertencer a um determinado autor identificado em um conjunto de treinamento $\mathcal{D}$. As estimativas de probabilidade são calculadas utilizando o conjunto de regras de associação geradas no momento da classificação da referência. Se o valor da métrica de confiança for menor do que um limite $\delta_{min}$, a associação é considerada duvidosa e, então, a citação é selecionada como uma candidata para classificação manual. As citações são *apresentadas ao usuário* na ordem crescente da métrica de confiança a fim de maximizar a quantidade de informações úteis que podem ser obtidas com o *feedback*. Após a associação manual, o conjunto de treinamento é atualizado e o classificador as-

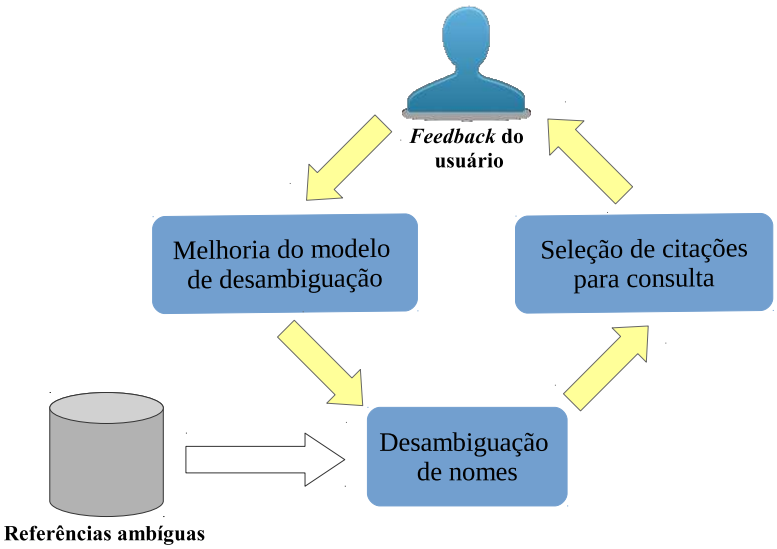



Figura 2.3: Ilustração do uso de *relevance feedback* na tarefa de desambiguação de nomes.

sociativo é executado novamente. Novas iterações deste processo são possíveis a partir da alteração do limiar $\delta_{min}$. Este método foi avaliado em coleções extraídas a partir da DBLP e os resultados indicaram ganhos médios de 10% quando por volta de 5% das citações são classificadas manualmente, considerando a métrica pF1.

Em Godoi et al. [2013] foi proposto um método de desambiguação que combina uma técnica de programação genética (PG) [Torres et al., 2009] com o classificador Floresta de Caminhos Ótimos (FCO) [Papa et al., 2012] em um esquema iterativo para explorar o *feedback* do usuário. Nesse método, um conjunto inicial de treinamento é obtido de forma similar ao SAND. Esse conjunto é utilizado pelo algoritmo de PG para definir uma função de similaridade entre as citações. Essa função é utilizada na construção de um grafo completo e ponderado no qual cada vértice representa uma citação. Esse classificador seleciona um conjunto de representantes de cada classe (protótipos) definidas no treino e então utiliza o menor caminho entre estes representantes e as citações a serem classificadas para determinar os autores de cada referência. O método proposto também tenta identificar a presença de novos autores utilizando um valor máximo, $\tau$, para a distância entre a citação de teste e o protótipo. Para a consulta com o usuário, são selecionadas um determinado número das citações cuja distância ao protótipo é mais próxima do limite $\tau$. Na avaliação experimental, foram utilizadas as mesmas coleções utilizadas em Ferreira et al. [2012c]. Na maior coleção, considerando a métrica K, os resultados variaram cerca de 70% a 85% após cinco iterações com a consulta de cinco citações por iteração.

Li et al. [2014] desenvolveram um método para explorar três tipos de user *feedback* conforme a credibilidade do usuário: usuários totalmente confiáveis, usuários confiáveis e usuários comumente credíveis. O método proposto utiliza um *perceptron* [Bishop, 2006] para combinar sete características relacionadas com pares de publicações (os atributos utilizados incluem coautores, afiliações, citações, similaridade entre os títulos e *homepages*) mais três relacionadas com os tipos de *feedback*. Os autores propuseram utilizar um formulário no qual seria possível obter uma grande quantidade de informação através de um pequeno esforço do usuário. Para isso, após a consulta de um usuário por um determinado pesquisador, seriam exibidos conjuntos de artigos que pertencem e não pertencem a este autor conforme definido pelo *perceptron*. O usuário deveria então clicar nas publicações de forma a excluir ou adicionar elementos nas listas exibidas. Em cada clique o sistema seria capaz de criar, para cada par das publicações exibidas, um registro contendo a resposta do usuário (pertencem ou não ao mesmo autor). Esses registros seriam então utilizados para refinar o treinamento do *perceptron*. O *feedback* relacionado aos usuários totalmente confiáveis restringe o resultado do *perceptron*, enquanto que os relativo a outros usuários são utilizados na forma de probabilidade para alimentar duas das entradas do classificador. Para a avaliação do método os autores simularam 620 iterações de usuários com diferentes níveis de precisão. Foram utilizados 41 grupos ambíguos extraídos a partir da DBLP, IEEE, ACM e Springer. Os experimentos demonstraram que, apesar dos ruídos incluídos durante os *feedbacks*, o método proposto obteve melhores resultados em comparação à abordagem sem *relevance feedback*.

### 2.1.5 Métodos Incrementais

Devido à constante evolução das BDs, a utilização de métodos tradicionais de desambiguação sobre todos os registros pode se tornar inviável. O ideal é que a desambiguação possa ocorrer incrementalmente, sempre que novas citações forem adicionadas ao banco de dados. Para isso, o método de desambiguação deve ser capaz de identificar não apenas a autoria das citações, mas também a presença de novos autores e as mudanças nos perfis de publicação dos autores já identificados. A seguir são apresentados dois trabalhos desenvolvidos para serem utilizados de maneira não supervisionada em um cenário incremental.

Em [Carvalho et al., 2011] foi proposto o método *Incremental Name Disambiguation* (INDi). O método compara cada referência de uma nova citação com os *clusters* representativos já identificados no repositório da BD para verificar se a referência pertence a um *cluster* conhecido. Se a referência for compatível com um *cluster* (autor)

pré-existente, ele é associado ao *cluster*. Caso contrário, a referência é considerada apontar a um autor não referenciado na BD e um novo *cluster* é criado para ele. Para decidir se uma referência e um *cluster* são compatíveis, INDi inicialmente utiliza o algoritmo de comparação de fragmentos desenvolvido por Oliveira [2005] para comparar os nomes ambíguos encontrados em cada *cluster* e então selecionar um conjunto de *clusters* candidatos. Em seguida, um *cluster* candidato é considerado compatível se a citação possuir pelo menos um coautor em comum com o *cluster* e a similaridade cosseno entre os títulos for maior do que o limite $\alpha_{Title}$, ou a similaridade cosseno entre os nomes dos locais de publicação for maior do que o limite $\alpha_{Venue}$. Quando a citação possui apenas um autor, ou todas as citações em um cluster possuirem apenas um autor (ou seja, quando o atributo coautores for vazio), o algoritmo verifica apenas as similaridades entre os títulos e entre os locais de publicação, mas aumenta os valores dos limites $\alpha_{Title}$ e $\alpha_{Venue}$ conforme o valor de um parâmetro $\delta$. Na avaliação experimental, foi utilizada uma coleção extraída a partir da BDBComp e duas coleções sintéticas geradas utilizando a ferramenta *Synthetic Generator of Authorship Records* (SyGAR) [Ferreira et al., 2012a] para simular a introdução de novos autores em uma BD. Em termos da métrica K, o método proposto obteve os valores de 87% na BDBComp e 76% na coleção sintética simulando a introdução de novos autores a uma taxa de 10%.

Esperidião et al. [2014] propuseram estender o método INDi investigando algumas estratégias para reduzir a fragmentação dos *clusters* de citações que representam os autores. O algoritmo desenvolvido utiliza as novas referências inseridas no repositório como evidências para identificar *clusters* fragmentados. Dada uma nova citação $c_k$, o algoritmo seleciona um conjunto de *clusters* $S$ que provavelmente contém referências do mesmo autor de $c_k$ utilizando a função proposta por Carvalho et al. [2011] na qual são verificadas a presença de coautores em comum e as similaridades cosseno dos títulos e locais de publicação. Se $S$ for vazio, o método considera que a referência de $c_k$ pertence a um novo autor; caso contrário, ele une os *clusters* encontrados em $S$ e associa a citação ao novo *cluster*. Com o objetivo de filtrar possíveis ruídos e preservar a pureza dos *clusters* obtidos, os autores avaliaram várias estratégias para selecionar referências representativas de cada *cluster* antes de calcular as similaridades e realizar as uniões dos *clusters* candidatos. Na avaliação experimental, foram utilizadas coleções sintéticas simulando a introdução de novos autores e mudanças nos perfis de publicação, e coleções reais obtidas a partir da DBLP e BDBComp. Comparando com o INDi, o trabalho reportou ganhos de até 16,6% em uma das coleções reais e até 7% em uma das coleções sintéticas.

## 2.2 Extração de Características Baseadas na Coocorrência de Palavras

Na área de recuperação de informação, especialmente na tarefa de classificação de textos, é comum o processamento de textos a fim de extrair características a serem utilizadas por métodos de aprendizado de máquina [Baeza-Yates & Ribeiro-Neto, 2011]. Um desafio encontrado ao realizar esta extração é como obter eficientemente um grande conjunto de características discriminativas. Neste contexto, alguns pesquisadores investigaram a utilização da coocorrência de termos [Zaïane & Antonie, 2002; Tan et al., 2002; Figueiredo et al., 2011]. Para ilustrar a utilização desta estratégia na tarefa de classificação de texto, na qual o objetivo é determinar a(s) categoria(s) de um dado documento, Figueiredo et al. [2011] destacam algumas características extraídas durante seus experimentos em uma coleção de documentos da área de medicina. Nesta coleção, o termo *pain* está relacionado com diversas categorias como *nervous system diseases*, *musculoskeletal diseases* e *pathological conditions, signs and symptoms*. Entretanto, os autores observaram que a utilização de características que representam pares de termos como {*pain*, *facial*} e {*pain*, *postoperative*} tinham alto poder discriminativo, o que auxiliou a classificação dos documentos de teste.

As técnicas de extração de caracteríistiscas baseadas na coocorrência de palavras também podem ser aplicadas para resolver o problema de desambiguação de nomes ao considerar o conjunto de citações de cada autor uma determinada categoria e cada citação um documento. Neste trabalho foi investigado a utilização do procedimento proposto por Figueiredo et al. [2011] conforme apresentado a seguir.

A estratégia de extração de características desenvolvida por Figueiredo et al. [2011] consiste em quatro passos executados antes do treinamento de um classificador e um durante o estágio de teste, conforme ilustrado na Figura 2.4. Para o treino, o primeiro passo consiste na seleção das palavras, ou termos, que serão utilizados para gerar as características baseadas na coocorrência dos termos, ou *c-features*. Os autores propõe realizar esta seleção com base no ganho de informação [Mitchell, 1997] de cada termo. No contexto da tarefa de classificação de texto (que é similar a tarefa de desambiguação de nomes), o ganho de informação de um termo $t_i$ é calculado através da Equação 2.1:

$$Gain(\mathcal{D}, t_i) = Entropy(\mathcal{D}) - \frac{|\mathcal{D}_i|}{|\mathcal{D}|} Entropy(\mathcal{D}_i) - \frac{|\mathcal{D} - \mathcal{D}_i|}{|\mathcal{D}|} Entropy(\mathcal{D} - \mathcal{D}_i) \quad (2.1)$$

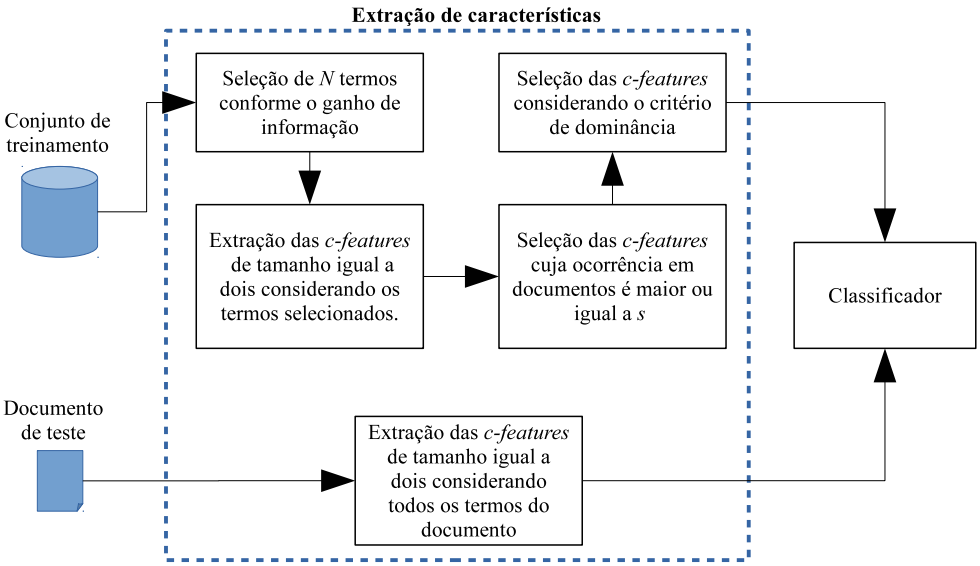



Figura 2.4: Ilustração da estratégia de extração de *c-features* proposta por Figueiredo et al. [2011].

onde, $\mathcal{D}$ representa o conjunto de documentos de treinamento e $\mathcal{D}_i$ o subconjunto dos documentos em $\mathcal{D}$ quem possuem o termo $t_i$. A função *Entropy* é definida como:

$$Entropy(\mathcal{S}) = \sum_{i \in \mathcal{C}} -p_i \log_2 p_i \tag{2.2}$$

onde, $\mathcal{C}$ representa o conjunto de classes (ou categorias) e $p_i$ a proporção de $\mathcal{S}$ que pertence a classe $i$.

Após a seleção dos $N$ termos com os maiores ganhos de informação, todas as combinações de tamanho igual a dois são obtidas para compor as *c-features* (ou seja, as características são geradas sem nenhuma restrição quanto à ordem ou distância entre as palavras dentro de um documento). No terceiro passo, ocorre a seleção das *c-features* cuja frequência na coleção seja maior do que um determinado valor $s$. Posteriormente, é determinado, para cada classe, o poder discriminativo das características com base no critério de dominância. Este critério é equivalente à probabilidade condicional da classe dado a ocorrência da *c-feature*. Finalmente, no quarto passo, ocorre a inclusão das *c-features* nos documentos se o critério de dominância for maior do que um limite $\tau$. Na etapa de teste são incluídas todas as *c-features* possíveis de serem formadas.

Figueiredo et al. [2011] avaliaram a estratégia proposta em quatro coleções relacionadas com a tarefa de classificação de texto, utilizando três técnicas de aprendizado de máquina. Os resultados demonstraram ganhos em quase todos os cenários, variando

de 4,8% a 10,3% com a utilização do KNN e de 1,3% a 10% com o SVM considerando a métrica *macro F1*.

# Capítulo 3

# Método Proposto

Neste capítulo, é apresentado o método de desambiguação proposto, os procedimentos que podem ser utilizados para estimar os seus parâmetros e uma análise da complexidade de tempo do seu algoritmo. Neste trabalho, as citações são representadas por conjuntos de termos que incluem os nomes da lista de autores, as palavras do título e palavras do nome do local de publicação[1]. Os termos das citações foram pré-processados utilizando as seguintes tarefas: remoção de caracteres não alfanuméricos, conversão de caracteres maiúsculos em minúsculos, remoção de *stop-words*[2] e extração de radicais utilizando o algoritmo de Porter [1997]. As duas últimas tarefas foram aplicadas apenas nos termos do título e do local de publicação. A seguir é apresentado o modelo de desambiguação desenvolvido, que pode ser classificado como um algoritmo baseado em associação de autores. Para as equações definidas, são utilizadas as notações apresentadas na Tabela 3.1.

## 3.1 Desambiguação Baseada no Autor mais Similar

O perfil de publicação de um autor pode ser caracterizado pela distribuição dos termos encontrados em suas referências bibliográficas. A lista de coautores captura a sua rede de colaboração enquanto os conjuntos de termos do título e do local de publicação capturam os seus interesses de pesquisa. Portanto, cada termo presente na citação fornece alguma evidência da associação entre uma referência da citação e um autor. A força dessa evidência varia conforme o atributo ao qual o termo pertence e conforme a sua capacidade discriminativa. Por exemplo, normalmente a presença de coautores

[1]Neste trabalho foram utilizados apenas os atributos mais comuns encontrados nas citações com o objetivo de demonstrar sua praticidade em cenários reais. Entretanto, outros atributos, como endereços e palavras-chave, podem facilmente serem incorporados no modelo proposto.

[2]http://en.wikipedia.org/wiki/Stop_words (acesso em: 27 de julho de 2015).

Tabela 3.1: Tabela de notações

| **Notação** | **Descrição** |
|---|---|
| $c_i = \ c_i^a \cup c_i^c \cup c_i^t \cup c_i^v$ | Citação representada por um conjunto de termos obtidos a partir do nome do autor ($c_i^a$), coautores ($c_i^c$), título ($c_i^t$) e local de publicação ($c_i^v$). |
| $c_i^x = \{t_l, t_m, ...\}$ | Atributo de uma citação representada por um conjunto de termos. |
| $a_i = \{c_l, c_m, ...\}$ | Autor $i$ representado por um conjunto de citações. |
| $a_i^x = \{c_l^x, c_m^x, ...\}$ | Conjunto de termos obtidos a partir de um atributo $x$ das citações em $a_i$. |
| $\mathcal{A} = \{a_1, a_2, ..., a_n\}$ | Conjunto de autores definidos em um conjunto de treinamento. |
| $\mathcal{D} = \bigcup_{i=1}^{n} a_i$ | Conjunto de citações encontradas em um conjunto de treinamento. |
| $n_i = \lvert\{a_j : a_j \in A \wedge \exists c_l \in a_j, t_i \in c_l\}\rvert$ | Número de autores que utilizaram o termo $t_i$ em alguma citação. |
| $f_i = \lvert\{c_j : c_j \in D \wedge t_i \in c_j\}\rvert$ | Número de citações que possuem o termo $t_i$. |
| $f_{i,j} = \lvert\{c_k : c_k \in a_j \wedge t_i \in c_k\}\rvert$ | Número de citações em um grupo $a_j$ que possuem o termo $t_i$. |
| $w_a, w_c, w_t, w_v$ | Pesos associados ao nome do autor, lista de coautores, título e local de publicação respectivamente. |

em comum em duas citações é uma evidência mais forte de que ambas as citações pertencem ao mesmo autor do que a presença de um termo em comum nos títulos. Essa diferença de importância dos atributos tem sido explorada por diversos métodos de desambiguação, seja utilizando uma técnica de clusterização [Cota et al., 2010; Liu et al., 2015] ou de classificação [Han et al., 2004; Lee et al., 2005].

Com base nessas observações, um método simples de desambiguação consiste na definição de uma função de similaridade entre citações e autores (representados pelos grupos ou *clusters* de citações). A desambiguação de uma citação $c_k$ é realizada a partir da identificação do autor mais similar. Esta abordagem foi utilizada por Liu et al. [2015] para identificar homônimos, conforme descrito na Seção 2.1.2. Diferente desse trabalho, esta dissertação propõe utilizar uma nova função de similaridade entre atributos baseada em heurísticas, além de estratégias que permitem a utilização do método de forma incremental e não supervisionada. O método desenvolvido consiste em três fases: (i) seleção de *clusters* candidatos, (ii) cálculo das similaridades entre cada citação e *clusters* candidatos, e (iii) atualização do conjunto de treinamento. A última fase é composta pelas etapas: (a) identificação de novos autores, (b) atualização de associações duvidosas, e (c) identificação de *clusters* fragmentados. Cada fase explora heurísticas específicas do domínio com o objetivo de criar automaticamente

um conjunto de treinamento utilizado para definir as associações entre referências e autores. As principais etapas de cada fase são mostradas no Algoritmo 1 e detalhadas nas próximas seções.

**Algoritmo 1** Desambiguação de Nomes de Autores

**Entrada:** Conjunto de *clusters* $\mathcal{A}$, citação de teste $c_k$, conjunto de citações duvidosas $\mathcal{E}$.
**Saída:** Associa um cluster (autor) $a_l$ à citação $c_k$

$\mathcal{G} \leftarrow \emptyset$
**for all** $a_j \in A$ **do**
  **if** $comparacaoDeFragamentos(c_k^a, a_j^a)$ **then** ▷ Seleção de *clusters* candidatos
    Calcula $sim(c_k, a_j)$ de acordo com a Equação 3.1
    **if** $sim(c_k, a_j) > 0$ **then**
      $\mathcal{G} \leftarrow \mathcal{G} \cup \{a_j\}$
    **end if**
  **end if**
**end for**

Seja $a_l$ o cluster com o maior valor de similaridade
**if** $sim(c_k, a_l) \leq \gamma$ **then** ▷ Verificação de novo autor
  $a_z \leftarrow \{c_k\}$
  $\mathcal{A} \leftarrow \mathcal{A} \cup \{a_z\}$
  $\mathcal{E} \leftarrow \mathcal{E} \cup \{c_k\}$
**else**
  $a_l \leftarrow a_l \cup \{c_k\}$
  Calcula $\Delta(c_k)$ de acordo com a Equação 3.3
  **if** $\Delta(c_k) \leq \gamma$ **then** ▷ Verifica se a associação é confiável
    $\mathcal{E} \leftarrow \mathcal{E} \cup \{c_k\}$
  **else**
    $\mathcal{E}_l \leftarrow \{c_j | c_j \in \mathcal{E} \wedge c_j \cap c_k \neq \emptyset\}$
    **for all** $c_j \in \mathcal{E}_l$ **do** ▷ Verifica associações duvidosas
      Remove $c_j$ do conjunto de treinamento $\mathcal{A}$
      Reclassifica $c_j$
      **if** $\Delta(c_j) > \gamma$ **then**
        $\mathcal{E} \leftarrow \mathcal{E} - \{c_j\}$
        Procura por fragmentos de *clusters*
      **end if**
    **end for**
    **for all** $a_j \in \mathcal{G}$ **do** ▷ Procura fragmentos do cluster $a_l$
      Calcula a similaridade entre $a_l$ e $a_j$ conforme a Equação 3.4
      **if** $sim(a_l, a_j) > \gamma$ **then**
        $a_l \leftarrow a_l \cup a_j$
        $\mathcal{A} \leftarrow \mathcal{A} - \{a_j\}$
      **end if**
    **end for**
  **end if**
**end if**

### 3.1.1 Seleção de *Clusters* Candidatos

Dada uma citação de teste $c_k$ e uma referência ambígua $c_k^a$, a primeira fase do método proposto consiste na seleção dos *clusters* em $\mathcal{A}$ que possuem pelo menos um nome compatível com $c_k^a$ utilizando o algoritmo de comparação de nomes desenvolvido por Oliveira [2005], chamado de Comparação de Fragmentos (linha 3 do Algoritmo 1). Este algoritmo compara cada palavra de dois nomes normalizados. Ele considera que as partes dos nomes de um mesmo autor devem ter em comum pelo menos a inicial do primeiro nome e o último nome. Nomes do meio não precisam aparecer na mesma ordem e se uma palavra possui apenas uma letra e esta for equivalente à inicial de uma palavra em outro nome, a primeira é considerada uma abreviação. No final, se todas as palavras forem compatíveis ou se houver apenas uma palavra não compatível, os nomes são considerados possíveis referências a uma mesma pessoa.

Essa fase evita a necessidade de comparação entre a citação de teste e todos os *clusters* do conjunto de treinamento e também contribui para manter a pureza dos grupos de citações.

### 3.1.2 Cálculo da Similaridade entre Autor e Citação

A similaridade entre uma citação $c_k$ e um autor $a_j$ (*cluster* de citações) é calculada a partir da soma ponderada das similaridades entre cada atributo da citação $c_k$ e das citações do *cluster* $a_j$, conforme mostrado na Equação 3.1.

$$sim(c_k, a_j) = w_a d(c_k^a, a_j) + w_c d(c_k^c, a_j) + w_t d(c_k^t, a_j) + w_v d(c_k^v, a_j) \tag{3.1}$$

A função de similaridade entre atributos é definida como:

$$d(c_k^x, a_j) = \sum_{t_i \in c_k^x \cap a_j^x} w(t_i, a_j) \tag{3.2}$$

onde,

$$w(t_i, a_j) = \begin{cases} \left(1 + \dfrac{1 - n_i}{|\mathcal{A}|}\right) \sqrt{\left(\dfrac{f_{i,j}^2 + 1}{|a_j| f_i + 2}\right)} & \text{se } n_i > 0 \\ 0 & \text{caso contrário} \end{cases}$$

A função $w(t_i, a_j)$ pondera os termos de cada citação, retornando um valor entre 0 e 1, definido de acordo com algumas heurísticas:

- Quanto maior o número de autores que utilizam o termo $t_i$, menor é seu poder discriminativo. A soma $(1 + (1 - n_i)/|\mathcal{A}|)$ retorna 1 se o termo é utilizado em

apenas um grupo, ou $1/|\mathcal{A}|$ se o termo encontrado em todos os *clusters*.

- Dada a ocorrência de um termo $t_i$, o valor da probabilidade condicional $P(a_j|t_i)$ fornece uma boa evidência da associação entre a citação que possui o termo $t_i$ e o autor representado pelo *cluster* $a_j$. Essa probabilidade pode ser estimada através da fração $f_{i,j}/f_i$, entretanto a estimativa pode ser enviesada devido a altas diferenças das classes comumente encontradas no problema de desambiguação de nomes de autores. Estas diferenças refletem padrões de publicação encontrados na comunidade científica: poucos autores com um grande número de publicações e um grande número de autores com apenas algumas publicações. A fim de reduzir o efeito causado por este tipo de viés, o valor desta estimativa é multiplicado pela distribuição do termo no grupo, estimado através da razão $f_{i,j}/|a_j|$.

- Em uma tarefa de desambiguação incremental, o conjunto de treinamento pode ser inicialmente vazio e então crescer na medida em que novas citações são classificadas. Para suavizar o valor das estimativas descritas no item anterior, foram adicionadas as constantes 1 e 2 no cálculo do quociente e a raiz quadrada do valor resultante.

Nos experimentos realizados neste trabalho, o critério de blocagem utilizado foi agrupar as referências que compartilham a mesma inicial do primeiro nome e o mesmo último nome. Dessa forma, os nomes ambíguos que estão no formato curto fornecem pouca ou nenhuma informação a respeito da classe da citação, portanto, nesses casos, esses termos não foram incluídos nos cálculos das similaridades.

### 3.1.3 Atualização do Conjunto de Treinamento

Um método de desambiguação de nomes de autores a ser aplicado em uma biblioteca digital deve ser capaz de identificar novos autores e lidar com mudanças nos perfis de publicação dos autores já identificados. Essas características refletem fenômenos que ocorrem nos repositórios das BDs e são, portanto, essenciais para a construção de um método de desambiguação incremental. A última fase do método proposto utiliza os valores de similaridade entre as citações e os autores para lidar com ambos os problemas. Nas seções a seguir são detalhados os procedimentos utilizados para atualizar automaticamente o conjunto de treinamento.

#### 3.1.3.1 Identificação de Novos Autores

O baixo valor de similaridade entre uma citação $c_k$ e o *cluster* mais similar do conjunto de treinamento, $a_l$, pode indicar a presença de um novo autor (não representado por nenhum grupo em $\mathcal{A}$) ou simplesmente uma mudança da área de pesquisa de um autor já identificado. Portanto, foi definido um valor mínimo de similaridade, $\gamma$, para possibilitar a criação de novos *clusters* durante o processo de desambiguação. Esta etapa corresponde a verificação realizada na linha 11 do Algoritmo 1. O limite $\gamma$ também contribui para manter a pureza dos *clusters*, uma vez que quanto maior o valor de $sim(c_k, a_l)$, maior a probabilidade da correta associação entre o *cluster* a citação.

#### 3.1.3.2 Atualização de Associações Duvidosas

No momento em que novas citações são incluídas no conjunto de treinamento, é possível utilizar as associações mais confiáveis para reclassificar as associações duvidosas, e, possivelmente, corrigir erros de classificação. Para determinar se uma associação é confiável ou não, foi utilizada uma métrica de confiança baseada nos valores de similaridade obtidos entre $c_k$ e cada *cluster* em $\mathcal{A}$, conforme mostrado na Equação 3.3:

$$\Delta(c_k) = \frac{sim(c_k,a_l)^2 - sim(c_k,a_m)^2}{\sum_{a_i \in \mathcal{A}} sim(c_k, a_i)} \tag{3.3}$$

onde $a_l$ e $a_m$ representam respectivamente o primeiro e segundo *cluster* mais similar a $c_k$.

Essa métrica fornece um valor proporcional à similaridade entre $c_k$ e $a_l$ e à distância entre $c_k$ e o restante dos grupos definidos no conjunto de treinamento, especialmente em relação ao segundo grupo mais similar. Estes valores refletem a confiança na precisão da classificação, portanto dado um limite $\Delta_{min}$, as citações com $\Delta(c_k) \leq \Delta_{min}$ são incluídas em um conjunto $\mathcal{E}$ para futuras reclassificações. Estas reclassificações ocorrem sempre que uma citação é incluída no treino e a associação de sua referência a um autor é considerada confiável. Nestes casos, todas as citações incluídas no conjunto $\mathcal{E}$ que compartilham pelo menos um termo em comum com a citação confiável (linhas 21 a 29 do Algoritmo 1) são reavaliadas.

O valor de $\Delta(c_k)$ é limitado pelo valor de $sim(c_k,a_l)$. Quando $sim(c_k,a_l) < \gamma$, um novo *cluster* é criado devido à falta de evidência, portanto, é natural que o limite $\Delta_{min}$ seja maior ou igual a $\gamma$. Como a Equação 3.3 tende a retornar valores menores com o aumento do número de *clusters*, foi observado que o valor de $\Delta_{min}$ pode ser configurado com o valor de $\gamma$ a fim de se obter um bom equilíbrio entre o número de

reclassificações devido ao tamanho de $\mathcal{E}$ e o número de possíveis correções que podem ser obtidas durante as reclassificações.

#### 3.1.3.3 Identificação de *Clusters* Fragmentados

Durante a desambiguação, os grupos de citações dos autores que trabalham em diferentes linhas de pesquisa podem ser fragmentados, mesmo se forem utilizados pequenos valores para o parâmetro $\gamma$. Para lidar com esse problema, sempre que uma nova citação é incluída no conjunto de treinamento e sua classificação é considerada confiável, são calculadas as similaridades entre o grupo da citação e todos os outros grupos que compartilham pelo menos um termo em comum com a citação (linhas 27 e 30 a 36 do Algoritmo 1). Se essa similaridade for maior do que $\gamma$, os *clusters* são unidos. Neste procedimento, a similaridade dos pares de *clusters* resultantes de um processo manual de desambiguação é definida como 0 a fim de evitar associações indevidas[3]. Para comparar dois *clusters* foi utilizada uma função de similaridade baseada na Equação 3.1, mas com uma função diferente para comparar os atributos das citações:

$$sim(a_l, a_m) = w_a d(a_l^a, a_m^a) + w_c d(a_l^c, a_m^c) + w_t d(a_l^t, a_m^t) + w_v d(a_l^v, a_m^v) \tag{3.4}$$

onde $|a_l| < |a_m|$ e,

$$d(a_l^x, a_m^x) = \frac{1}{|a_l|} \sum_{t_i \in a_l^x \cap a_m^x} w(t_i, a_m) \tag{3.5}$$

Quanto maior forem os *clusters* comparados, maior deverá ser o número de termos compartilhados entre eles, mesmo que não representem grupos de citações de um mesmo autor. Portanto, a normalização utilizada no somatório dos pesos dos termos de cada atributo tem a função de reduzir os valores obtidos conforme o tamanho do menor *cluster*. Além disso, quando $a_l$ possui apenas uma citação, $c_k$, $sim(a_l, a_m) = sim(c_k, a_m)$. Isto é importante para manter a coerência com a estratégia utilizada para identificar novos autores.

É importante observar que o valor de $\gamma$ define: (1) o valor mínimo de similaridade entre uma citação e um grupo de citações para realizar uma associação, (2) o valor mínimo de similaridade entre dois *clusters* para que eles sejam considerados de um mesmo autor, e (3) o valor mínimo de $\Delta$ para que uma classificação seja considerada confiável. Desta forma, o valor deste parâmetro pode ser utilizado para controlar a taxa

[3]Neste casos consideramos que não houve erros no processo manual de rotulação.

de fragmentação e pureza dos *clusters* gerados durante a desambiguação das citações.

## 3.2 Definição dos Valores dos Parâmetros

O método apresentado possui cinco parâmetros: $w_a$, $w_c$, $w_t$, $w_v$ e $\gamma$. Valores adequados para estes parâmetros podem ser obtidos utilizando procedimentos padrões de validação cruzada a partir de um conjunto de treinamento [Mitchell, 1997]. Entretanto, a fim de aumentar a eficiência desta busca, foi proposta uma estratégia que consiste em duas etapas: (1) definição dos valores dos pesos e (2) definição do valor de $\gamma$.

### 3.2.1 Pesos dos Atributos

Para que uma citação $c_k$ seja corretamente associada a um autor $a_l$, os pesos devem ser definidos de forma que $sim(c_k, a_l) > sim(c_k, a_m), \forall a_m \in A - \{a_l\}$. A partir dos dados de treinamento, é possível definir um conjunto de inequações baseadas na diferença de similaridades entre cada atributo na forma: $w_a diff_a + w_c diff_c + w_t diff_t + w_v diff_v > 0$, onde $diff_x$ representa a diferença $d(c_k^x, a_l) - d(c_k^x, a_m)$.

Através de um procedimento de validação cruzada, é possível obter aproximadamente $\bar{n}|\mathcal{D}|$ inequações que compõem um sistema provavelmente insolúvel. Entretanto, os valores das diferenças ($diff_a$, $diff_c$, $diff_t$ e $diff_v$) refletem o grau de importância de cada atributo. Por exemplo, se vários autores publicarem em uma mesma conferência, alguns valores de $diff_v$ serão pequenos (ou até negativos), indicando que o peso $w_v$ não deve ser maior do que $w_c$ ou $w_a$. A partir destas observações, a estratégia adotada para definir os valores dos pesos dos atributos utiliza as diferenças negativas de similaridades entre cada atributo $x$ obtidas durante as classificações, como mostrado abaixo:

$$diff(c_k^x, a_l, a_m) = \begin{cases} d(c_k^x, a_l) - d(c_k^x, a_m) & \text{se } d(c_k^x, a_l) < d(c_k^x, a_m) \\ 0 & \text{caso o contrário} \end{cases}$$

A partir de um procedimento de validação cruzada de 10 divisões (*folds*), para cada citação $c_k$ do conjunto de treinamento $\mathcal{D}$, são calculados os valores de $diff(c_k^x, a_l, a_m)$ para cada atributo $x$ e *cluster* $a_m \in \mathcal{A} - \{a_l\}$. As soma destes valores são normalizadas utilizando a equação mostrada abaixo a fim de definir o valor de cada peso $w_x$. Esta normalização garante o valor mínimo igual a 1 ao atributo menos discriminativo:

$$w_x = \frac{\max_i \log(-\sum_{c_k \in \mathcal{D}} \sum_{a_m \in \mathcal{A} - \{a_l\}} diff(c_k^i, a_l, a_m))}{\log(-\sum_{c_k \in \mathcal{D}} \sum_{a_m \in \mathcal{A} - \{a_l\}} diff(c_k^x, a_l, a_m))}$$

### 3.2.2 Evidência Mínima

O valor do parâmetro $\gamma$ deve ser definido de forma a maximizar o *trade-off* entre a taxa de acerto na associação de uma citação a um autor em $\mathcal{A}$ e a taxa de acerto na identificação de novos autores. Quanto maior o valor de $\gamma$, maior é a chance de identificar novos autores, mas maior também é a chance de aumentar a fragmentação dos *clusters* em $\mathcal{A}$. Este *trade-off* pode ser representado pela soma de probabilidades mostrada abaixo:

$$\hat{p}(sim(c_k, a_m) \leq \gamma | a_l \notin \mathcal{A})\hat{p}(a_l \notin \mathcal{A}) \quad + \\ \hat{p}(sim(c_k, a_l) > \gamma | a_l \in \mathcal{A})\hat{p}(a_l \in \mathcal{A})$$

onde $a_m$ representa o grupo de maior similaridade com $c_k$ e $a_l$ o autor correto de $c_k$.

A probabilidade da citação pertencer a um autor ausente no conjunto de treinamento, $\hat{p}(a_l \notin \mathcal{A})$, foi manualmente configurada com o valor igual a 0,5[4]. As probabilidades condicionais foram estimadas baseadas nas seguintes etapas:

1. Para cada citação $c_k \in \mathcal{D}$ e *cluster* $a_j \in \mathcal{A}$, são calculadas as similaridades $sim(c_k, a_j)$, desconsiderando a presença de $c_k$ no conjunto de treinamento. Isso é similar ao procedimento de validação cruzada *leave-one-out* [Mitchell, 1997].

2. Para cada citação $c_k \in \mathcal{D}$, é armazenado, em um conjunto $\mathcal{F}$, o valor de similaridade entre a citação e o grupo que representa seu autor, $sim(c_k, a_l)$.

3. Para cada citação $c_k \in \mathcal{D}$, é armazenado, em um conjunto $\mathcal{G}$, o maior valor de similaridade desconsiderando a presença de $a_l$ em $\mathcal{A}$, $\max_{a_i \in \mathcal{A} - \{a_l\}} sim(c_k, a_i)$.

4. O conjunto $\mathcal{F}$ é ordenado de forma ascendente para que as posições de cada valor correspondam ao número de citações com autores no conjunto de treinamento que seriam incorretamente classificadas se o parâmetro $\gamma$ fosse maior do que $sim(c_k, a_l)$. Essas posições são utilizadas para estimar as probabilidades $\hat{p}(sim(c_k, a_l) > \gamma | a_l \in \mathcal{A})$.

5. O conjunto $\mathcal{G}$ é ordenado de forma descendente para que as posições de cada valor correspondam ao número de citações com autores que não iriam ser identificados se o valor de $\gamma$ fosse menor do que $sim(c_k, a_m)$. Essas posições são utilizadas para estimar as probabilidades $\hat{p}(sim(c_k, a_m) \leq \gamma | a_l \notin \mathcal{A})$.

[4]Esse valor foi escolhido a fim de equilibrar a importância entre a identificação de novos autores e redução da fragmentação. Outros valores podem ser utilizados conforme a necessidade do administrador de uma BD.

Com as probabilidades condicionais estimadas, o valor de $\gamma$ corresponde ao que maximiza o *trade-off* apresentado no início desta seção. Se a quantidade de treinamento for pequena, o valor de $\gamma$ também pode ser muito pequeno, portanto, foi definido o limite mínimo igual ao peso do atributo de menor valor. Este limite garante que existam pelo menos dois[5] termos em comum entre um *cluster* e uma citação do atributo menos discriminativo.

## 3.3 Treinamento Não Supervisionado

Quando não há um conjunto de treinamento disponível, os valores dos parâmetros podem ser configurados manualmente (para isso são fornecidas algumas orientações na Seção 4.6.4) ou estimados utilizando uma estratégia para aquisição automática de conjuntos de treinamento, como realizado no método SAND [Ferreira et al., 2014].

A estratégia proposta por Ferreira et al. [2014] utiliza as relações de coautorias para criar um conjunto de *clusters* puros. Posteriormente é selecionado um subconjunto destes *clusters* para compor um conjunto de treinamento. Utilizando o algoritmo apresentado nas seções anteriores, um conjunto de *clusters* puros pode ser obtido configurando os parâmetros com os valores: $w_a = 1$, $w_c = 1$, $w_t = 0$, $w_v = 0$ e $\gamma = 1$. Isso garante que uma citação seja associada a um *cluster* somente se possuir no mínimo dois coautores em comum, ou o mesmo nome do autor e um coautor em comum. Diferente da abordagem utilizada pelo método SAND, neste trabalho é proposto utilizar todos os *clusters* obtidos nesta etapa para compor um conjunto de treinamento e então calcular os valores dos pesos a partir do procedimento descrito na Seção 3.2.1. Após a definição desses valores, um valor apropriado para o parâmetro $\gamma$ pode ser encontrado entre os valores do menor e do maior peso. Enquanto o valor de menor peso representa um valor mínimo de similaridade esperado, conforme descrito na seção anterior, o valor do maior peso equivale à presença de dois termos em comum, entre a citação e o autor, do atributo mais discriminativo (por exemplo, a presença de dois coautores em comum).

Uma vez estabelecidos os limites inferiores e superiores para o parâmetro $\gamma$, um valor adequado pode ser encontrado utilizando a seguinte heurística: quanto menor a dúvida total das classificações realizadas, melhor é a qualidade dos *clusters* obtidos. Assim, a soma dos valores de confiança fornecidos pela Equação 3.3 para cada citação em $\mathcal{D}$ pode ser utilizada como critério para escolher o valor de $\gamma$ e, consequentemente, o melhor resultado obtido utilizando o Algoritmo 1. O procedimento completo para definição dos parâmetros a partir de um conjunto de teste $\mathcal{T}$ é mostrado no Algoritmo 2.

[5]Observe que o valor retornado pela função $w(t_i, a_l)$ nunca é igual a 1 devido à presença das constantes 1 e 2, conforme descrito na Seção 3.1.2.

**Algoritmo 2** Treinamento Não Supervisionado

**Entrada:** Conjunto de citações de teste $\mathcal{T}$, número de iterações $n$.
**Saída:** Valores dos atributos $w_a$, $w_c$, $w_t$, $w_v$ e $\gamma$.

1: $w_a = 1$, $w_c = 1$, $w_t = 0$, $w_v = 0$ e $\gamma = 1$
2: $\mathcal{A} = \varnothing$, $\mathcal{E} = \varnothing$
3: **for all** $c_k \in \mathcal{T}$ **do**
4: Classifica $c_k$ conforme Algoritmo 1.
5: **end for**

6: Calcula os valores de $w_a$, $w_c$, $w_t$ e $w_v$ utilizando o procedimento descrito na Seção 3.2.1.
7: Seja $w_{min}$ o peso de menor valor e $w_{max}$ o peso de maior valor.
8: $step = (w_{max} - w_{min})/(n-1)$
9: $\gamma = w_{min}$, $maxConfidence = 0$, $bestGamma = \gamma$
10: **while** $\gamma \leq w_{max}$ **do**
11: $\mathcal{A} = \varnothing$, $\mathcal{E} = \varnothing$
12: **for all** $c_k \in \mathcal{T}$ **do**
13: Classifica $c_k$ conforme Algoritmo 1.
14: **end for**
15: $confidence = 0$
16: **for all** $c_k \in \mathcal{T}$ **do**
17: $confidence = confidence + \Delta(c_k)$;
18: **end for**
19: **if** $confidence > maxConfidence$ **then**
20: $maxConfidence = confidence$
21: $bestGamma = \gamma$
22: **end if**
23: $\gamma = \gamma + step$
24: **end while**
25: $\gamma = bestGamma$

## 3.4 Inclusão de Características Baseadas na Coocorrência de Termos

O método de desambiguação apresentado nas seções anteriores utiliza os termos encontrados nas citações desconsiderando qualquer relação existente entre dois ou mais termos e os autores. Uma forma de incorporar informações derivadas dessas relações consiste na inclusão de características (representadas por novos termos) baseadas na coocorrência de palavras. Para isso foi utilizado a estratégia proposta por Figueiredo et al. [2011], descrita na Seção 2.2.

O procedimento de extração de *c-features* foi realizado considerando cada atributo da citação isoladamente. A estratégia requer a definição de três parâmetros: porcentagem, $p$, de termos selecionados para compor as *c-features* considerando como critério de ordenação o ganho de informação; frequência mínima em citações, $s$, que a

*c-feature* deve ter para ser selecionada e valor mínimo de dominância, $\tau$. Considerando os resultados reportados por Figueiredo et al. [2011] e as características do problema de desambiguação de nomes (menos termos relacionados com cada classe, em comparação ao problema de classificação de texto), os valores dos parâmetros $p$ e $s$ foram fixados em 100% e 2, respectivamente. Para o parâmetro $\tau$ foram testados os valores 0%, 25%, 50%, 75% e 100%, para cada coleção. Os resultados obtidos com a utilização desta estratégia são apresentados nas Seções 4.4.2.1 e 4.5.2.1.

## 3.5 Análise de Complexidade de Tempo

O algoritmo de desambiguação proposto pode ser implementado de maneira eficiente utilizando *tabelas hash*[6]. A análise apresentada a seguir considera que as operações de inserção e busca nesta estrutura de dados possuem complexidade de tempo $O(1)$.

Na primeira fase do método, para cada *cluster*, o algoritmo de Comparação de Fragmentos é executado para cada referência encontrada no grupo. Este algoritmo possui complexidade de tempo proporcional ao número de letras e nomes em cada referência. Como esses números são normalmente pequenos, o número de instruções executadas pode ser representada por uma constante. Portanto, esta fase possui a complexidade de tempo igual a $O(|\mathcal{D}|)$. Na segunda fase, o número de instruções executadas para calcular a similaridade entre a citação de teste e um *cluster* (Equação 3.1) é proporcional ao número de termos da citação, logo a complexidade de tempo nesta etapa é igual a $O(|\mathcal{A}||c_k|)$.

Após a classificação, se o valor de $sim(c_k, a_l)$ for maior do que $\gamma$, a citação é incluída no conjunto de treinamento em $O(|c_k|)$, caso o contrário um novo *cluster* é criado e a citação é associada a ele também em $O(|c_k|)$[7]. Se o valor de $\Delta(c_k) > \gamma$, o conjunto de citações duvidosas, $\mathcal{E}_l$, é obtido em $O(|\mathcal{E}||c_k|)$ (para cada termo de $c_k$, é necessário verificar as citações em $\mathcal{E}$). Posteriormente, para reclassificar as citações em $\mathcal{E}_l$, o número de instruções executadas é proporcional ao número médio de clusters candidatos vezes o número médio de termos por citação, logo esta etapa possui complexidade de tempo igual a $O(|\mathcal{E}_l||\mathcal{A}||\mathcal{V}|)$, onde $\mathcal{V}$ representa o conjunto dos termos encontrados nas citações do conjunto de treinamento.

O cálculo das similaridades entre cada par de *clusters* realizado nas linhas 27 e 31 do Algoritmo 1 possui complexidade de tempo proporcional ao número de termos do

[6]http://en.wikipedia.org/wiki/Hash_table (acesso em: 27 de julho de 2015).

[7]A inclusão de uma citação no conjunto de treinamento implica a atualização dos valores de frequência utilizados no cálculo das similaridades. Esses valores foram armazenados em tabelas *hash* indexadas pelo termo da citação.

menor *cluster*. Portanto, a complexidade de tempo para realizar a busca por fragmentos de *cluster* é igual a $O(|\mathcal{A}||\mathcal{V}|)$. O número de instruções executadas para realizar a união de dois *clusters* também é proporcional ao número de termos do *cluster*.

O pior caso do algoritmo ocorre quando $sim(c_k, a_l) > \gamma$ e $\Delta(c_k) > \gamma$. Neste caso, a soma do custos associados com cada tarefa é igual a $O(|\mathcal{D}|+|\mathcal{A}||c_k|+|\mathcal{E}||c_k|+|\mathcal{E}_l||\mathcal{A}||\mathcal{V}|)$. Como $|\mathcal{E}_l| \leq |\mathcal{E}|$ e $|c_k| \ll |\mathcal{V}|$, a complexidade de tempo total do algoritmo proposto pode ser representada por $O(|\mathcal{D}| + |\mathcal{E}||\mathcal{A}||\mathcal{V}|)$.

# Capítulo 4

# Avaliação Experimental

Neste capítulo são apresentados os resultados dos experimentos realizados para avaliar a qualidade do método proposto, aqui chamado de Desambiguação Incremental baseada no *Cluster* mais Similar (DICS), em diferentes cenários de aplicação. Inicialmente, são descritos os *baselines*, as coleções e métricas utilizadas para a avaliação experimental. Posteriormente, a Seção 4.4 compara o método desenvolvido com os principais métodos supervisionados encontrados na literatura. Na Seção 4.5, o método DICS é avaliado em cenários não supervisionados, comparado com dois métodos baseados em agrupamento de autores e um método não supervisionado baseado em associação de autores. Por último, a Seção 4.6 apresenta os resultados obtidos com a utilização do método DICS de forma incremental em coleções reais e sintéticas. Nesta seção também são avaliados os componentes do método, o tempo de execução e sua sensibilidade aos valores dos parâmetros.

## 4.1 *Baselines*

Para avaliar o desempenho do método desenvolvido foram utilizados cinco métodos baseados em associação de autores - SAND [Ferreira et al., 2014], SLAND [Veloso et al., 2012], Cosine [Lee et al., 2005], SVM [Han et al., 2004] e NB [Han et al., 2004] -, dois métodos baseados em agrupamentos de autores - LAVSM-DBSCAN [Huang et al., 2006] e HHC [Cota et al., 2010] - e dois métodos incrementais - INDi [Carvalho et al., 2011] e MINDi [Esperidião et al., 2014].

Para os métodos SAND, SLAND, HHC, INDi e MINDi foram utilizadas implementações fornecidas pelos próprios autores. Detalhes sobre cada um destes métodos podem ser encontrados na Seção 2.1. Os outros *baselines* foram implementados conforme descrito nos respectivos trabalhos. Todas as implementações, inclusive a do

método proposto, foram realizadas utilizando a linguagem Java.

No método SVM, cada autor representa uma classe para a qual um classificador é treinado. Cada citação é representada por um vetor de características formado a partir dos termos de cada atributo, ponderados utilizando TF-IDF. A associação entre citação e autor é realizada utilizando a abordagem *uma classe contra todas as outras* para classificações multiclasse. Para a implementação deste *baseline* foi utilizado o pacote libSVM disponível em http://www.csie.ntu.edu.tw/~cjlin/libsvm/ (acesso em: 27 de julho de 2015).

O método NB assume que cada citação é gerada por um modelo baseado no *Naïve Bayes*. Portanto, para estimar os parâmetros do modelo, é utilizado um conjunto de treinamento. Para cada autor $a_i$, é calculada a probabilidade do autor gerar uma citação utilizando regras derivadas do teorema de Bayes. Uma citação de teste é atribuída à classe que possui a maior probabilidade de tê-la produzido. Os parâmetros do modelo para o atributo coautores incluem estimativas de probabilidades de um determinado autor escrever um artigo com coautores conhecidos ou não. Todos os parâmetros foram calculados conforme definido por Han et al. [2004].

Cosine é uma versão do método proposto por Lee et al. [2005] para identificação de homônimos que utiliza a distância cosseno juntamente com a ponderação TF-IDF para calcular as similaridades entre citações e grupos. De maneira similar ao DICS, este método também utiliza pesos diferentes para cada atributo da citação, mas não considera a inclusão do nome da referência ambígua no cálculo das similaridades.

LASVM-DBSCAN combina o classificador online LASVM e o método de clusterização DBSCAN. Para a implementação deste baseline foi utilizado o pacote LASVM disponível em http://leon.bottou.org/projects/lasvm e a versão do DBSCAN disponível no pacote Weka: http://www.cs.waikato.ac.nz/ml/weka/ (acesso em: 27 de julho de 2015).

O método *Merge-oriented Incremental name Disambiguation* (MINDi) utiliza as novas referências para selecionar *clusters* compatíveis e realizar fusões de *clusters* fragmentados. Para isto, os autores avaliaram várias estratégias para o cálculo da similaridade entre o *cluster* e a citação e identificação dos *clusters* fragmentados. Neste trabalho foi utilizada a estratégia de redução de fragmentação que obteve melhores resultados conforme reportado por Esperidião et al. [2014]: utilizar uma porcentagem $p$ das citações mais próximas ao centróide do *cluster* para definir a similaridade entre o *cluster* e a referência e então unir apenas os dois *clusters* mais similares a esta nova referência.

## 4.2 Coleções

Os experimentos foram realizados utilizando duas coleções extraídas a partir da DBLP, uma a partir da BDBComp e quatro coleções sintéticas. A seguir são detalhadas cada uma destas coleções.

### 4.2.1 DBLP

A primeira coleção derivada da DBLP possui um total de 8.418 citações associadas a 477 autores distintos, o que significa uma média de aproximadamente 17,6 citações por autor. Esta coleção inclui 5.585 citações cuja referência ambígua está no formato curto. Variações desta coleção foram utilizadas em trabalhos anteriores [Han et al., 2004, 2005; Veloso et al., 2012; Shin et al., 2014]. A versão utilizada neste trabalho foi baseada na coleção criada por Han et al. [2005], a qual foi manualmente desambiguada pelos autores. Para isto, eles utilizaram páginas pessoais dos autores, nome das afiliações, e-mail e nomes dos coautores em formato completo. Também foram enviados e-mails para alguns autores a fim de confirmar suas autorias. Nesta dissertação, foram utilizados 14 grupos ambíguos com algumas alterações, que incluem remoção de duplicatas, correção de associações e alteração de nomes no formato curto para o formato como eles realmente são encontrados na DBLP.

### 4.2.2 BDBComp

A coleção extraída a partir da BDBComp possui um total de 361 citações associadas com 205 autores, o que equivale a aproximadamente dois registros por autor. Nesta coleção apenas seis nomes estão no formato curto. Embora esta seja a menor coleção utilizada neste trabalho, ela é difícil de desambiguar devido à pequena quantidade de citações por autor. Esta coleção contém 10 grupos ambíguos, e foi utilizada por Veloso et al. [2012] e Ferreira et al. [2014].

### 4.2.3 KISTI

A segunda coleção derivada da DBLP, aqui chamada de KISTI, foi criada pelo Instituto de Ciência e Tecnologia da Informação da Coréia [Kang et al., 2011] para desambiguação de nomes ingleses. Os mil mais frequentes nomes de autores foram obtidos a partir de uma versão da DBLP do final de 2007, juntamente com suas citações. Para desambiguar esta coleção, os autores submeteram consultas compostas pelo sobrenome do autor e o título da publicação de cada citação no Google com o objetivo de en-

contrar páginas pessoais. As primeiras 20 páginas retornadas por cada consulta foram manualmente verificadas para identificar as páginas pessoais corretas relacionadas com cada referência. As páginas identificadas foram então utilizadas para desambiguar as citações. Esta coleção possui 41.659 citações relacionadas com 867 grupos ambíguos e 6.908 autores.

### 4.2.4 Coleções Sintéticas

As coleções sintéticas foram geradas a fim de simular cenários em que referências para novos autores são continuamente inseridas em um repositório e novas citações reflitam mudanças no perfil de publicação dos autores. Para isto foi utilizado o gerador de referências bibliográficas desenvolvido por Ferreira et al. [2012a], chamado de *Synthetic Generator of Authorship Records* (SyGAR).

A principal entrada do SyGAR é uma coleção de citações previamente desambiguadas. Cada registro desta coleção compõe os três atributos mais comumente explorados pelos métodos de desambiguação: lista de autores e lista de termos únicos encontrados no título e no nome do local de publicação. O gerador utiliza vários parâmetros de configuração para definir um cenário a ser simulado, entre eles: o número de cargas, o número de registros por carga, a probabilidade de selecionar um novo coautor, a probabilidade de selecionar um novo local de publicação e a porcentagem de novos autores a serem inseridos em cada carga. Como saída, o SyGAR produz uma lista representativa de citações, sinteticamente geradas, na qual cada registro consiste no conjunto dos três atributos mencionados anteriormente.

Para gerar as coleções sintéticas, foi utilizada como entrada uma versão da coleção DBLP, descrita na seção anterior, com 4.272 registros associados a 220 autores. Foram geradas coleções sintéticas compostas de dez cargas cada, representando dados inseridos a cada ano em um repositório. O número de registros gerados para cada autor foi baseado na distribuição apresentada na Tabela 4.1, extraída a partir da coleção de entrada. O estado inicial, antes das dez cargas, foi gerado com o mesmo número de citações da coleção de entrada.

Foram geradas quatro coleções sintéticas a fim de avaliar dois cenários: (1) adição de citações com referências a novos autores e, (2) mudanças nos perfis de publicação (através de alterações na distribuição dos tópicos utilizados por cada autor). Para o primeiro cenário, foram geradas duas coleções, *SyGAR-N5* e *SyGAR-N10*, nas quais em cada carga foram adicionados um conjunto de referências relacionadas a novos autores, correspondente a, respectivamente, 5% e 10% do número total de autores presentes antes de cada nova carga. Para o segundo cenário, também foram geradas

Tabela 4.1: Distribuição do número médio de publicações por ano utilizado no SyGAR.

| | **Número médio de publicações por ano** | | | |
|---|---|---|---|---|
| | **Um** | **Dois** | **Três** | **Quatro** |
| Novos autores | 55% | 30% | 10% | 5% |
| Autores existentes | 14% | 42% | 28% | 16% |

duas coleções, SyGAR-C10 e SyGAR-C50, nas quais, para cada carga, respectivamente 10% e 50% dos autores alteraram o seu perfil de publicação.

## 4.3 Métricas de Avaliação

Para comparar os resultados obtidos pelos métodos de desambiguação, foram utilizadas duas métricas: a métrica K e a *pairwise* F1. Estas são métricas-padrão adotadas por pesquisadores em diversos trabalhos anteriores [Ferreira et al., 2012b]. A ideia principal, como é mostrado a seguir, consiste na comparação dos *clusters* extraídos pelo método de desambiguação com os *clusters* ideais, obtidos manualmente. Nas descrições, um *cluster* extraído por um método de desambiguação é chamado de *cluster empírico*, enquanto que um *cluster* ideal é chamado de *cluster teórico*.

### 4.3.1 Métrica K

A métrica K determina o *trade-off* entre a pureza média dos *clusters* (PMC) e a pureza média dos autores (PMA). Dado um grupo ambíguo, a PMC avalia a pureza dos *clusters* empíricos em relação aos *clusters* teóricos. Portanto, se um *cluster* empírico contiver apenas citações associadas a um mesmo autor, o valor correspondente de PMC será igual a 1. PMC é definida como:

$$\text{PMC} = \frac{1}{N} \sum_{i=1}^{e} \sum_{j=1}^{t} \frac{n_{ij}^2}{n_i} \tag{4.1}$$

onde $N$ é o número total de citações, $t$ é o número de *clusters* teóricos, $e$ é o número de *clusters* empíricos, $n_i$ é o número total de citações no *cluster* empírico $i$, e $n_{ij}$ é o número total de citações no *cluster* empírico $i$ os quais também estão no *cluster* teórico $j$.

Para um dado grupo ambíguo, PMA avalia a fragmentação do *cluster* empírico em relação ao *cluster* teórico. Se um *cluster* empírico não estiver fragmentado, o valor

correspondente de PMA será igual a 1. PMA é definida como:

$$\text{PMA} = \frac{1}{N} \sum_{j=1}^{t} \sum_{i=1}^{e} \frac{n_{ij}^2}{n_j} \tag{4.2}$$

onde $n_j$ é o número total de citações no *cluster* teórico $j$.

O valor de $K$ consiste na média geométrica entre os valores de PMC e PMA. Esta métrica avalia a pureza e coesão dos *clusters* empíricos extraídos por cada método em cada grupo ambíguo, sendo definida como:

$$K = \sqrt{\text{PMC} \times \text{PMA}} \tag{4.3}$$

### 4.3.2 Métrica *pairwise* F1

A métrica *pairwise* F1 ($p$F1) consiste na métrica F1 calculada utilizando precisão e revocação baseada em pares. A precisão baseada em pares ($pP$) é definida como:

$$pP = \frac{a}{a + c} \tag{4.4}$$

onde $a$ corresponde ao número de pares de citações em um *cluster* empírico que são corretamente associadas ao mesmo autor, e $c$ é o número de pares de citações de um *cluster* empírico que não correspondem ao mesmo autor.

A revocação baseada em pares ($pR$) é definida como:

$$pR = \frac{a}{a + b} \tag{4.5}$$

onde $b$ corresponde ao número de pares de citações associadas ao mesmo autor que não estão no mesmo *cluster* empírico.

A métrica pF1 é, portanto, definida como:

$$p\text{F1} = 2 \cdot \frac{pP \times pR}{pP + pR} \tag{4.6}$$

## 4.4 Comparação dos Métodos Supervisionados

Nesta seção são apresentados os experimentos utilizados para avaliar o método proposto (DICS) quando existe um conjunto de treinamento contendo exemplos de citações agrupadas por autor.

### 4.4.1 Configuração Experimental

Para avaliação dos métodos supervisionados foram utilizadas as coleções DBLP, BDBComp e KISTI. Os grupos ambíguos de cada coleção foram divididos em conjuntos de treino e teste, cada um com 50% do número total de citações. Esta divisão foi realizada de maneira aleatória e repetida 10 vezes. Todos os resultados obtidos foram comparados utilizando o teste $t$ pareado de duas caudas com a correção de Holm-Bonferroni [Holm, 1979] considerando o nível de confiança de 95%. Para cada grupo ambíguo foram executados os baselines: SLAND, Cosine, SVM e NB.

Os valores dos parâmetros do método proposto foram estimados utilizando os procedimentos descritos na Seção 3.2. Os pesos dos atributos utilizados pelo método Cosine também foram obtidos utilizando um procedimento similar ao apresentado na Seção 3.2.1. Os parâmetros do método NB foram estimados a partir dos dados de treino conforme descrito no trabalho original. O método SVM foi configurado para utilizar o *kernel Radial Basis Function* (RBF) e seus parâmetros foram obtidos através da ferramenta GRID disponível no pacote libSVM. Por último, para o método SLAND, foi utilizado um procedimento baseado em validações cruzadas no treino para encontrar os melhores valores dos seus parâmetros em cada grupo ambíguo. É importante observar que o conjunto de treinamento para o método DICS é utilizado não apenas para a configuração dos parâmetros do modelo, mas também para a criação dos *clusters* iniciais do conjunto $\mathcal{A}$.

### 4.4.2 Resultados

Os resultados obtidos em cada coleção são mostrados nas Tabelas 4.2 e 4.3, considerando os valores médios das métricas K e pF1 respectivamente. Para ambas as métricas, o método DICS obteve os melhores valores em todas as coleções com apenas um empate estatístico na coleção DBLP com o método Cosine em relação a métrica pF1. Nesta coleção, os ganhos variaram de aproximadamente 3,9% a 29,2% em relação à metrica K e de 2% a 48,6% em relação à métrica pF1, comparando com os métodos Cosine e NB respectivamente.

Nas coleções BDBComp e KISTI, os métodos SLAND e DICS foram mais efetivos que os outros *baselines* devido às capacidades de identificação de novos autores e atualização automática do conjunto de treinamento. Nestas coleções vários grupos ambíguos possuem autores com apenas uma ou duas citações. Consequentemente, a maioria destes autores não são representados pelos conjuntos de treinamento. Em termos da métrica K, os ganhos do método DICS em relação ao SLAND foram de aproximadamente 8,4% na BDBComp e 2,7% na KISTI. Comparando com os outros

Tabela 4.2: Valores médios da métrica K obtidos pelos métodos supervisionados.

| **Método** | **Coleção** | | |
|---|---|---|---|
| | **DBLP** | **BDBComp** | **KISTI** |
| DICS | **0,919** ± 0,026 | **0,956** ± 0,024 | **0,952** ± 0,002 |
| SLAND | 0,878 ± 0,027 | 0,882 ± 0,031 | 0,927 ± 0,002 |
| Cosine | 0,884 ± 0,028 | 0,746 ± 0,041 | 0,883 ± 0,003 |
| SVM | 0,777 ± 0,038 | 0,579 ± 0,042 | 0,797 ± 0,004 |
| NB | 0,711 ± 0,045 | 0,537 ± 0,067 | 0,768 ± 0,005 |

Tabela 4.3: Valores médios da métrica pF1 obtidos pelos métodos supervisionados.

| **Método** | **Coleção** | | |
|---|---|---|---|
| | **DBLP** | **BDBComp** | **KISTI** |
| DICS | **0,916** ± 0,032 | **0,728** ± 0,143 | **0,835** ± 0,009 |
| SLAND | 0,869 ± 0,034 | 0,512 ± 0,205 | 0,806 ± 0,008 |
| Cosine | **0,898** ± 0,029 | 0,400 ± 0,199 | 0,757 ± 0,008 |
| SVM | 0,702 ± 0,070 | 0,201 ± 0,124 | 0,623 ± 0,009 |
| NB | 0,616 ± 0,080 | 0,246 ± 0,157 | 0,596 ± 0,009 |

métodos, os ganhos foram superiores a 28% na BDBComp e a 7,8% na KISTI. Em termos da métrica pF1, os ganhos chegaram a 262% e 196% comparando com os método SVM e NB, respectivamente, na BDBComp. Na KISTI, comparando com estes mesmos *baselines*, os ganhos ultrapassaram 34%. É importante, entretanto, verificar que a maior diferença da métrica pF1 ocorre devido à sua forma de cálculo: quando um *cluster* é composto por apenas uma citação, não é contabilizado nenhum par de citações no cálculo da precisão e revocação baseada em pares, o que gera um valor igual a 0 e diminui a média geral.

As Tabelas 4.4 e 4.5 mostram os resultados obtidos em cada grupo ambíguo pelos três melhores métodos nas coleções DBLP e BDBComp respectivamente. Já na Tabela 4.6, são mostrados os resultados obtidos nos 40 maiores grupos da coleção KISTI (o conjunto formado por estes grupos foi utilizado nos trabalhos [Ferreira et al., 2012d,c; Godoi et al., 2013]).

Observando a Tabela 4.4, é possível verificar que o método proposto obteve os melhores resultados em termos da métrica K em todos os grupos ambíguos, exceto no grupo *S. Lee* (considerando os valores exatos, DICS foi cerca de 0,4% pior comparado com o método SLAND). Os maiores ganhos com a métrica K foram de 17,1% e 13,1% no grupo *J. Martin*, comparando com o SLAND e Cosine respectivamente. Ainda considerando a métrica K, foram verificados cinco empates estatísticos com o método SLAND e seis com o método Cosine. Em relação à métrica pF1, as diferenças das médias são menores quando comparado com o método Cosine (apenas 3 ganhos

Tabela 4.4: Resultados obtidos por SLAND, Cosine e DICS em cada grupo ambíguo na coleção DBLP.

| **Grupo** | **SLAND** | | **Cosine** | | **DICS** | |
|---|---|---|---|---|---|---|
| **ambíguo** | **K** | **pF1** | **K** | **pF1** | **K** | **pF1** |
| A. Gupta | 0,89 ± 0,02 | 0,90 ± 0,03 | 0,91 ± 0,01 | 0,94 ± 0,01 | **0,95** ± 0,01 | **0,96** ± 0,01 |
| A. Kumar | **0,93** ± 0,02 | **0,92** ± 0,03 | **0,92** ± 0,02 | **0,93** ± 0,02 | **0,94** ± 0,01 | **0,93** ± 0,02 |
| C. Chen | 0,84 ± 0,01 | **0,85** ± 0,02 | 0,82 ± 0,01 | **0,83** ± 0,02 | **0,87** ± 0,01 | **0,82** ± 0,01 |
| D. Johnson | **0,86** ± 0,04 | **0,85** ± 0,05 | **0,85** ± 0,02 | **0,85** ± 0,02 | **0,86** ± 0,02 | **0,84** ± 0,03 |
| J. Lee | 0,84 ± 0,01 | 0,83 ± 0,01 | 0,85 ± 0,01 | **0,87** ± 0,01 | **0,88** ± 0,01 | **0,87** ± 0,01 |
| J. Martin | 0,82 ± 0,04 | 0,74 ± 0,06 | 0,85 ± 0,04 | 0,82 ± 0,05 | **0,96** ± 0,01 | **0,96** ± 0,02 |
| J. Robinson | **0,93** ± 0,03 | 0,92 ± 0,03 | **0,93** ± 0,02 | **0,94** ± 0,03 | **0,96** ± 0,02 | **0,97** ± 0,02 |
| J. Smith | 0,90 ± 0,01 | 0,91 ± 0,01 | **0,89** ± 0,01 | **0,93** ± 0,01 | **0,91** ± 0,01 | **0,93** ± 0,01 |
| K. Tanaka | 0,94 ± 0,02 | 0,94 ± 0,02 | **0,96** ± 0,01 | **0,97** ± 0,01 | **0,98** ± 0,01 | **0,98** ± 0,01 |
| M. Brown | **0,92** ± 0,04 | **0,91** ± 0,05 | **0,89** ± 0,06 | **0,90** ± 0,06 | **0,94** ± 0,03 | **0,94** ± 0,03 |
| M. Jones | 0,78 ± 0,03 | 0,76 ± 0,04 | 0,84 ± 0,03 | 0,86 ± 0,03 | **0,89** ± 0,02 | **0,90** ± 0,02 |
| M. Miller | 0,94 ± 0,01 | 0,95 ± 0,01 | 0,98 ± 0,01 | **0,99** ± 0,01 | **0,98** ± 0,01 | **0,99** ± 0,01 |
| S. Lee | **0,82** ± 0,01 | **0,82** ± 0,01 | 0,80 ± 0,01 | **0,83** ± 0,01 | **0,82** ± 0,01 | 0,80 ± 0,01 |
| Y. Chen | 0,88 ± 0,01 | 0,87 ± 0,01 | 0,89 ± 0,01 | **0,93** ± 0,01 | **0,92** ± 0,01 | **0,92** ± 0,01 |
| **Média** | 0,88 ± 0,03 | 0,87 ± 0,03 | 0,88 ± 0,03 | **0,90** ± 0,03 | **0,92** ± 0,03 | **0,92** ± 0,03 |

Tabela 4.5: Resultados obtidos por SLAND, Cosine e DICS em cada grupo ambíguo na coleção BDBComp.

| **Grupo** | **SLAND** | | **Cosine** | | **DICS** | |
|---|---|---|---|---|---|---|
| **ambíguo** | **K** | **pF1** | **K** | **pF1** | **K** | **pF1** |
| A. Oliveira | **0,90** ± 0,04 | **0,88** ± 0,08 | 0,80 ± 0,08 | **0,77** ± 0,11 | **0,94** ± 0,03 | **0,91** ± 0,06 |
| A. Silva | 0,87 ± 0,02 | 0,74 ± 0,06 | 0,78 ± 0,03 | 0,60 ± 0,07 | **0,98** ± 0,02 | **0,94** ± 0,06 |
| F. Silva | 0,93 ± 0,03 | **0,42** ± 0,28 | 0,71 ± 0,05 | 0,08 ± 0,07 | **0,97** ± 0,02 | **0,56** ± 0,30 |
| J. Oliveira | 0,82 ± 0,05 | **0,64** ± 0,10 | 0,68 ± 0,04 | 0,49 ± 0,08 | **0,88** ± 0,03 | **0,72** ± 0,08 |
| J. Silva | 0,84 ± 0,06 | 0,62 ± 0,14 | 0,80 ± 0,03 | 0,59 ± 0,08 | **0,96** ± 0,03 | **0,85** ± 0,12 |
| J. Souza | **0,89** ± 0,05 | **0,81** ± 0,09 | 0,80 ± 0,06 | 0,69 ± 0,09 | **0,92** ± 0,05 | **0,87** ± 0,10 |
| L. Silva | 0,84 ± 0,04 | 0,56 ± 0,15 | 0,77 ± 0,05 | 0,50 ± 0,12 | **0,98** ± 0,02 | **0,93** ± 0,06 |
| M. Silva | 0,86 ± 0,03 | 0,00 ± 0,00 | 0,77 ± 0,07 | 0,15 ± 0,12 | **0,97** ± 0,02 | **0,55** ± 0,29 |
| R. Santos | **0,96** ± 0,03 | **0,27** ± 0,31 | 0,63 ± 0,04 | **0,05** ± 0,06 | **0,99** ± 0,02 | **0,40** ± 0,37 |
| R. Silva | 0,90 ± 0,03 | 0,19 ± 0,16 | 0,71 ± 0,05 | 0,09 ± 0,10 | **0,96** ± 0,04 | **0,54** ± 0,36 |
| **Média** | 0,88 ± 0,03 | 0,51 ± 0,21 | 0,75 ± 0,04 | 0,40 ± 0,20 | **0,96** ± 0,02 | **0,73** ± 0,14 |

estatísticos). Ainda assim, os ganhos chegaram a 28,6% comparando com o SLAND e a 16,7% comparando com o Cosine (também no grupo *J. Martin*).

Na BDBComp, DICS obteve os melhores valores em todos os grupos ambíguos. Comparando com o método Cosine, houve apenas dois empates estatísticos considerando a métrica pF1 (grupos *A. Oliveira* e *R. Santos*). Em relação à métrica K os ganhos variaram de 2,7% (grupo *R. Santos*) a 16,7% (grupo *L. Silva*) e de 15,7% (grupo *J. Souza*) a 55,8% (grupo *R. Santos*) comparando com os métodos SLAND e Cosine, respectivamente. Em relação a métrica pF1 a variação dos valores é maior devido a

Tabela 4.6: Resultados obtidos por SLAND, Cosine e DICS nos 40 maiores grupo ambíguos da KISTI.

| **Grupo** | **SLAND** | | **Cosine** | | **DICS** | |
|---|---|---|---|---|---|---|
| **ambíguo** | **K** | **pF1** | **K** | **pF1** | **K** | **pF1** |
| A. Choudhary | **1,00** ± 0,00 | **1,00** ± 0,00 | **1,00** ± 0,00 | **1,00** ± 0,00 | **1,00** ± 0,00 | **1,00** ± 0,00 |
| A. Gupta | 0,88 ± 0,03 | 0,88 ± 0,04 | 0,90 ± 0,03 | 0,90 ± 0,03 | **0,94** ± 0,02 | **0,93** ± 0,02 |
| D. Eppstein | **1,00** ± 0,00 | **1,00** ± 0,00 | **1,00** ± 0,00 | **1,00** ± 0,00 | **1,00** ± 0,00 | **1,00** ± 0,00 |
| D. Lee | 0,93 ± 0,03 | 0,93 ± 0,03 | 0,89 ± 0,02 | 0,88 ± 0,04 | **0,97** ± 0,01 | **0,96** ± 0,01 |
| H. Chen | 0,83 ± 0,04 | 0,78 ± 0,05 | 0,86 ± 0,02 | 0,86 ± 0,04 | **0,94** ± 0,01 | **0,93** ± 0,02 |
| H. Wang | 0,90 ± 0,03 | 0,87 ± 0,05 | 0,91 ± 0,02 | 0,90 ± 0,03 | **0,97** ± 0,02 | **0,97** ± 0,02 |
| J. Chen | 0,84 ± 0,04 | 0,80 ± 0,05 | 0,89 ± 0,03 | 0,95 ± 0,02 | **0,97** ± 0,01 | **0,99** ± 0,01 |
| J. Halpern | **0,92** ± 0,01 | **0,92** ± 0,01 | 0,77 ± 0,03 | 0,78 ± 0,04 | 0,86 ± 0,05 | **0,87** ± 0,05 |
| J. Kim | 0,79 ± 0,02 | 0,56 ± 0,05 | 0,81 ± 0,04 | 0,64 ± 0,07 | **0,93** ± 0,01 | **0,84** ± 0,04 |
| J. Lee | 0,81 ± 0,02 | 0,67 ± 0,04 | 0,83 ± 0,03 | 0,75 ± 0,05 | **0,93** ± 0,02 | **0,86** ± 0,03 |
| J. Li | 0,85 ± 0,03 | 0,75 ± 0,06 | 0,87 ± 0,02 | 0,88 ± 0,03 | **0,97** ± 0,01 | **0,96** ± 0,02 |
| J. Liu | 0,84 ± 0,03 | 0,81 ± 0,05 | 0,89 ± 0,03 | 0,92 ± 0,03 | **0,97** ± 0,01 | **0,97** ± 0,01 |
| J. Mitchell | **0,99** ± 0,01 | **0,99** ± 0,01 | 0,96 ± 0,02 | 0,97 ± 0,02 | **0,99** ± 0,00 | **0,99** ± 0,00 |
| J. Smith | 0,83 ± 0,02 | 0,83 ± 0,03 | 0,87 ± 0,03 | 0,92 ± 0,02 | **0,94** ± 0,01 | **0,96** ± 0,01 |
| J. Wang | 0,85 ± 0,02 | 0,80 ± 0,04 | 0,88 ± 0,02 | 0,88 ± 0,02 | **0,95** ± 0,01 | **0,93** ± 0,02 |
| J. Wu | 0,94 ± 0,02 | 0,94 ± 0,02 | 0,91 ± 0,02 | 0,92 ± 0,01 | **0,99** ± 0,01 | **0,99** ± 0,00 |
| J. Zhang | 0,85 ± 0,03 | 0,81 ± 0,05 | 0,86 ± 0,03 | 0,82 ± 0,04 | **0,95** ± 0,01 | **0,93** ± 0,01 |
| L. Zhang | 0,87 ± 0,03 | 0,83 ± 0,05 | 0,90 ± 0,02 | 0,90 ± 0,03 | **0,95** ± 0,01 | **0,94** ± 0,02 |
| M. Chen | 0,89 ± 0,03 | 0,87 ± 0,03 | 0,94 ± 0,01 | 0,96 ± 0,01 | **0,98** ± 0,01 | **0,99** ± 0,01 |
| M. Pedram | **1,00** ± 0,00 | **1,00** ± 0,00 | **1,00** ± 0,00 | **1,00** ± 0,00 | **1,00** ± 0,00 | **1,00** ± 0,00 |
| M. Vardi | **0,99** ± 0,00 | **0,99** ± 0,00 | 0,89 ± 0,04 | 0,89 ± 0,05 | 0,91 ± 0,05 | 0,90 ± 0,06 |
| N. Jennings | **1,00** ± 0,00 | **1,00** ± 0,00 | **1,00** ± 0,00 | **1,00** ± 0,00 | **0,99** ± 0,01 | **0,99** ± 0,01 |
| N. Jha | **1,00** ± 0,00 | **1,00** ± 0,00 | **1,00** ± 0,00 | **1,00** ± 0,00 | **1,00** ± 0,00 | **1,00** ± 0,00 |
| N. Lynch | **1,00** ± 0,00 | **1,00** ± 0,00 | **1,00** ± 0,00 | **1,00** ± 0,00 | **1,00** ± 0,00 | **1,00** ± 0,00 |
| P. Yu | 0,98 ± 0,01 | 0,98 ± 0,01 | 0,96 ± 0,02 | 0,96 ± 0,02 | **1,00** ± 0,01 | **0,99** ± 0,01 |
| Q. Yang | **0,92** ± 0,03 | 0,94 ± 0,03 | 0,91 ± 0,01 | 0,96 ± 0,01 | **0,95** ± 0,01 | **0,98** ± 0,01 |
| S. Jajodia | 0,95 ± 0,01 | 0,95 ± 0,02 | 0,89 ± 0,02 | 0,92 ± 0,02 | **0,96** ± 0,01 | **0,97** ± 0,01 |
| S. Kim | 0,84 ± 0,04 | 0,71 ± 0,11 | 0,87 ± 0,02 | 0,84 ± 0,04 | **0,96** ± 0,01 | **0,95** ± 0,02 |
| S. Lee | 0,81 ± 0,03 | 0,70 ± 0,08 | 0,84 ± 0,03 | 0,83 ± 0,05 | **0,98** ± 0,01 | **0,98** ± 0,01 |
| T. Henzinger | **1,00** ± 0,00 | **1,00** ± 0,00 | **1,00** ± 0,00 | **1,00** ± 0,00 | **1,00** ± 0,00 | **1,00** ± 0,00 |
| W. Wang | 0,81 ± 0,04 | 0,68 ± 0,06 | 0,80 ± 0,02 | 0,76 ± 0,04 | **0,91** ± 0,02 | **0,89** ± 0,03 |
| X. Li | 0,80 ± 0,02 | 0,74 ± 0,05 | 0,83 ± 0,03 | **0,79** ± 0,04 | **0,90** ± 0,01 | **0,82** ± 0,03 |
| X. Zhou | 0,92 ± 0,03 | 0,91 ± 0,04 | 0,91 ± 0,04 | 0,89 ± 0,05 | **0,98** ± 0,01 | **0,98** ± 0,01 |
| Y. Chen | 0,83 ± 0,02 | 0,71 ± 0,06 | 0,85 ± 0,02 | 0,79 ± 0,02 | **0,94** ± 0,01 | **0,91** ± 0,03 |
| Y. Li | 0,86 ± 0,02 | 0,76 ± 0,05 | 0,84 ± 0,03 | 0,76 ± 0,05 | **0,93** ± 0,01 | **0,86** ± 0,02 |
| Y. Liu | 0,89 ± 0,02 | 0,89 ± 0,02 | 0,92 ± 0,01 | 0,94 ± 0,01 | **0,97** ± 0,01 | **0,97** ± 0,01 |
| Y. Wang | 0,85 ± 0,02 | 0,80 ± 0,03 | 0,87 ± 0,01 | 0,86 ± 0,02 | **0,93** ± 0,01 | **0,91** ± 0,02 |
| Y. Yang | 0,90 ± 0,02 | 0,89 ± 0,03 | 0,89 ± 0,03 | 0,89 ± 0,04 | **0,97** ± 0,01 | **0,96** ± 0,01 |
| Y. Zhang | 0,84 ± 0,02 | 0,75 ± 0,03 | 0,87 ± 0,03 | **0,84** ± 0,06 | **0,93** ± 0,02 | **0,88** ± 0,04 |
| Z. Zhang | 0,87 ± 0,02 | 0,84 ± 0,04 | 0,86 ± 0,01 | 0,84 ± 0,02 | **0,93** ± 0,01 | **0,93** ± 0,02 |
| **Média** | 0,90 ± 0,02 | 0,86 ± 0,04 | 0,90 ± 0,02 | 0,89 ± 0,03 | **0,96** ± 0,01 | **0,95** ± 0,02 |

formação de um alto número de *clusters* com apenas uma citação.

Nos 40 maiores grupos da KISTI, o método DICS obteve resultados menores apenas nos grupos *J. Halpern*, *M. Vardi* e *N. Jennings*, quando comparado com o método SLAND, e no grupo *N. Jennings* quando comparado com o método Cosine. Em todos estes casos, os grupos possuem apenas um autor, ou dois autores sendo que um

deles possui poucas citações. Nestes grupos, o método SLAND gera um alto número de regras de associação relacionadas ao autor mais prolífico, portanto nenhum novo *cluster* é criado e todas as citações são associadas ao maior *cluster*. Já no método DICS, com o aumento do número de citações, os valores das similaridades diminuem (devido à redução dos valores das distribuições dos termos nos grupos), consequentemente, novos *clusters* são criados. Entretanto, este comportamento é desejado na maioria dos cenários que envolvem um grande número de citações, pois ajuda a preservar a pureza dos *clusters* já identificados. Em relação à métrica K, houve apenas 11 empates estatísticos com o método SLAND e 7 com o método Cosine. Os maiores ganhos alcançados foram de 20% e 15,6% considerando a métrica K, e de 49,6% a 32,2% considerando a métrica pF1, comparando com os métodos SLAND e Cosine respectivamente (grupos *S. Lee* e *J. Kim*).

#### 4.4.2.1 Inclusão de Características Baseadas em Coocorrência de Palavras

A estratégia de incorporação de características baseadas em coocorrência de palavras proposta por Figueiredo et al. [2011] foi adaptada para ser aplicada no problema de desambiguação de nomes. Cada atributo da citação com mais de um termo (lista de coautores, título e local da publicação) foi expandido com a incorporação de *c-features* definidas pela coocorrência de dois termos conforme descrito na Seção 3.4.

As Tabelas 4.7 e 4.8 mostram os melhores resultados obtidos em cada coleção, considerando a expansão dos atributos coautores, título e local de publicação, *DICS + (c, t, v)*; somente dos atributos coautores e título, *DICS + (c, t)*; e somente do atributo lista de coautores, *DICS + (c)*. Os melhores valores para o parâmetro $\tau$ foram: 0 para todas as variações nas coleções BDBComp e KISTI, e 0, 0,5 e 0,75 para as estratégias *DICS + (c, t, v)*, *DICS + (c, t)* e *DICS + (c)*, respectivamente, na coleção DBLP.

Foram obtidas melhorias apenas na coleção BDBComp (o maior ganho foi de aproximadamente 1,1% em relação a métrica pF1), entretanto todas as médias foram estatisticamente iguais. Comparando as três estratégias de expansão, os melhores resultados foram obtidos com a expansão dos termos dos atributos lista de coautores e título da publicação. No caso da coleção KISTI, o atributo local de publicação possui apenas um termo que representa o nome abreviado do local de publicação, portanto não gera *c-features*.

Estes resultados podem ser explicados por dois motivos: (i) os resultados obtidos com o modelo original (sem a utilização das *c-features*) já estão próximos ao limite que pode ser alcançado com um método de classificação baseado em uma função de similaridade linear, ou (ii) devido ao pequeno número de termos compartilhados entre

Tabela 4.7: Valores médios da métrica K obtidos com a utilização de *co-features* no treino e teste.

| **Método** | **Coleção** | | |
|---|---|---|---|
| | **DBLP** | **BDBComp** | **KISTI** |
| DICS | **0,919** ± 0,026 | **0,956** ± 0,024 | **0,952** ± 0,002 |
| DICS + (c) | **0,914** ± 0,031 | **0,957** ± 0,022 | **0,952** ± 0,002 |
| DICS + (c, t) | **0,916** ± 0,030 | **0,959** ± 0,022 | **0,952** ± 0,002 |
| DICS + (c, t, v) | **0,901** ± 0,039 | **0,957** ± 0,020 | **0,952** ± 0,002 |

Tabela 4.8: Valores médios da métrica pF1 obtidos com a utilização de *co-features* no treino e teste.

| **Método** | **Coleção** | | |
|---|---|---|---|
| | **DBLP** | **BDBComp** | **KISTI** |
| DICS | **0,916** ± 0,032 | **0,728** ± 0,143 | **0,835** ± 0,009 |
| DICS + (c) | **0,914** ± 0,035 | **0,732** ± 0,145 | **0,835** ± 0,009 |
| DICS + (c, t) | **0,916** ± 0,034 | **0,736** ± 0,148 | **0,835** ± 0,009 |
| DICS + (c, t, v) | **0,903** ± 0,041 | **0,728** ± 0,157 | **0,835** ± 0,009 |

os atributos da citações, a incorporação de *c-features* tende a afetar apenas as citações que já são corretamente classificadas sem a utilização das *c-features*. Observe que uma *c-feature* só é gerada quando dois ou mais termos de um atributo coocorrem em duas ou mais citações.

Para avaliar como a expansão dos termos afeta os valores de similaridade e a precisão das classificações, foram registradas as taxas de acertos obtidas em diferentes níveis de similaridade para cada variação do método DICS, desconsiderando os procedimentos de atualização do conjunto de treinamento. Estes valores são mostrados nas Figuras 4.1, 4.2 e 4.3 para as coleções DBLP, BDBComp e KISTI respectivamente.

É possível observar que as taxas de acerto para cada nível de similaridade diminuem com a utilização de *c-features*, especialmente com a expansão do atributo local de publicação. Pequenas melhorias na taxa de acerto em relação ao método DICS podem ser verificadas quando os valores das similaridades são maiores do que 4. Entretanto, nestes casos as taxas médias de acertos já são superiores a 90%. Esse comportamento suporta a segunda hipótese levantada anteriormente.

## 4.5 Comparação dos Métodos Não Supervisionados

Nesta seção são apresentados os resultados dos experimentos utilizados para avaliar o método DICS quando não há um conjunto de treinamento. Em um cenário não

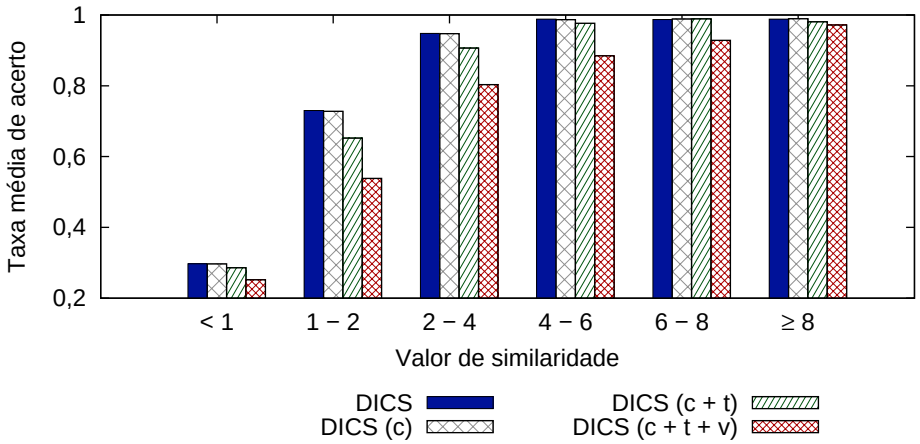



Figura 4.1: Taxas de acerto por valor de similaridade obtidas com a utilização de *c-features* na DBLP.

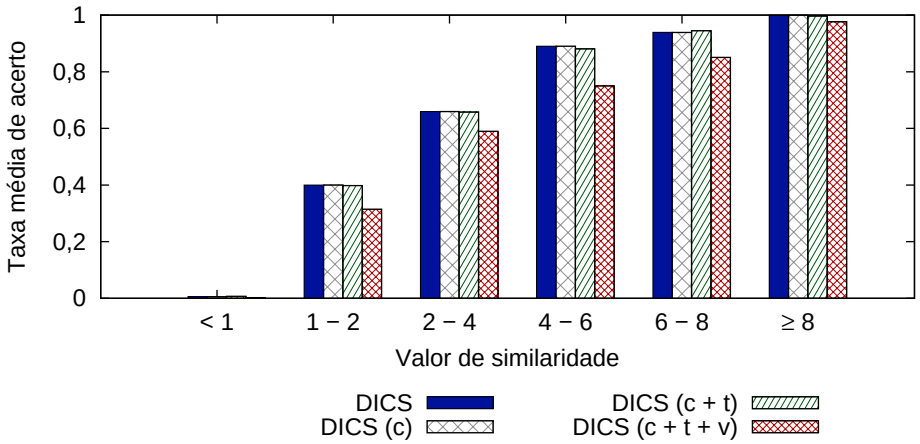



Figura 4.2: Taxas de acerto por valor de similaridade obtidas com a utilização de *c-features* na BDBComp.

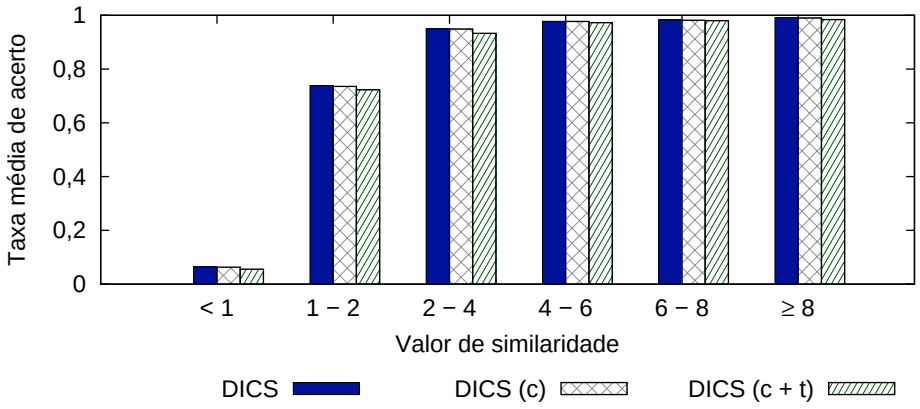



Figura 4.3: Taxa de acerto por valor de similaridade obtidos com a utilização de *c-features* na KISTI.

supervisionado, o método de desambiguação deve ser baseado em uma técnica de clusterização, ou ser capaz de identificar automaticamente a presença de novos autores. A seguir é detalhada como foi realizado a configuração de cada método e a execução dos experimentos.

### 4.5.1 Configuração Experimental

Para estes experimentos foram utilizadas as mesmas divisões de teste descritas na seção anterior. Os *baselines* executados foram: SAND, HHC e LASVM-DBSCAN. Para a definição dos valores dos parâmetros do método proposto, foi utilizado o procedimento descrito na Seção 3.3. Para encontrar o valor do parâmetro $\gamma$ foram utilizadas seis iterações (entrada $k$ do Algoritmo 2)[1].

O método SAND constrói automaticamente um conjunto de treinamento para o classificador SLAND utilizando apenas uma função de comparação de nomes para definir a dissimilaridade entre *clusters*, portanto não possui parâmetros. Para o método HHC, foram usados os mesmos valores dos parâmetros utilizados em [Cota et al., 2010]. Para o método LASVM-DBSCAN, as funções de similaridade foram aprendidas pelo LASVM utilizando as divisões de treino de cada grupo ambíguo. Observe que, para este último método, qualquer conjunto de treinamento poderia ser utilizado, pois ele define apenas a função de similaridade, enquanto a clusterização é realizada pelo DBSCAN.

### 4.5.2 Resultados

As Tabelas 4.9 e 4.10 mostram os resultados obtidos por cada método em cada coleção em relação às métricas K e pF1, respectivamente. O método DICS obteve os melhores resultados médios nas coleções DBLP e KISTI. Na coleção BDBComp, o método HHC obteve a maior média em relação à métrica pF1, entretanto, houve empate estatístico entre DICS e os métodos SAND e HHC. Em relação à métrica K, não houve diferenças estatísticas na BDBComp.

Na coleção DBLP os ganhos obtidos em relação a métrica K foram superiores a 14%, chegando a 64,9%, comparando com o método LASVM-DBSCAN. Em relação à métrica pF1, o resultado foi cerca 23% melhor se comparado com o segundo melhor *baseline* nesta coleção (HHC). Comparando com os resultados obtidos com a utilização do conjunto de treinamento, as quedas nos valores foram de aproximadamente 13,9% e 19,5% em relação às métricas K e pF1 respectivamente.

[1]Como o intervalo de busca do valor de $\gamma$ é normalmente pequeno, não é necessário utilizar um alto número de iterações.

Tabela 4.9: Valores médios da métrica K obtidos pelos métodos não supervisionados.

| **Método** | **Coleção** | | |
|---|---|---|---|
| | **DBLP** | **BDBComp** | **KISTI** |
| DICS | **0,791** ± 0,059 | **0,944** ± 0,025 | **0,942** ± 0,003 |
| SAND | 0,674 ± 0,091 | **0,942** ± 0,022 | 0,892 ± 0,003 |
| HHC | 0,692 ± 0,084 | **0,937** ± 0,021 | 0,862 ± 0,003 |
| LASVM-DBSCAN | 0,479 ± 0,097 | **0,883** ± 0,042 | 0,858 ± 0,004 |

Tabela 4.10: Valores médios da métrica pF1 obtidos pelos métodos não supervisionados.

| **Método** | **Coleção** | | |
|---|---|---|---|
| | **DBLP** | **BDBComp** | **KISTI** |
| DICS | **0,737** ± 0,110 | **0,712** ± 0,175 | **0,831** ± 0,008 |
| SAND | 0,556 ± 0,151 | **0,691** ± 0,155 | 0,734 ± 0,008 |
| HHC | 0,598 ± 0,143 | **0,722** ± 0,156 | 0,671 ± 0,008 |
| LASVM-DBSCAN | 0,319 ± 0,092 | 0,333 ± 0,231 | 0,638 ± 0,011 |

Na coleção KISTI, DICS obteve ganhos superiores a 5,5% se comparado ao método SAND (melhor *baseline*). Em relação aos métodos HHC e LASVM-DBSCAN, os ganhos foram superiores a 9% e 23% considerando as métricas K e pF1, respectivamente. Comparando com os valores obtidos com a utilização dos conjuntos de treinamento, a diferença foi menor do que 1,1%. Na coleção BDBComp, a queda dos valores também foi pequena, cerca de 1,3% e 2,2% para as métricas K e pF1 respectivamente. Na DBLP, a maior queda de qualidade se deve a menor quantidade de termos discriminativos compartilhados entre as citações. Estes resultados mostram a eficácia do procedimento proposto para encontrar automaticamente os valores dos parâmetros.

Nas Tabelas 4.11 e 4.12 são apresentados os valores de cada métrica obtidos pelos métodos HHC, SAND e DICS em cada grupo ambíguo das coleções DBLP, BDBComp. Nestas coleções, o método proposto obteve melhores resultados ou foi estatisticamente empatado com os baselines. Na DBLP, os métodos HHC e SAND obtiveram melhores resultados apenas no grupo *D. Johnson* (com ganhos de aproximadamente 5,5% em relação ao método DICS considerando a métrica pF1). Nos outros grupos, DICS obteve ganhos de até 103% e 59% em relação aos métodos SAND e HHC, respectivamente, considerando a métrica K (grupos *M. Jones* e *Y. Chen*). A grande diferença de desempenho nos grupos *J. Lee*, *M. Jones*, *M. Miller*, *S. Lee* e *Y. Chen*, quando são comparados os métodos DICS e SAND, ocorre devido à ausência, ou pequeno número, de referências com nomes completos. Como descrito na Seção 2.1.2, a similaridade dos nomes é utilizada pelo SAND como critério na seleção dos *clusters* que irão compor o conjunto de treinamento do SLAND. Nestes grupos ambíguos, o treinamento do classi-

Tabela 4.11: Resultados obtidos pelos métodos HHC, SAND e DICS em cada grupo ambíguo na coleção DBLP utilizando apenas as divisões de teste.

| **Grupo** | **HHC** | | **SAND** | | **DICS** | |
|---|---|---|---|---|---|---|
| **ambíguo** | **K** | **pF1** | **K** | **pF1** | **K** | **pF1** |
| A. Gupta | 0,78 ± 0,03 | 0,78 ± 0,05 | **0,85** ± 0,01 | **0,86** ± 0,02 | **0,85** ± 0,02 | **0,87** ± 0,03 |
| A. Kumar | 0,75 ± 0,04 | 0,71 ± 0,09 | 0,75 ± 0,05 | 0,66 ± 0,08 | **0,87** ± 0,03 | **0,85** ± 0,05 |
| C. Chen | 0,55 ± 0,02 | 0,36 ± 0,03 | 0,61 ± 0,02 | 0,41 ± 0,03 | **0,67** ± 0,02 | **0,52** ± 0,05 |
| D. Johnson | **0,75** ± 0,03 | **0,76** ± 0,04 | **0,75** ± 0,02 | **0,77** ± 0,03 | **0,74** ± 0,04 | **0,72** ± 0,07 |
| J. Lee | 0,46 ± 0,02 | 0,13 ± 0,02 | 0,45 ± 0,02 | 0,08 ± 0,01 | **0,59** ± 0,02 | **0,27** ± 0,06 |
| J. Martin | 0,87 ± 0,01 | 0,81 ± 0,04 | 0,86 ± 0,03 | 0,79 ± 0,05 | **0,93** ± 0,02 | **0,89** ± 0,04 |
| J. Robinson | 0,77 ± 0,04 | 0,72 ± 0,07 | 0,77 ± 0,04 | 0,72 ± 0,08 | **0,90** ± 0,01 | **0,91** ± 0,03 |
| J. Smith | 0,69 ± 0,03 | 0,70 ± 0,04 | 0,73 ± 0,05 | 0,67 ± 0,10 | **0,82** ± 0,02 | **0,83** ± 0,03 |
| K. Tanaka | 0,82 ± 0,04 | 0,82 ± 0,05 | **0,84** ± 0,05 | **0,82** ± 0,08 | **0,87** ± 0,03 | **0,89** ± 0,03 |
| M. Brown | **0,84** ± 0,04 | **0,82** ± 0,06 | **0,83** ± 0,04 | **0,82** ± 0,05 | **0,86** ± 0,03 | **0,86** ± 0,05 |
| M. Jones | 0,66 ± 0,03 | 0,47 ± 0,05 | 0,38 ± 0000 | 0,25 ± 0,01 | **0,78** ± 0,03 | **0,74** ± 0,04 |
| M. Miller | **0,87** ± 0,06 | **0,88** ± 0,07 | 0,66 ± 0,05 | 0,58 ± 0,06 | **0,85** ± 0,03 | **0,85** ± 0,04 |
| S. Lee | 0,40 ± 0,02 | 0,15 ± 0,01 | 0,39 ± 0,01 | 0,10 ± 0,01 | **0,57** ± 0,03 | **0,36** ± 0,11 |
| Y. Chen | 0,49 ± 0,02 | 0,27 ± 0,03 | 0,55 ± 0,02 | 0,24 ± 0,03 | **0,78** ± 0,06 | **0,76** ± 0,13 |
| **Média** | 0,69 ± 0,08 | 0,60 ± 0,14 | 0,67 ± 0,09 | 0,56 ± 0,15 | **0,79** ± 0,06 | **0,74** ± 0,11 |

Tabela 4.12: Resultados obtidos pelos métodos HHC, SAND e DICS em cada grupo ambíguo na coleção BDBComp utilizando apenas as divisões de teste.

| **Grupo** | **HHC** | | **SAND** | | **DICS** | |
|---|---|---|---|---|---|---|
| **ambíguo** | **K** | **pF1** | **K** | **pF1** | **K** | **pF1** |
| A. Oliveira | **0,93** ± 0,04 | **0,91** ± 0,06 | **0,92** ± 0,04 | **0,86** ± 0,09 | **0,96** ± 0,03 | **0,93** ± 0,06 |
| A. Silva | **0,97** ± 0,01 | **0,93** ± 0,04 | **0,98** ± 0,02 | **0,93** ± 0,08 | **0,98** ± 0,02 | **0,96** ± 0,04 |
| F. Silva | **0,94** ± 0,03 | **0,50** ± 0,23 | **0,97** ± 0,02 | **0,58** ± 0,31 | **0,97** ± 0,02 | **0,65** ± 0,20 |
| J. Oliveira | **0,90** ± 0,03 | **0,78** ± 0,06 | **0,88** ± 0,04 | **0,76** ± 0,09 | **0,87** ± 0,02 | **0,72** ± 0,06 |
| J. Silva | **0,95** ± 0,03 | **0,90** ± 0,07 | **0,96** ± 0,03 | **0,91** ± 0,09 | **0,94** ± 0,03 | **0,87** ± 0,08 |
| J. Souza | **0,94** ± 0,02 | **0,89** ± 0,05 | **0,93** ± 0,05 | **0,87** ± 0,11 | **0,95** ± 0,02 | **0,91** ± 0,04 |
| L. Silva | 0,90 ± 0,02 | **0,69** ± 0,09 | **0,91** ± 0,03 | **0,68** ± 0,15 | **0,94** ± 0,02 | **0,80** ± 0,10 |
| M. Silva | **0,99** ± 0,02 | **0,83** ± 0,23 | **0,97** ± 0,02 | **0,65** ± 0,27 | **0,99** ± 0,02 | **0,70** ± 0,29 |
| R. Santos | **0,95** ± 0,03 | **0,45** ± 0,34 | **0,95** ± 0,01 | **0,30** ± 0,29 | **0,96** ± 0,03 | **0,33** ± 0,32 |
| R. Silva | 0,89 ± 0,03 | **0,34** ± 0,21 | **0,94** ± 0,03 | **0,39** ± 0,23 | **0,91** ± 0,05 | **0,25** ± 0,17 |
| **Média** | **0,94** ± 0,02 | **0,72** ± 0,16 | **0,94** ± 0,02 | **0,69** ± 0,16 | **0,94** ± 0,03 | **0,71** ± 0,18 |

ficador SLAND não é suficiente para permitir a correta desambiguação de grande parte das citações.

A Tabela 4.13 mostra os resultados obtidos pelos métodos HHC, SAND e DICS nos 40 maiores grupos da coleção KISTI. Nestes grupos, DICS foi inferior ao SAND apenas nos grupos ambíguos *J. Halpern* e *L. Zhang*, com diferença de desempenho de até 6,7% considerando a métrica K. Comparando com o método HHC, DICS foi inferior, mas estatisticamente igual, nos grupos *J. Kim* e *L. Zhang*, com a maior diferença de desempenho igual a 6,3% considerando a métrica K. Nos outros grupos, os ganhos

Tabela 4.13: Resultados obtidos pelos métodos HHC, SAND e DICS nos 40 maiores grupo ambíguos da coleção KISTI utilizando apenas as divisões de teste.

| **Grupo** | **HHC** | | **SAND** | | **DICS** | |
|---|---|---|---|---|---|---|
| **ambíguo** | **K** | **pF1** | **K** | **pF1** | **K** | **pF1** |
| A. Choudhary | 0,92 ± 0,03 | 0,92 ± 0,04 | **1,00** ± 0000 | **1,00** ± 0000 | **1,00** ± 0000 | **1,00** ± 0000 |
| A. Gupta | 0,82 ± 0,03 | 0,84 ± 0,05 | 0,83 ± 0,02 | 0,86 ± 0,03 | **0,92** ± 0,01 | **0,92** ± 0,02 |
| D. Eppstein | 0,84 ± 0,04 | 0,82 ± 0,04 | **1,00** ± 0000 | **1,00** ± 0000 | **1,00** ± 0000 | **1,00** ± 0000 |
| D. Lee | 0,77 ± 0,02 | 0,63 ± 0,05 | 0,82 ± 0,03 | 0,76 ± 0,06 | **0,94** ± 0,01 | **0,94** ± 0,01 |
| H. Chen | 0,82 ± 0,02 | 0,63 ± 0,06 | 0,80 ± 0,02 | 0,58 ± 0,06 | **0,88** ± 0,02 | **0,83** ± 0,04 |
| H. Wang | 0,89 ± 0,02 | 0,88 ± 0,03 | 0,87 ± 0,03 | 0,82 ± 0,08 | **0,93** ± 0,01 | **0,92** ± 0,02 |
| J. Chen | 0,84 ± 0,02 | 0,61 ± 0,08 | 0,87 ± 0,04 | 0,78 ± 0,09 | **0,95** ± 0,01 | **0,98** ± 0,01 |
| J. Halpern | 0,78 ± 0,04 | 0,77 ± 0,04 | **0,92** ± 0,01 | **0,92** ± 0,01 | **0,86** ± 0,15 | **0,84** ± 0,18 |
| J. Kim | **0,91** ± 0,01 | **0,79** ± 0,05 | 0,90 ± 0,02 | 0,70 ± 0,08 | **0,90** ± 0,01 | **0,79** ± 0,04 |
| J. Lee | 0,84 ± 0,01 | 0,61 ± 0,03 | 0,84 ± 0,02 | 0,61 ± 0,05 | **0,91** ± 0,01 | **0,83** ± 0,04 |
| J. Li | 0,91 ± 0,02 | 0,87 ± 0,05 | 0,86 ± 0,03 | 0,75 ± 0,09 | **0,96** ± 0,01 | **0,95** ± 0,02 |
| J. Liu | 0,83 ± 0,02 | 0,61 ± 0,07 | 0,85 ± 0,04 | 0,70 ± 0,12 | **0,95** ± 0,01 | **0,96** ± 0,01 |
| J. Mitchell | 0,74 ± 0,03 | 0,68 ± 0,05 | 0,81 ± 0,09 | 0,78 ± 0,11 | **0,99** ± 0000 | **0,99** ± 0000 |
| J. Smith | 0,85 ± 0,03 | 0,84 ± 0,04 | 0,80 ± 0,04 | 0,70 ± 0,12 | **0,90** ± 0,01 | **0,92** ± 0,01 |
| J. Wang | 0,83 ± 0,02 | 0,71 ± 0,04 | 0,85 ± 0,02 | 0,77 ± 0,05 | **0,91** ± 0,01 | **0,86** ± 0,02 |
| J. Wu | 0,86 ± 0,05 | 0,84 ± 0,08 | 0,87 ± 0,04 | 0,85 ± 0,05 | **0,99** ± 0,01 | **0,99** ± 0000 |
| J. Zhang | 0,87 ± 0,01 | 0,74 ± 0,04 | 0,85 ± 0,02 | 0,72 ± 0,05 | **0,90** ± 0,01 | **0,79** ± 0,03 |
| L. Zhang | **0,87** ± 0,02 | **0,80** ± 0,06 | **0,85** ± 0,02 | **0,76** ± 0,06 | 0,82 ± 0,02 | 0,66 ± 0,04 |
| M. Chen | 0,83 ± 0,02 | 0,82 ± 0,05 | 0,87 ± 0,02 | 0,92 ± 0,03 | **0,93** ± 0,01 | **0,96** ± 0,01 |
| M. Pedram | 0,93 ± 0,03 | 0,93 ± 0,04 | **1,00** ± 0000 | **1,00** ± 0000 | **1,00** ± 0000 | **1,00** ± 0000 |
| M. Vardi | 0,68 ± 0,07 | 0,62 ± 0,09 | **0,99** ± 0,01 | **0,99** ± 0,01 | **0,99** ± 0,01 | **0,99** ± 0,01 |
| N. Jennings | 0,90 ± 0,02 | 0,89 ± 0,02 | 0,92 ± 0,04 | 0,91 ± 0,05 | **0,98** ± 0,01 | **0,98** ± 0,01 |
| N. Jha | 0,98 ± 0,02 | 0,97 ± 0,02 | **1,00** ± 0000 | **1,00** ± 0000 | **1,00** ± 0000 | **1,00** ± 0000 |
| N. Lynch | 0,80 ± 0,03 | 0,77 ± 0,03 | **1,00** ± 0000 | **1,00** ± 0000 | **1,00** ± 0000 | **1,00** ± 0000 |
| P. Yu | 0,91 ± 0,03 | 0,90 ± 0,03 | 0,90 ± 0,03 | 0,89 ± 0,04 | **0,99** ± 0,01 | **0,99** ± 0,01 |
| Q. Yang | 0,74 ± 0,04 | 0,64 ± 0,07 | 0,83 ± 0,07 | 0,81 ± 0,09 | **0,96** ± 0,01 | **0,98** ± 0,01 |
| S. Jajodia | 0,88 ± 0,02 | 0,89 ± 0,02 | **0,94** ± 0,01 | **0,94** ± 0,01 | **0,94** ± 0,01 | **0,94** ± 0,01 |
| S. Kim | 0,89 ± 0,01 | 0,81 ± 0,03 | 0,90 ± 0,02 | 0,83 ± 0,04 | **0,95** ± 0,01 | **0,95** ± 0,02 |
| S. Lee | 0,90 ± 0,02 | 0,75 ± 0,07 | 0,86 ± 0,02 | 0,69 ± 0,07 | **0,97** ± 0,02 | **0,95** ± 0,05 |
| T. Henzinger | 0,95 ± 0,02 | 0,95 ± 0,02 | **1,00** ± 0000 | **1,00** ± 0000 | **1,00** ± 0000 | **1,00** ± 0000 |
| W. Wang | **0,79** ± 0,02 | **0,64** ± 0,05 | **0,78** ± 0,03 | **0,61** ± 0,05 | **0,79** ± 0,02 | 0,59 ± 0,04 |
| X. Li | 0,81 ± 0,02 | **0,68** ± 0,04 | 0,77 ± 0,03 | 0,59 ± 0,06 | **0,83** ± 0,02 | **0,67** ± 0,04 |
| X. Zhou | 0,89 ± 0,02 | 0,87 ± 0,04 | 0,88 ± 0,04 | 0,84 ± 0,06 | **0,98** ± 0,01 | **0,98** ± 0,01 |
| Y. Chen | 0,82 ± 0,02 | 0,64 ± 0,04 | 0,82 ± 0,02 | 0,68 ± 0,05 | **0,90** ± 0,01 | **0,83** ± 0,02 |
| Y. Li | **0,89** ± 0,01 | **0,76** ± 0,03 | 0,87 ± 0,01 | 0,72 ± 0,04 | **0,90** ± 0,01 | **0,78** ± 0,03 |
| Y. Liu | 0,85 ± 0,02 | 0,76 ± 0,04 | **0,87** ± 0,02 | **0,81** ± 0,06 | **0,89** ± 0,01 | **0,86** ± 0,02 |
| Y. Wang | 0,82 ± 0,02 | 0,62 ± 0,04 | 0,81 ± 0,02 | 0,60 ± 0,05 | **0,89** ± 0,01 | **0,85** ± 0,01 |
| Y. Yang | 0,77 ± 0,01 | 0,54 ± 0,03 | 0,83 ± 0,06 | 0,69 ± 0,12 | **0,98** ± 0,01 | **0,98** ± 0,01 |
| Y. Zhang | **0,84** ± 0,02 | 0,67 ± 0,05 | **0,82** ± 0,03 | 0,62 ± 0,07 | **0,86** ± 0,01 | **0,74** ± 0,02 |
| Z. Zhang | 0,86 ± 0,02 | 0,75 ± 0,05 | 0,83 ± 0,03 | 0,71 ± 0,07 | **0,93** ± 0,01 | **0,91** ± 0,01 |
| **Média** | 0,85 ± 0,02 | 0,76 ± 0,04 | 0,88 ± 0,02 | 0,80 ± 0,04 | **0,93** ± 0,02 | **0,90** ± 0,03 |

obtidos foram de até 22,3% comparando com o método SAND (grupo *J. Mitchell*), e 45,3% comparando com o método HHC (grupo *M. Vardi*), também em relação à métrica K.

#### 4.5.2.1 Inclusão de Características Baseadas em Coocorrência de Palavras

O procedimento de expansão de termos também pode ser aplicado em um cenário não supervisionado. Neste caso, entretanto, não é possível realizar a seleção de termos que irão compor as *c-features*, nem o descarte das características pouco discriminativas. Todas as combinações possíveis de dois termos são utilizadas para formar as *c-features* e expandir o conjunto de termos de cada atributo das citações.

As Tabelas 4.14 e 4.15 mostram os resultados obtidos com a inclusão das *c-features* utilizando apenas as divisões de teste das coleções DBLP, BDBComp e KISTI, em relação às métricas K e pF1 respectivamente. Nestes experimentos foram utilizadas as mesmas variações do método DICS descritas na Seção 4.4.2.1. Em todas as coleções houve redução dos valores de K e pF1, na medida em que foram incorporadas as *c-features* em cada atributo. As diferenças de desempenho foram significativas na coleção KISTI em relação a todas as variações do método DICS e na coleção DBLP em relação à variação *DICS + (c, t, v)* considerando a métrica K. Estes resultados mostram a importância dos procedimentos de seleção de *c-features* propostos por Figueiredo et al. [2011] a fim de se obter ganhos com a utilização deste tipo de característica.

Tabela 4.14: Valores da métrica K obtidos com a utilização de *co-features* no teste.

| **Método** | **Coleção** | | |
|---|---|---|---|
| | **DBLP** | **BDBComp** | **KISTI** |
| DICS | **0,791** $\pm$ 0,059 | **0,944** $\pm$ 0,025 | **0,942** $\pm$ 0,003 |
| DICS + (c) | **0,775** $\pm$ 0,061 | **0,944** $\pm$ 0,025 | 0,939 $\pm$ 0,003 |
| DICS + (c, t) | **0,774** $\pm$ 0,052 | **0,944** $\pm$ 0,025 | 0,929 $\pm$ 0,003 |
| DICS + (c, t, v) | 0,745 $\pm$ 0,059 | **0,942** $\pm$ 0,021 | 0,929 $\pm$ 0,003 |

Tabela 4.15: Valores da métrica pF1 obtidos com a utilização de *co-features* no teste.

| **Método** | **Coleção** | | |
|---|---|---|---|
| | **DBLP** | **BDBComp** | **KISTI** |
| DICS | **0,737** $\pm$ 0,110 | **0,712** $\pm$ 0,175 | **0,831** $\pm$ 0,008 |
| DICS + (c) | **0,712** $\pm$ 0,112 | **0,711** $\pm$ 0,175 | 0,827 $\pm$ 0,008 |
| DICS + (c, t) | **0,723** $\pm$ 0,078 | **0,708** $\pm$ 0,174 | 0,812 $\pm$ 0,008 |
| DICS + (c, t, v) | **0,680** $\pm$ 0,087 | **0,706** $\pm$ 0,168 | 0,812 $\pm$ 0,008 |

## 4.6 Comparação dos Métodos Incrementais

Nesta seção são apresentados os experimentos realizados para avaliar o método proposto considerando cenários nas quais a desambiguação ocorre de forma incremental. Foram utilizados como *baselines* os métodos desenvolvidos por Carvalho et al. [2011] e Esperidião et al. [2014]. Também são apresentadas uma análise das capacidades do método DICS, a comparação dos tempos de execução dos métodos incrementais e uma análise de sensibilidade aos valores dos parâmetros.

### 4.6.1 Configuração Experimental

Para simular a evolução de BDs ao longo do tempo, as citações das coleções BDBComp e KISTI foram ordenadas em ordem cronológica e os métodos INDi, MINDi e DICS foram aplicados sobre os conjuntos de citações de cada ano. Nestes experimentos foram utilizadas as citações do período de 1987 a 2007. De maneira similar, os métodos foram executados nas coleções sintéticas nas quais cada carga representa um ano. Para possibilitar comparações estatísticas entre os resultados, foram geradas cinco variações das coleções BDBComp e KISTI a partir da alteração da ordem das citações dentro de cada ano. Já para as coleções sintéticas, para cada cenário simulado, foram gerados cinco conjuntos de citações utilizando diferentes sementes para os geradores de números aleatórios. Para a comparação dos valores obtidos, foi utilizado o nível de confiança de 99% no teste $t$ pareado de duas caudas com a correção Holm-Bonferroni.

Nas coleções reais (BDBComp e KISTI) foram realizadas buscas pelos melhores resultados considerando variações de 0 a 1 com passos de 0,1 para os parâmetros dos métodos INDi e MINDi. Os valores referentes aos melhores resultados encontrados são mostrados na Tabela 4.16. Nestas mesmas coleções, para o método DICS, foi realizada uma busca pelos melhores resultados variando os valores dos parâmetros de 1 a 5 em passos de 1. Para os resultados mostrados a seguir, foi utilizado a configuração que gerou os melhores resultados nas duas coleções: $w_a = 3$, $w_c = 3$, $w_t = 1$, $w_v = 1$ e $\gamma = 2$.

Nas coleções sintéticas, cada método inicia com uma coleção inicial desambiguada e então são apresentadas 10 cargas para simular a evolução do repositório ao longo de 10 anos. A carga inicial foi utilizada para realizar o ajuste dos valores dos parâmetros do método DICS conforme descrito na Seção 3.2. Para os métodos INDi e MINDi, foi realizado um procedimento baseado em validações cruzadas para encontrar os valores dos parâmetros, entretanto os resultados obtidos foram piores do que os reportados em Esperidião et al. [2014]. Portanto, foram utilizadas as mesmas configurações usadas

pelos autores destes métodos.

Nos resultados apresentados a seguir, foram utilizadas as métricas PMA, PMC e K para avaliar a qualidade de cada método em relação à coesão, à pureza e ao equilíbrio entre fragmentação e pureza, respectivamente, dos *clusters* obtidos independente da quantidade de citações encontradas nestes *clusters*.

Tabela 4.16: Melhores valores encontrados para os parâmetros dos métodos INDi e MINDi nas coleções reais.

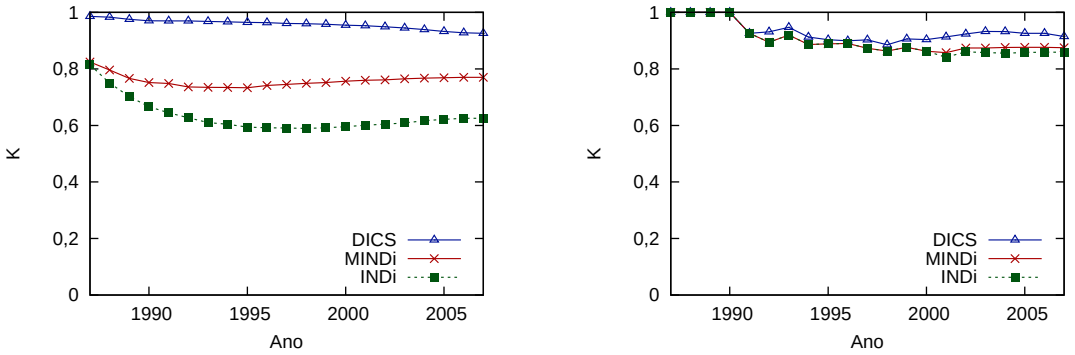
| **Método** | **Coleção** | **Parâmetros** | | | |
|---|---|---|---|---|---|
| | | $\alpha_{Title}$ | $\alpha_{Venue}$ | $\delta$ | $p$ |
| INDi | KISTI | 0,0 | 0,1 | 0,6 | - |
| | BDBComp | 0,0 | 0,1 | 0,4 | - |
| MINDi | KISTI | 0,0 | 0,0 | 0,0 | 1,0 |
| | BDBComp | 0,0 | 0,0 | 0,0 | 1,0 |

### 4.6.2 Resultados

As Figuras 4.4 e 4.5 mostram os resultados obtidos em termos da métrica K para cada ano e carga nas coleções reais e sintéticas respectivamente. As Tabelas 4.17 e 4.18 mostram os valores obtidos com a desambiguação completa destas coleções. Analisando os resultados obtidos nas coleções reais, em relação à métrica K, DICS obteve os maiores valores com ganhos de até 8,7% e 50% comparando com INDi nas coleções BDBComp e KISTI, respectivamente. Comparando com o método MINDi, os ganhos foram menores, mas com um leve aumento ao longo dos anos na KISTI. Na BDBComp, entre 1987 a 1996, a coleção possui em média apenas duas citações por ano, portanto, neste período há grandes variações nos resultados devido a pequenos erros. Após 1999, os valores da métrica K ficam estáveis para todos os métodos.

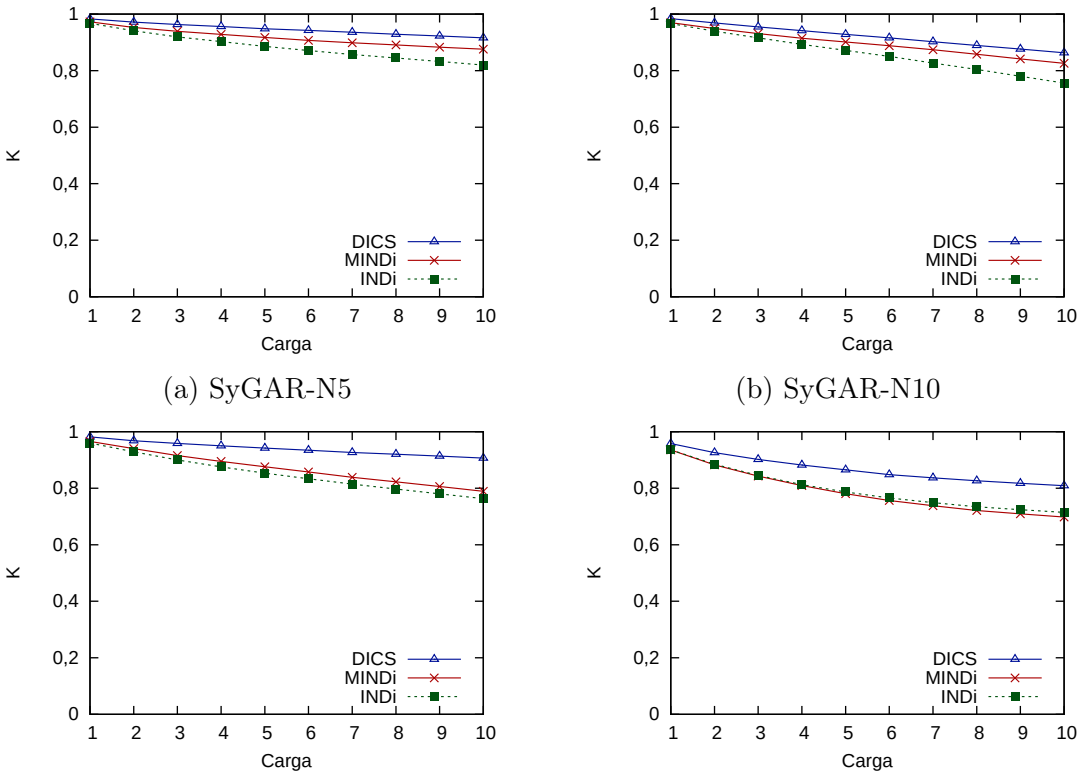
O método INDi obteve os maiores níveis de pureza (PMC) nas coleções reais, enquanto que em termos de coesão, DICS superou todos os baselines. Comparando com o método INDi, DICS foi por volta de 7,3% menos efetivo em relação à métrica PMC, mas cerca de 26% e 143% mais efetivo considerando a métrica PMA nas coleções BDBComp e KISTI, respectivamente.

Analisando os resultados obtidos nas coleções sintéticas é possível observar como cada método se comporta em relação às mudanças no repositório da BD. Em todos os cenários, DICS superou os baselines em termos de coesão (PMA) e equilíbrio entre fragmentação e pureza (K). Nos cenários simulando a introdução de novos autores, o ganho comparando com o método MINDi foi de aproximadamente 1% na primeira carga, chegando a 4,5% na última carga. Comparando com o método INDi o ganho




(a) KISTI (b) BDBComp

Figura 4.4: Valores da métrica K obtidos pelos métodos incrementais em cada ano das coleções reais.




(a) SyGAR-N5 (b) SyGAR-N10

(c) SyGAR-C10 (d) SyGAR-C50

Figura 4.5: Valores da métrica K obtidos pelos métodos incrementais em cada carga das coleções sintéticas.

Tabela 4.17: Resultados obtidos após o último ano das coleções reais.

| Coleção | Método | PMC | PMA | K |
|---|---|---|---|---|
| KISTI | DICS | 0,911 ± 0,000 | **0,967** ± 0,000 | **0,938** ± 0,000 |
| | MINDi | 0,966 ± 0,001 | 0,614 ± 0,002 | 0,770 ± 0,001 |
| | INDi | **0,982** ± 0,000 | 0,398 ± 0,004 | 0,625 ± 0,003 |
| BDBComp | DICS | 0,934 ± 0,000 | **0,933** ± 0,000 | **0,934** ± 00,00 |
| | MINDi | 0,984 ± 0,000 | 0,778 ± 0,000 | 0,875 ± 0,000 |
| | INDi | **0,997** ± 0,000 | 0,740 ± 0,015 | 0,859 ± 0,008 |

Tabela 4.18: Resultados obtidos após a última carga das coleções sintéticas.

| Coleção | Método | PMC | PMA | K |
|---|---|---|---|---|
| SyGAR-N5 | DICS | **0,922** ± 0,016 | **0,910** ± 0,019 | **0,916** ± 0,006 |
| | MINDi | **0,915** ± 0,028 | 0,839 ± 0,011 | 0,876 ± 0,011 |
| | INDi | **0,928** ± 0,018 | 0,724 ± 0,012 | 0,820 ± 0,007 |
| SyGAR-N10 | DICS | **0,826** ± 0,029 | **0,902** ± 0,016 | **0,863** ± 0,020 |
| | MINDi | **0,827** ± 0,024 | 0,825 ± 0,015 | 0,826 ± 0,020 |
| | INDi | **0,825** ± 0,017 | 0,692 ± 0,021 | 0,756 ± 0,019 |
| SyGAR-C10 | DICS | 0,973 ± 0,003 | **0,844** ± 0,017 | **0,906** ± 0,010 |
| | MINDi | 0,973 ± 0,006 | 0,640 ± 0,022 | 0,789 ± 0,016 |
| | INDi | **0,987** ± 0,002 | 0,590 ± 0,020 | 0,763 ± 0,013 |
| SyGAR-C50 | DICS | **0,951** ± 0,011 | **0,688** ± 0,004 | **0,809** ± 0,006 |
| | MINDi | 0,925 ± 0,004 | 0,527 ± 0,009 | 0,698 ± 0,007 |
| | INDi | 0,902 ± 0,010 | 0,566 ± 0,021 | 0,714 ± 0,015 |

ultrapassou 11% na última carga. A queda mais acentuada do valor da métrica K em relação aos resultados obtidos pelo método INDi está relacionada com a maior redução da métrica PMA em cada carga. Comparando com DICS e MINDi, INDi não possui a capacidade de realizar a fusão de *clusters* que representam áreas distintas de atuação dos novos autores, portanto tende a fragmentar os conjuntos de citações destes autores. Em termos de pureza (PMC) houve empate estatístico entre o método proposto e os *baselines*.

Nos cenários simulando mudanças nos perfil de publicação dos autores, o método proposto obteve ganhos aproximados de 14,8% e 19,8%, em relação à métrica K, comparado com o método MINDi nas coleções SyGAR-C10 e SyGAR-C50 respectivamente. Os ganhos em relação ao método INDi foram similares, sendo que este teve um desempenho um pouco melhor comparado com o método MINDi quando 50% dos autores alteraram seu perfil de publicação. Esta diferença pode ser explicada pela estratégia adotada pelo MINDi para realizar a seleção de *clusters* para fusão, na qual é utilizada uma determinada porcentagem das referências no calculo das similaridades. Devido à alteração na distribuição dos tópicos do autor, as citações selecionadas podem não

representar corretamente a sua nova área de interesse. Como os valores de similaridade ficam muito baixos, não há fusões e a fragmentação dos *clusters* tende a aumentar. Nestes cenários, o método DICS obteve os melhores valores de PMA nas duas coleções e de PMC na coleção SyGAR-C50.

### 4.6.3 Avaliação das Capacidades do Algoritmo

Para analisar o impacto de cada componente do algoritmo proposto na qualidade da desambiguação incremental realizada em cada coleção, foram executados experimentos após a remoção de algumas capacidades. Foram avaliadas três variações do método DICS:

- DICS-FR: DICS sem a capacidade de realizar fusões e reclassificar citações duvidosas;
- DICS-R: DICS sem a capacidade de reclassificar citações duvidosas;
- DICS-F: DICS sem a capacidade de realizar fusões de *clusters*.

A Tabela 4.19 mostra os resultados obtidos por cada variação comparando com a versão completa. Quando o método é utilizado sem a capacidade de realizar fusões de *clusters* e de reclassificar citações duvidosas (DICS-FR), os valores da métrica K diminuem cerca de 0,3% na KISTI, 3,6% na BDBComp e, em média, por volta de 7% nas coleções sintéticas. Quando o método foi utilizado sem a capacidade de reclassificar citações duvidosas (DICS-R) houve diferenças estatísticas apenas no cenário simulando a inclusão de novos autores, nestes casos a efetividade do método reduziu até 1,4%. Por último, quando o método foi executado sem a capacidade de realizar fusões (DICS-F), o valor de K diminuiu por volta de 1,4% nas coleções simulando mudanças nos perfis de publicação, e cerca de 1% na coleção SyGAR-N5.

Estes resultados mostram que a capacidade de reclassificar citações duvidosas ajuda a identificar (a partir de alterações nas classificações) citações que pertencem a novos autores, enquanto a capacidade de realizar fusões possibilita melhorar a coesão dos conjuntos de citações dos autores que atuam em diferentes áreas de pesquisa. Entretanto, como os ganhos foram pequenos, uma das duas capacidades poderia ser removida com o objetivo de reduzir o tempo de execução do método sem prejudicar consideravelmente a sua eficácia.

A fim de avaliar melhor o desempenho de cada componente do método proposto nas coleções utilizadas, foram registradas algumas estatísticas durante a execução do DICS. A Tabela 4.20 mostra os números médios obtidos em cada coleção. A partir

Tabela 4.19: Valores da métrica K obtidos utilizando variações do método DICS.

| **Coleção** | **Método** | | | |
|---|---|---|---|---|
| | **DICS-FR** | **DICS-R** | **DICS-F** | **DICS** |
| KISTI | $0{,}935 \pm 0{,}000$ | $\mathbf{0{,}938} \pm 0{,}000$ | $\mathbf{0{,}938} \pm 0{,}001$ | $\mathbf{0{,}938} \pm 0{,}000$ |
| BDBComp | $0{,}900 \pm 0{,}006$ | $\mathbf{0{,}934} \pm 0{,}000$ | $\mathbf{0{,}925} \pm 0{,}011$ | $\mathbf{0{,}934} \pm 0{,}000$ |
| SyGAR-N5 | $0{,}851 \pm 0{,}011$ | $0{,}908 \pm 0{,}005$ | $0{,}906 \pm 0{,}005$ | $\mathbf{0{,}916} \pm 0{,}006$ |
| SyGAR-N10 | $0{,}796 \pm 0{,}013$ | $0{,}851 \pm 0{,}016$ | $\mathbf{0{,}862} \pm 0{,}014$ | $\mathbf{0{,}863} \pm 0{,}020$ |
| SyGAR-C10 | $0{,}844 \pm 0{,}008$ | $\mathbf{0{,}900} \pm 0{,}011$ | $0{,}895 \pm 0{,}007$ | $\mathbf{0{,}906} \pm 0{,}010$ |
| SyGAR-C50 | $0{,}755 \pm 0{,}008$ | $\mathbf{0{,}805} \pm 0{,}009$ | $0{,}796 \pm 0{,}005$ | $\mathbf{0{,}809} \pm 0{,}006$ |

Tabela 4.20: Estatísticas sobre a desambiguação incremental realizada pelo método DICS em cada coleção.

| **Estatística** | **Coleção** | | | | | |
|---|---|---|---|---|---|---|
| | **KISTI** | **BDB-Comp** | **SyGAR-N5** | **SyGAR-N10** | **SyGAR-C10** | **SyGAR-C50** |
| 1. Nº de *clusters* criados: | 6126,0 | 208,2 | 537,8 | 740,4 | 610,4 | 1199,6 |
| 2. Nº de novos autores: | 6849,0 | 205,0 | 150,2 | 366,0 | 0,0 | 0,0 |
| 3. Nº de autores identificados: | 5606,6 | 192,0 | 107,6 | 231,8 | 0,0 | 0,0 |
| 4. Nº de citações: | 40384,0 | 361 | 8391,4 | 10903,6 | 6454,6 | 6500,0 |
| 5. Nº de reclassificações: | 4851,8 | 61,8 | 183663,8 | 301002,0 | 91618,4 | 58124,4 |
| 6. Nº de mudanças | 16,6 | 0,4 | 301,6 | 535,2 | 247,8 | 473,2 |
| 7. Nº de mudanças corretas: | 14,0 | 0,4 | 196,8 | 289,0 | 186,2 | 350,6 |
| 8. Nº de comparações de *clusters*: | 3639,6 | 47,8 | 66821,8 | 127898,4 | 24265,2 | 25937,0 |
| 9. Nº de fusões: | 13,4 | 3,2 | 92,0 | 134,8 | 63,4 | 58,6 |
| 10. Nº de fusões corretas: | 10,8 | 2,8 | 73,6 | 82,8 | 58,4 | 51,6 |

da análise das primeiras três linhas é possível verificar a precisão e cobertura da estratégia utilizada para identificar novos autores. Nas coleções reais, mais de 91% dos *clusters* criados correspondem corretamente a novos autores. 81% e 93% dos autores foram identificados na KISTI e BDBComp, respectivamente. Nas coleções sintéticas simulando a introdução de novos autores, a taxa de identificação dos autores caiu para aproximadamente 63% sendo que, na média, apenas 25% dos *clusters* criados corretamente correspondem a novos autores. Como SyGAR utiliza uma porcentagem dos tópicos dos coautores para criar um novo autor, é esperado que o poder discriminativo dos termos diminua ao longo das cargas, impactando os valores das similaridades e, consequentemente, aumentando o número de *clusters*. No cenário simulando mudanças nos perfis de publicação, SyGAR produz uma pequena alteração na distribuição dos tópicos dos autor. Isto também afeta a capacidade discriminativa dos termos, uma vez que o número de autores que compartilham um determinado termo aumenta. Nestas coleções, DICS cria um alto número de *clusters* mesmo não havendo novos autores. Entretanto, quando comparado com os *baselines*, foi capaz de manter altos níveis de pureza com a mais baixa fragmentação.

As linhas 5, 6 e 7 da Tabela 4.20 mostram alguns números médios obtidos em relação a capacidade do DICS de reclassificar citações duvidosas. Como pode ser observado, DICS realiza um alto número de reclassificações, especialmente nas coleções sintéticas nas quais o número de citações duvidosas aumenta consideravelmente após cada carga. Entretanto, os números de alterações foram pequenos, representam menos de 1% das reclassificações em todas as coleções, com precisão média de 92% e 67% nas coleções reais e sintéticas respectivamente.

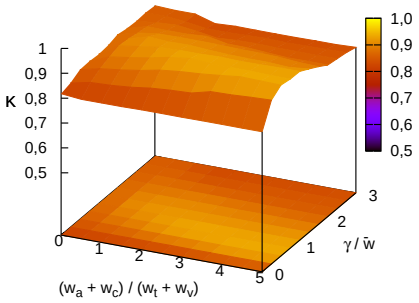
Em relação à capacidade de realizar fusões (três ultimas linhas da Tabela 4.20), é possível observar que o número de fusões é pequeno, mas com precisão média de 84% nas coleções reais, 70% no cenário simulando a introdução de novos autores e 90% no cenário simulando as alterações dos perfis de publicação. O número total de comparações de *clusters* é proporcional ao número de *clusters* e ao número de reclassificações, portanto este número também é afetado pela redução dos valores de similaridade ao longo das cargas nas coleções sintéticas.

De maneira geral, é possível concluir que todos os componentes do método DICS contribuem positivamente para a tarefa de desambiguação, mas a precisão destes componentes pode ser afetada pela ausência de termos discriminativos no conjunto de treinamento. Nos conjuntos de dados utilizados, a capacidade de reclassificar citações duvidosas foi o componente menos importante e de maior custo computacional.

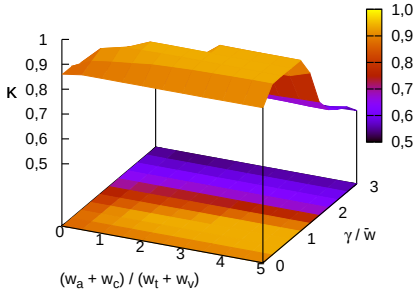
### 4.6.4 Análise de Sensibilidade aos Valores dos Parâmetros

Nesta seção é apresentada uma análise de sensibilidade do método DICS aos valores dos seus parâmetros juntamente com algumas orientações de configuração a fim de se alcançar bons resultados, em termos de pureza e coesão, em um cenário genérico. A análise realizada considera as seguintes observações: (i) normalmente os nomes dos autores são termos mais discriminativos do que os termos do título da publicação e o do nome do local de publicação; e (ii) o melhor valor do parâmetro $\gamma$ é determinado em função dos valores dos pesos dos atributos. Para avaliar o desempenho do método proposto em relação a variação dos valores dos seus parâmetros, considerando as observações acima, foram executados experimentos nas coleções KISTI, BDBComp, SyGAR-N5 e SyGAR-C10; variando os pesos dos atributos $w_a$ e $w_c$ de 0 a 5 com passos de 0,5, enquanto que os pesos $w_t$ e $w_v$ foram mantidos com o valor igual a 1. Para cada configuração dos pesos, o valor de $\gamma$ foi variado de 0 a 3 vezes a média dos pesos com passos de 0,3.

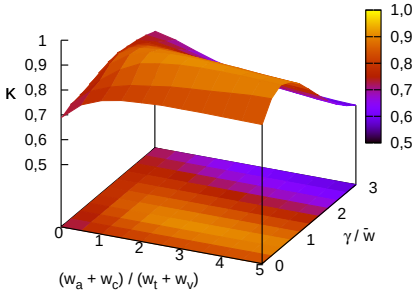
A Figura 4.6 mostra a média dos valores de K obtidos no último ano/carga em cada execução do método DICS utilizando as configurações descritas acima. É possível



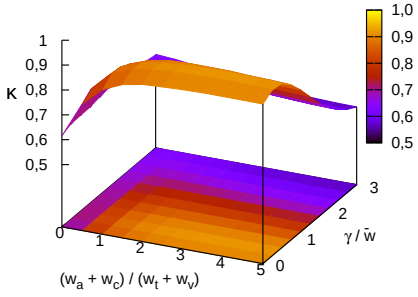

(a) BDBComp




(b) KISTI




(c) SyGAR-N5




(d) SyGAR-C10

Figura 4.6: Análise de sensibilidade so método DICS aos valores dos seus parâmetros.

verificar que, em todas as coleções, os melhores valores de K são obtidos quando os pesos $w_a$ e $w_c$ são de 3 a 5 vezes maiores do que os pesos $w_t$ e $w_v$, e o valor de $\gamma$ está entre 0,6 a 1,2 vezes a média dos pesos dos atributos. Estes resultados indicam que configurações como $w_a = 3, w_c = 3, w_t = 1, w_v = 1$ e $\gamma = 2$, são boas escolhas quando não há conhecimento sobre o conjunto de citações a ser desambiguado.

As variações dos valores de $w_a$ em relação ao peso de $w_c$, e de $w_t$ em relação ao peso de $w_v$, de maneira geral, tendem a impactar pouco os resultados. Entretanto, nos grupos ambíguos que contem a maior parte das referências no formato curto, ou que possuem um nome ambíguo muito comum (como alguns nomes de origem asiática, por exemplo), $w_a$ deve ser menor do que $w_c$. Em relação aos atributos título e local de publicação, se este último conter apenas as siglas dos nomes, como ocorre na coleção KISTI, bons resultados podem ser obtidos ao configurar um valor para o peso $w_v$ maior do que o peso $w_t$. Em relação ao valor de $\gamma$, a média dos pesos deve ser considerada como referência. Para obter *clusters* com altos níveis de pureza, deve ser utilizado um

valor maior do que a média dos pesos dos atributos.

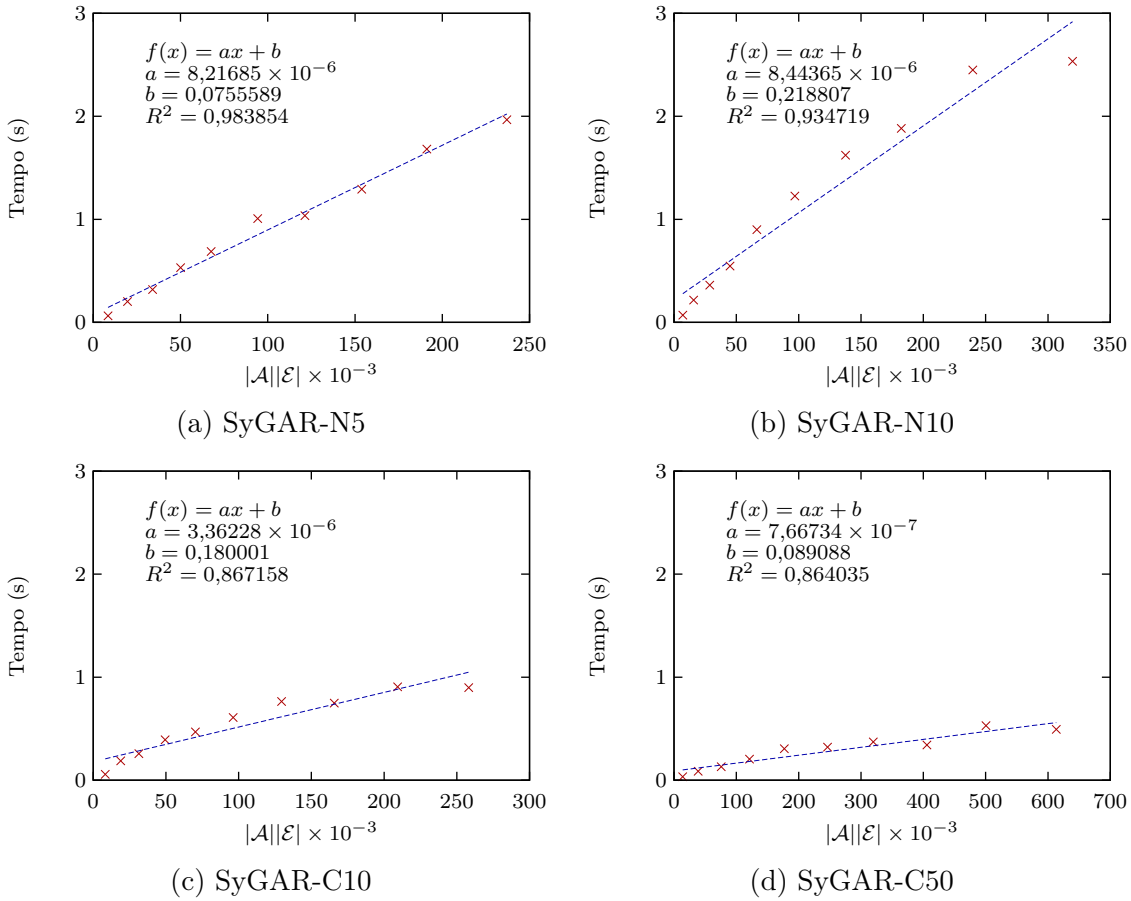
### 4.6.5 Análise do Tempo de Execução

Para mostrar a eficiência do método proposto, foram medidos os tempos de execução do DICS e dos *baselines* MINDi e INDi em cada coleção. Todos os experimentos foram realizados em um computador pessoal com o sistema operacional Linux (Linux Mint 16 Cinnamon 64 bits) com o processador Intel Core i5-3570, CPU de 3.4GHz x 4 e 8GB de memória RAM.

A Tabela 4.21 mostra o tempo médio de execução em segundos gasto por cada método em cada coleção. Como pode ser observado, o método DICS obteve os melhores tempo médios nas coleções sintéticas. Comparado com o método MINDi, DICS foi cerca de 600% mais rápido nas coleções sintéticas, e 189% mais rápido na KISTI. Comparado com o método INDi, DICS foi cerca de 37% mais rápido nas coleções sintéticas e 18% mais lento na KISTI. O maior tempo gasto nas coleções sintéticas simulando a introdução de novos autores se deve ao aumento do número de reclassificações e comparações entre *clusters*.

Para verificar como o aumento do número de *clusters*, $|\mathcal{A}|$, e do número de citações duvidosas, $|\mathcal{E}|$, impactam o tempo de execução, foram registrados os tempos obtidos com a execução do método DICS no maior grupo ambíguo presente nas coleções sintéticas (representado pelos autores que possuem o nome curto igual a "C. Chen"). Em cada carga foram registrados os valores de $|\mathcal{A}|$ e $|\mathcal{E}|$ para comparação com o tempo de execução.

A Figura 4.7 mostra o tempo de execução registrado em função do produto $|\mathcal{A}|\cdot|\mathcal{E}|$, juntamente com o resultado de uma regressão linear sobre os pontos obtidos. É possível verificar que os tempos de execução podem ser representados por funções lineares (o coeficiente de determinação, $R^2$, foi maior do que 0,85 em todas as coleções). Apesar do aumento dos conjuntos $\mathcal{A}$ e $\mathcal{E}$ ser maior nos cenários simulando mudanças nos perfis de publicação, nestas coleções poucas associações são consideradas confiáveis e, portanto, menos citações são reclassificadas. Estes resultados estão em conformidade com a análise de complexidade apresentada na Seção 3.5.




(a) SyGAR-N5 (b) SyGAR-N10

(c) SyGAR-C10 (d) SyGAR-C50

Figura 4.7: Média do tempo de execução do método DICS por número de *clusters* vezes o número de citações duvidosas no grupo ambíguo “C. Chen".

Tabela 4.21: Tempo de execução de cada método com intervalo de confiança de 95%.

| Coleção | Tempo de execução em segundos | | |
|---|---|---|---|
| | DICS | MINDi | INDi |
| KISTI | 5,812 ± 2,697 | 16,825 ± 0,499 | **4,753** ± 0,265 |
| BDBComp | 0,217 ± 0,134 | 0,122 ± 0,062 | **0,065** ± 0,046 |
| SyGAR-N5 | **10,865** ± 0,964 | 70,021 ± 5,481 | 13,296 ± 0,440 |
| SyGAR-N10 | **16,115** ± 1,928 | 124,969 ± 4,336 | 24,195 ± 2,494 |
| SyGAR-C10 | **6,922** ± 0,883 | 40,067 ± 1,063 | 8,930 ± 0,692 |
| SyGAR-C50 | **5,410** ± 0,775 | 44,177 ± 1,785 | 7,925 ± 0,777 |

# Capítulo 5

# Conclusões e Trabalhos Futuros

Neste trabalho, foi proposto um novo método incremental de desambiguação baseado em heurísticas capaz de criar e atualizar automaticamente um conjunto de treinamento utilizado para determinar os autores de cada citação. Foram propostos procedimentos para realizar a configuração automática dos parâmetros com e sem a presença de dados de treinamento. Foi realizada uma extensa avaliação experimental, utilizando coleções reais e sintéticas, a fim de avaliar o desempenho do método quando este é aplicado de forma supervisionada, não supervisionada e incremental. Em todos os cenários, o método proposto superou todos os *baselines*, com ganhos em quase todos os grupos ambíguos. Também foram apresentadas: a avaliação de uma estratégia de incorporação de características baseadas em coocorrência de palavras, uma análise das capacidades do método, uma análise de sensibilidade aos valores dos parâmetros e uma análise da complexidade de tempo do algoritmo.

Os experimentos realizados demonstraram que o método proposto é prático (possui poucos parâmetros de fácil configuração), eficiente (possui baixa complexidade de tempo) e efetivo (é capaz de gerar *clusters* de alta pureza e baixa fragmentação, e de lidar com o ambiente dinâmico das BDs). Foi verificado que a estratégia de incorporação de *c-features*, quando utilizada de forma supervisionada, é capaz de proporcionar ganhos, mas estes não foram significativos. Também foi verificado que todas as capacidades do algoritmo contribuem para sua efetividade em diferêntes cenários, entretanto a capacidade de reclassificação de citações, ou de fusão de *clusters*, poderia ser removida com o objetivo de se reduzir a complexidade de tempo, sem grandes prejuízos na efetividade do método.

Para trabalhos futuros, pretende-se explorar estratégias de *relevance feedback* a fim de permitir melhorias no modelo de desambiguação a partir da interação com um administrador da BD. Para isto, uma possibilidade é utilizar a métrica de confiança de-

senvolvida como parte do critério de seleção de citações a serem consultadas. Também pretende-se: (i) avaliar estratégias para alteração automática dos valores dos parâmetros, a medida que o método atualiza seu conjunto de treinamento; (ii) avaliar a inclusão de novos atributos quando estes estão disponíveis, como ano da publicação e endereço dos autores; e (iii) avaliar a utilização do método proposto em BDs de outras áreas do conhecimento.

# Referências Bibliográficas


Baeza-Yates, R. & Ribeiro-Neto, B. (2011). *Modern Information Retrieval: The Concepts and Technology behind Search (2nd Edition) (ACM Press Books)*. Addison-Wesley Professional, 2 edição. ISBN 0321416910.

Bhattacharya, I. & Getoor, L. (2006). A Latent Dirichlet Model for Unsupervised Entity Resolution. Em *Proceedings of the 2006 SIAM International Conference on Data Mining*, pp. 47--58.

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA. ISBN 0387310738.

Blei, D. M.; Ng, A. Y. & Jordan, M. I. (2003). Latent Dirichlet Allocation. *Journal of Machine Learning Research*, 3:993--1022. ISSN 1532-4435.

Bordes, A.; Ertekin, S.; Weston, J. & Bottou, L. (2005). Fast Kernel Classifiers with Online and Active Learning. *Journal of Machine Learning Research*, 6:1579--1619. ISSN 1532-4435.

Bornmann, L. & Mutz, R. (2014). Growth rates of modern science: A bibliometric analysis based on the number of publications and cited references. http://arxiv.org/abs/1402.4578. Acesso em: 27 de julho de 2015.

Carvalho, A. P.; Ferreira, A. A.; Laender, A. H. F. & Gonçalves, M. A. (2011). Incremental Unsupervised Name Disambiguation in Cleaned Digital Libraries. *Journal of Information and Data Management*, 2(3):289–304.

Cota, R. G.; Ferreira, A. A.; Nascimento, C.; Gonçalves, M. A. & 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. ISSN 1532-2890.

Deerwester, S.; Dumais, S. T.; Furnas, G. W.; Landauer, T. K. & Harshman, R. (1990). Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391--407.

Dempster, A. P.; Laird, N. M. & Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B*, 39(1):1--38.

Esperidião, L. V. B.; Ferreira, A. A.; Laender, A. H. F.; Gomes, D. M.; Tavares, A. I. & Assis, G. T. (2014). Reducing Fragmentation in Incremental Author Name Disambiguation. *Journal of Information and Data Management*, 5(3). ISSN 2178-7107.

Ester, M.; Kriegel, H.; Sander, J. & Xu, X. (1996). A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. Em *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, pp. 226--231. AAAI Press.

Fan, X.; Wang, J.; Pu, X.; Zhou, L. & Lv, B. (2011). On Graph-Based Name Disambiguation. *Journal of Data and Information Quality*, 2(2):10:1--10:23. ISSN 1936-1955.

Fegley, B. D. & Torvik, V. I. (2013). Has Large-Scale Named-Entity Network Analysis Been Resting on a Flawed Assumption? *PLoS One*, 8(7):e70299.

Ferreira, A. A.; Gonçalves, M. A.; Almeida, J. M.; Laender, A. H. F. & Veloso, A. (2012a). A Tool for Generating Synthetic Authorship Records for Evaluating Author Name Disambiguation Methods. *Information Sciences*, 206:42--62. ISSN 0020-0255.

Ferreira, A. A.; Gonçalves, M. A. & Laender, A. H. (2012b). A Brief Survey of Automatic Methods for Author Name Disambiguation. *SIGMOD Record*, 41(2):15--26. ISSN 0163-5808.

Ferreira, A. A.; Machado, T. M. & Gonçalves, M. A. (2012c). Improving Author Name Disambiguation with User Relevance Feedback. *Journal of Information and Data Management*, 3(3):332–347.

Ferreira, A. A.; Silva, R.; Gonçalves, M. A.; Veloso, A. & Laender, A. H. (2012d). Active Associative Sampling for Author Name Disambiguation. Em *Proceedings of the 12th ACM/IEEE-CS Joint Conference on Digital Libraries*, JCDL '12, pp. 175--184, New York, NY, USA. ACM.

Ferreira, A. A.; Veloso, A.; Gonçalves, M. A. & Laender, A. H. F. (2010). Effective Self-training Author Name Disambiguation in Scholarly Digital Libraries. Em *Proceedings of the 10th Annual Joint Conference on Digital Libraries*, pp. 39--48.

Ferreira, A. A.; Veloso, A.; Gonçalves, M. A. & Laender, A. H. F. (2014). Self-training author name disambiguation for information scarce scenarios. *Journal of the Association for Information Science and Technology*, 65(6):1257--1278. ISSN 2330-1643.

Figueiredo, F.; Rocha, L.; Couto, T.; Salles, T.; Gonçalves, M. A. & Meira Jr., W. (2011). Word Co-occurrence Features for Text Classification. *Information Systems*, 36(5):843--858. ISSN 0306-4379.

Frey, B. J. & Dueck, D. (2007). Clustering by passing messages between data points. *Science*, 315(5814):972--976.

Godoi, T. A.; Torres, R. da S.; Carvalho, A. M.; Gonçalves, M. A.; Ferreira, A. A.; Fan, W. & Fox, E. A. (2013). A Relevance Feedback Approach for the Author Name Disambiguation Problem. Em *Proceedings of the 13th ACM/IEEE-CS Joint Conference on Digital Libraries*, JCDL '13, pp. 209--218, New York, NY, USA. ACM.

Gonçalves, M. A.; Fox, E. A.; Watson, L. T. & 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. ISSN 1046-8188.

Gravano, L.; Ipeirotis, P. G.; Koudas, N. & Srivastava, D. (2003). Text Joins in an RDBMS for Web Data Integration. Em *Proceedings of the 12th International Conference on World Wide Web*, WWW '03, pp. 90--101, New York, NY, USA. ACM.

Han, H.; Giles, L.; Zha, H.; Li, C. & Tsioutsiouliklis, K. (2004). Two Supervised Learning Approaches for Name Disambiguation in Author Citations. Em *Proceedings of the 4th ACM/IEEE-CS Joint Conference on Digital Libraries*, JCDL '04, pp. 296--305, New York, NY, USA. ACM.

Han, H.; Xu, W.; Zha, H. & Giles, C. L. (2005). A hierarchical naive Bayes mixture model for name disambiguation in author citations. Em *Proceedings of the ACM Symposium on Applied Computing*, pp. 1065--1069.

Holm, S. (1979). A Simple Sequentially Rejective Multiple Test Procedure. *Scandinavian Journal of Statistics*, 6(2):65--70. ISSN 03036898.

Huang, J.; Ertekin, S. & Giles, C. L. (2006). Efficient Name Disambiguation for Large-scale Databases. Em *Proceedings of the 10th European Conference on Principle and Practice of Knowledge Discovery in Databases*, PKDD'06, pp. 536--544, Berlin, Heidelberg. Springer-Verlag.

Kalashnikov, D. V. & Mehrotra, S. (2006). Domain-independent data cleaning via analysis of entity-relationship graph. *ACM Transactions on Database Systems (ACM TODS)*, 31(2):716--767.

Kang, I.-S.; Kim, P.; Lee, S.; Jung, H. & You, B.-J. (2011). Construction of a large-scale test set for author disambiguation. *Information Processing & Management*, 47(3):452--465. ISSN 0306-4573.

Kass, R. E. & Wasserman, L. (1995). A Reference Bayesian Test for Nested Hypotheses and its Relationship to the Schwarz Criterion. *Journal of the American Statistical Association*, 90(431):928--934. ISSN 01621459.

Kindermann, R. & Snell, J. L. (1980). *Markov Random Fields and Their Applications*. American Mathematical Society. ISBN 9780821850015.

Klaas, V. C. (2007). Who's who in the World Wide Web: Approaches to name disambiguation. Dissertação de mestrado, LMU München, Informatik, Diplomarbeit.

Kurakawa, K.; Takeda, H.; Takaku, M.; Aizawa, A.; Shiozaki, R.; Morimoto, S. & Uchijima, H. (2014). Researcher Name Resolver: identifier management system for Japanese researchers. *International Journal on Digital Libraries*, 14(1-2):39–58. ISSN 1432-5012.

Laender, A. H.; Gonçalves, M. A.; Cota, R. G.; Ferreira, A. A.; Santos, R. L. T. & Silva, A. J. (2008). Keeping a digital library clean: New solutions to old problems. Em *Proceedings of the Eighth ACM Symposium on Document Engineering*, DocEng '08, pp. 257--262, New York, NY, USA. ACM.

Lee, D.; On, B.-W.; Kang, J. & Park, S. (2005). Effective and Scalable Solutions for Mixed and Split Citation Problems in Digital Libraries. Em *Proceedings of the 2nd international workshop on Information Quality in Information Systems*, pp. 69--76.

Li, Y.; Wen, A.; Lin, Q.; Li, R. & Lu, Z. (2014). Name disambiguation in scientific cooperation network by exploiting user feedback. *Artificial Intelligence Review*, 41(4):563–578. ISSN 0269-2821.

Liu, W.; Islamaj Doğan, R.; Kim, S.; Comeau, D. C.; Kim, W.; Yeganova, L.; Lu, Z. & Wilbur, W. J. (2014). Author name disambiguation for PubMed. *Journal of the Association for Information Science and Technology*, 65(4):765--781. ISSN 2330-1643.

Liu, Y.; Li, W.; Huang, Z. & Fang, Q. (2015). A fast method based on multiple clustering for name disambiguation in bibliographic citations. *Journal of the Association for Information Science and Technology*, 66(3):634--644. ISSN 2330-1643.

Mitchell, T. M. (1997). *Machine Learning*. McGraw-Hill, Inc., New York, NY, USA, 1 edição. ISBN 0070428077, 9780070428072.

Oliveira, J. W. A. (2005). Uma estratégia para remoção de ambiguidades na identificação de autoria de objetos bibliográficos. Dissertação de mestrado, Universidade Federal de Minas Gerais, Belo Horizonte, Brazil.

On, B.-W.; Lee, D.; Kang, J. & Mitra, P. (2005). Comparative Study of Name Disambiguation Problem Using a Scalable Blocking-based Framework. Em *Proceedings of the 5th ACM/IEEE-CS Joint Conference on Digital Libraries*, JCDL '05, pp. 344--353, New York, NY, USA. ACM.

Papa, J. P.; Falcão, A. X.; de Albuquerque, V. H. C. & Tavares, J. M. R. (2012). Efficient supervised optimum-path forest classification for large datasets. *Pattern Recognition*, 45(1):512 – 520. ISSN 0031-3203.

Porter, M. F. (1997). Readings in information retrieval. pp. 313--316. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

Santana, A. F.; Gonçalves, M. A.; Laender, A. & Ferreira, A. A. (2015). On the combination of domain-specific heuristics for author name disambiguation: the nearest cluster method. *International Journal on Digital Libraries*, pp. 1–18. ISSN 1432-5012.

Santana, A. F.; Goncalves, M. A.; Laender, A. H. F. & Ferreira, A. A. (2014). Combining domain-specific heuristics for author name disambiguation. Em *Digital Libraries (JCDL), 2014 IEEE/ACM Joint Conference on*, pp. 173–182.

Scoville, C. L.; Johnson, E. D. & McConnell, A. L. (2003). When A Rose is not A. Rose: the vagaries of author searching. *Medical Reference Services Quaterly*, 22(4):1--11.

Shafer, G. (1976). *A mathematical theory of evidence*. Princeton University Press.

Shin, D.; Kim, T.; Choi, J. & Kim, J. (2014). Author name disambiguation using a graph model with node splitting and merging based on bibliographic information. *Scientometrics*, 100(1):15–50. ISSN 0138-9130.

Smalheiser, N. R. & Torvik, V. I. (2009). Author name disambiguation. *Annual Review of Information Science and Technology*, 43(1):1--43. ISSN 1550-8382.

Strotmann, A. & Zhao, D. (2012). Author name disambiguation: What difference does it make in author-based citation analysis? *Journal of the American Society for Information Science and Technology*, 63(9):1820--1833. ISSN 1532-2890.

Strotmann, A.; Zhao, D. & Bubela, T. (2009). Author name disambiguation for collaboration network analysis and visualization. *Proceedings of the American Society for Information Science and Technology*, 46(1):1--20. ISSN 1550-8390.

Tan, C.-M.; Wang, Y.-F. & Lee, C.-D. (2002). The use of bigrams to enhance text categorization. *Information Processing & Management*, 38(4):529 – 546. ISSN 0306-4573.

Tang, J.; Fong, A. C. M.; Wang, B. & Zhang, J. (2012). A Unified Probabilistic Framework for Name Disambiguation in Digital Library. *IEEE Transactions on Knowledge and Data Engineering*, 24(6):975--987. ISSN 1041-4347.

Torres, R. S.; Falcão, A. X.; Gonçalves, M. A.; Papa, J. P.; Zhang, B.; Fan, W. & Fox, E. A. (2009). A genetic programming framework for content-based image retrieval. *Pattern Recognition*, 42(2):283 – 292. ISSN 0031-3203.

Veloso, A.; Ferreira, A. A.; Gonçalves, M. A.; Laender, A. H. F. & Meira, Jr., W. (2012). Cost-effective On-demand Associative Author Name Disambiguation. *Information Processing & Management*, 48(4):680--697. ISSN 0306-4573.

Wang, X.; Tang, J.; Cheng, H. & Yu, P. S. (2011). Adana: Active name disambiguation. Em *Proceedings of the 2011 IEEE 11th International Conference on Data Mining*, ICDM '11, pp. 794--803, Washington, DC, USA. IEEE Computer Society.

Wu, H.; Li, B.; Pei, Y. & He, J. (2014). Unsupervised author disambiguation using Dempster–Shafer theory. *Scientometrics*, 101(3):1955–1972. ISSN 0138-9130.

Zadrozny, B. & Elkan, C. (2002). Transforming Classifier Scores into Accurate Multiclass Probability Estimates. Em *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '02, pp. 694--699, New York, NY, USA. ACM.

Zaïane, O. R. & Antonie, M.-L. (2002). Classifying Text Documents by Associating Terms with Text Categories. *Australian Computer Science Communications*, 24(2):215--222.

Zhang, T. (2004). Solving Large Scale Linear Prediction Problems Using Stochastic Gradient Descent Algorithms. Em *ICML 2004: Proceedings of the Twenty-first International Conference on Machine Learning*, pp. 919--926.