UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO

RAQUEL KOLITSKI STASIU

Avaliação da Qualidade de Funções de Similaridade no Contexto de Consultas por Abrangência

Tese apresentada como requisito parcial
para a obtenção do grau de
Doutor em Ciência da Computação

Prof. Dr. Carlos Alberto Heuser
Orientador

Porto Alegre, julho de 2007

CIP – CATALOGAÇÃO NA PUBLICAÇÃO

Stasiu, Raquel Kolitski

Avaliação da Qualidade de Funções de Similaridade no Contexto de Consultas por Abrangência / Raquel Kolitski Stasiu. – Porto Alegre: PPGC da UFRGS, 2007.

115 f.: il.

Tese (doutorado) – Universidade Federal do Rio Grande do Sul. Programa de Pós-Graduação em Computação, Porto Alegre, BR–RS, 2007. Orientador: Carlos Alberto Heuser.

1. Consultas por abrangência. 2. Funções de similaridade. 3. Avaliação da qualidade. 4. Revocação e precisão. I. Heuser, Carlos Alberto. II. Título.

"Aos meus pais, Izidoro Stasiu e Josaphata Kolitski Stasiu
com todo amor e carinho..."

AGRADECIMENTOS

Meus sinceros agradecimentos a todas as pessoas que incentivaram e contribuíram para o desenvolvimento deste trabalho, de diferentes formas. Muitos ajudaram anonimamente, e têm o meu sincero agradecimento, igualmente, mesmo sem citar os nomes. Em especial, tenho alguns agradecimentos.

O meu primeiro agradecimento e homenagem é para os meus pais (*in memoriam*) Josaphata Kolitski Stasiu e Izidoro Stasiu, que, mesmo distantes fisicamente, com certeza estão muito orgulhosos do resultado deste trabalho como fruto da persistência e coragem que me ensinaram desde a mais tenra idade.

Ao meu irmão, Renato, pelo apoio e incentivo, sempre disponível em todos os momentos. À minha cunhada Denise, pelo apoio e principalmente pela substituição na minha ausência como filha. Aos meus familiares, principalmente meus sobrinhos, Renato e João Victor, por compreenderem a minha ausência.

Ao meu querido Mauro, companheiro de todas as horas, agradeço pela ajuda, paciência, amizade, incentivo e amor. Agradeço também pelo apoio incondicional, pelas críticas construtivas, pela ajuda nos experimentos, pelo apoio moral, por acreditar no meu potencial quando eu mesma duvidava, enfim, por estar ao meu lado, nas horas alegres e nas horas tristes, tanto relacionadas ao desenvolvimento deste trabalho como no aspecto pessoal. Com toda certeza foi um alicerce fundamental para a finalização deste trabalho, principalmente durante os últimos anos de desenvolvimento deste trabalho.

Ao meu orientador, Professor Carlos Alberto Heuser, agradeço não só pela orientação, pelas críticas construtivas e pelo exemplo de formação profissional, docente e pesquisador, mas também pelo apoio e compreensão como ser humano.

Aos meus amigos e colegas de trabalho da UFRGS, em especial, Adrovane Kade, Alexander Vinson, Andrei Lima, Carina Dorneles, Daniela Musa, Deise Saccol, Eduardo Kroth, Juliana Bonato, Mariusa Warpechovski, Renata Galante, Renata Zanela, Rodrigo Gasparoni, Sérgio Mergen, Vanessa Braganholo e Viviane Moreira Orengo que sempre foram presentes e motivadores, incansáveis colaboradoras nas mais diversas tarefas como prévias de apresentação dos trabalhos, leituras dos trabalhos, correções, crítica das idéias, liberação das implementações do trabalho de conclusão de curso, *happy hours*/jantares/churrascos, finais de semana de trabalho, entre outros, e por simplesmente estar por perto.

Aos professores e funcionários da UFRGS, que com dedicação e bom senso tornaram aulas, trabalhos e demais atividades realizadas durante esta tese, uma experiência positiva e produtiva. Em especial, agradeço ao Luís Otávio, Ida Rossi, Elisiane e Margareth pelos diversos favores prestados.

Aos professores Roberto da Silva, Viviane Moreira Orengo, agradeço o compar-

tilhamento das idéias, discussões e o trabalho árduo que resultaram em excelentes publicações em conjunto.

Aos professores José Palazzo, Altigran da Silva, Leandro Wives, Cirano Iochpe, Nina Edelweiss e Clésio Saraiva, agradeço as importantes contribuições nas bancas e avaliações intermediárias, bem como, nas discussões e idéias.

À minha irmã de coração, Andreia Malucelli e à minha amiga Eliane Bodanese, pela constante força, pelos puxões de orelha, por mostrar a força que muitas vezes nem eu percebia que tinha, enfim, pelo suporte incondicional.

Aos meus amigos gaúchos de fora da UFRGS, Ester Góes, Edilson Marques, Paula Brofman, Regina Verdin, Roger Paranhos, Valesca Jungblut, agradeço pelos momentos alegres, descontraídos, divertidos, filosóficos, gastronômicos, caminhadas.

Ao Professor Mariano Consens, pela orientação durante o doutorado sanduíche na Universidade de Toronto, pela visão crítica e aberta, pela oportunidade de integração com o grupo e com os demais serviços da Universidade de Toronto, e principalmente, pelo apoio, compreensão e incentivo nos momentos difíceis.

Aos colegas do laboratório da Universidade de Toronto durante o doutorado sanduíche Universidade de Toronto, agradeço pelo acolhimento e troca de experiências culturais (e gastronômicas) dos diversos países de origem: Adrian - Romênia; Flavio - Argentina; Jingwei, Xin, Xing, Wei e John - China; Mariano e Lydia - Uruguai; Sadek - Canada; Yaron - Israel; e eu do Brasil.

À Susanne McGarvey Isreig, pelo carinho de mãe durante o doutorado sanduíche.

À Capes, pelo financiamento do doutorado sanduíche na Universidade de Toronto, no Canadá.

À UTFPR (antigo e mais carinhoso CEFET-PR), pelo suporte institucional. Em especial aos meus colegas e amigos José Marcos Marcassi Rodrigues, Julio Puccini, Luciano Scandelari, Patrícia Strapassom e Solange Chiocarello, por entenderem a necessidade e importância da realização deste trabalho, pela cobertura no desenvolvimento das atividades e principalmente, pela torcida organizada de sempre.

À PUCPR, pelo suporte institucional, e em especial aos colegas e amigos, Andreia Malucelli, Attilio Zanelatto Neto, Celso Kaestner, Edson Scalabrin, Henri Eberspacher, João da Silva Dias, Manoel Camillo Penna, Marcos Shmeil e Robert Carlisle Burnett, pela liberação e incentivo, pelas sugestões e intervenções no momento certo, por apresentar o orientador e a área de banco de dados, e pelo companheirismo e incentivo em diversos momentos, os quais foram fundamentais para permitir o desenvolvimento deste trabalho.

SUMÁRIO

LISTA DE ABREVIATURAS E SIGLAS

SGBDR	Sistema Gerenciador de Banco de Dados Relacional
WHIRL	*Word-based Heterogeneous Information Representation Language*
XML	*eXtensible Markup Language*
PDF	*Probability Density Function*
TF-IDF	*Term-Frequency Inverse Document Frequency*
R&P	Revocação e Precisão
MSD	*Mean Square Deviation*
MPL	Média da Precisão por Limiar
MRL	Média da Revocação por Limiar
UDF	*User Defined Functions*
MAP	*Mean Average Precision*
AP	*Average Precision*

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ALGORITMOS

RESUMO

Em sistemas reais, os dados armazenados tipicamente apresentam inconsistências causadas por erros de grafia, abreviações, caracteres trocados, entre outros. Isto faz com que diferentes representações do mesmo objeto do mundo real sejam registrados como elementos distintos, causando um problema no momento de consultar os dados. Portanto, o problema investigado nesta tese refere-se às consultas por abrangência, que procuram "encontrar objetos que representam o mesmo objeto real consultado". Esse tipo de consulta não pode ser processado por coincidência exata, necessitando de um mecanismo de consulta com suporte à similaridade. Para cada consulta submetida a uma determinada coleção, a função de similaridade produz um *ranking* dos elementos dessa coleção ordenados pelo valor de similaridade entre cada elemento e o objeto consulta. Como somente os elementos que são variações do objeto consulta são relevantes e deveriam ser retornados, é necessário o uso de um limiar para delimitar o resultado.

O primeiro desafio das consultas por abrangência é a definição do limiar. Geralmente é o especialista humano que faz a estimativa manualmente através da identificação de elementos relevantes e irrelevantes para cada consulta e em seguida, utiliza uma medida como revocação e precisão (R&P). A alta dependência do especialista humano dificulta o uso de consultas por abrangência na prática, principalmente em grandes coleções. Por esta razão, o método apresentado nesta tese tem por objetivo estimar R&P para vários limiares com baixa dependência do especialista humano. Como um sub-produto do método, também é possível selecionar o limiar mais adequado para uma função sobre uma determinada coleção.

Considerando que as funções de similaridade são imperfeitas e que apresentam níveis diferentes de qualidade, é necessário avaliar a função de similaridade para cada coleção, pois o resultado é dependente dos dados. Um limiar para uma coleção pode ser totalmente inadequado para outra coleção, embora utilizando a mesma função de similaridade. Como forma de medir a qualidade de funções de similaridade no contexto de consultas por abrangência, esta tese apresenta a discernibilidade. Trata-se de uma medida que define a habilidade da função de similaridade de separar elementos relevantes e irrelevantes. Comparando com a precisão média, a discernibilidade captura variações que não são percebidas pela precisão média, o que mostra que a discernibilidade é mais apropriada para consultas por abrangência.

Uma extensa avaliação experimental usando dados reais mostra a viabilidade tanto do método de estimativas como da medida de discernibilidade para consultas por abrangência.

Palavras-chave: Consultas por abrangência, funções de similaridade, avaliação da qualidade, revocação e precisão.

Quality evaluation of similarity functions for range queries

ABSTRACT

In real systems, stored data typically have inconsistencies caused by typing errors, abbreviations, transposed characters, amongst others. For this reason, different representations of the same real world object are stored as distinct elements, causing problems during query processing. In this sense, this thesis investigates range queries which " find objects that represent the same real world object being queried". This type of query cannot be processed by exact matching, thus requiring the support for querying by similarity. For each query submitted to a given collection, the similarity function produces a ranked list of all elements in this collection. This ranked list is sorted decreasingly by the similarity score value with the query object. Only the variations of the query object should be part of the result as only those items are relevant. For this reason, it is necessary to apply a threshold value to properly split the ranking.

The first challenge of range queries is the definition of a proper threshold. Usually, a human specialist makes the estimation manually through the identification of relevant and irrelevant elements for each query. Then, he/she uses measures such as recall and precision (R&P). The high dependency on the human specialist is the main difficulty related to use of range queries in real situations, specially for large collections. In this sense, the method presented in this thesis has the objective of estimating R&P at several thresholds with low human intervention. As a by-product of this method, it is possible to select the optimal threshold for a similarity function in a given collection.

Considering the fact that the similarity functions are imperfect and vary in quality, it is necessary to evaluate the similarity function for each collection as the result is domain dependent. A threshold value for a collection could be totally inappropriate for another, even though the same similarity function is applied. As a measure of quality of similarity functions for range queries, this thesis introduces discernability. This is a measure to quantify the ability of the similarity function in separating relevant and irrelevant elements. Comparing discernability and mean average precision, the first one can capture variations that are not noticed by precision-based measures. This property shows that discernability presents better results for evaluating similarity functions for range queries.

An extended experimental evaluation using real data shows the viability of both, the estimation method and the discernability measure, applied to range queries.

Keywords: quality evaluation, similarity queries, similarity function, range queries, threshold estimation.

1 INTRODUÇÃO

A disseminação da tecnologia da informação diversificou e ampliou os horizontes dos sistemas de informação. Hoje em dia, utilizando técnicas apropriadas, é possível obter resultados satisfatórios às consultas submetidas, mesmo com pouco ou nenhum conhecimento do conteúdo armazenado. Entretanto, os dados armazenados apresentam variações tanto ortográficas como de formato, principalmente quando recebem dados provenientes de digitação ou de sistemas que foram integrados. Durante a entrada de dados as pessoas tendem a variar a ortografia de dados, usando abreviações, siglas, erros e acentuação em atributos como nomes de pessoas, nomes de conferências, endereços, etc. Também podem ocorrer variações no formato de datas, números, valores monetários, unidades de medida e outros, e que ainda assim tais dados podem referir-se ao mesmo objeto real. Processadores tradicionais de consulta (ULLMAN; GARCIA-MOLINA; WIDOM, 2002) não têm suporte quando as consultas envolvem tais variações por basearem-se no critério de igualdade.

Por exemplo, uma consulta sobre uma coleção de referências bibliográficas, por exemplo, que envolva um predicado com nome de autor "Bill Silver", mesmo contendo variações como "B. Silver" e "Bil Silver" correspondentes ao mesmo nome consultado, retornaria um conjunto vazio como resposta quando utilizado um mecanismo de consulta baseado em igualdade. Dados provenientes de sistemas que integram diferentes fontes bibliográficas podem conter o nome deste mesmo autor de forma invertida, como "Silver, Bill", representando o mesmo autor. Portanto, para recuperar todas as ocorrências do autor citado, o critério de igualdade não se aplica. Outro exemplo sobre a mesma base de referências bibliográficas, pode-se ter alguns artigos cujo nome da conferência tenha sido cadastrado como "VLDB" e em outros artigos apareça como "*Very Large Database*". Ao submeter uma consulta sobre essa base bibliográfica com um predicado envolvendo o nome da conferência, seria adequado que o processador da consulta recuperasse todas as representações desse objeto, independente do formato usado no argumento da consulta, uma vez que os dois valores referem-se à mesma conferência.

Portanto, o problema investigado nesta tese refere-se às consultas que procuram "encontrar objetos que representam o mesmo objeto real". Como esse tipo de consulta não pode ser processado por coincidência exata, necessita de um mecanismo de consulta que suporte similaridade.

As consultas por similaridade utilizam uma função de similaridade que permite produzir um *ranking* dos elementos da coleção para um determinado objeto consulta, incluindo o respectivo grau de similaridade denominado escore. De maneira geral, pode-se dizer que o escopo da consulta é retornar todos os elementos da coleção. Entretanto, é intuitivo perceber que muitos elementos não deveriam ser retornados

por não serem considerados relevantes à consulta, apesar de, mesmo assim ter um certo grau de similaridade. Por esta razão, é importante determinar um limite do escore mínimo aceitável como relevante. No contexto desta tese, relevantes são os elementos que correspondem ao mesmo objeto consulta no mundo real.

Portanto, utiliza-se o termo `consulta por abrangência` para designar a consulta por similaridade com um valor limite, o qual determina quais representações de objetos armazenados podem ser consideradas como sendo o mesmo objeto consulta no mundo real. Esse valor limite é conhecido como limiar (*threshold*), determina o raio de abrangência de uma consulta por similaridade e garante que todos os elementos retornados terão um escore igual ou acima desse limite. O valor do limiar baixa caracteriza as consultas Web e as consultas por similaridade em geral.

As consultas por abrangência têm aplicações em diversas áreas, utilizando diferentes nomenclaturas, tais como: mecanismo de consulta *vague queries* (MOTRO, 1988), consultas imprecisas (NAMBIAR; KAMBHAMPATI, 2003); deduplicação como *record linkage* (WINKLER, 1999; FELLEGI; SUNTER, 1969), *merge/purge problem* (HERNANDEZ; STOLFO, 1995), *duplicate detection* (BILENKO; MOONEY, 2003; SARAWAGI; BHAMIDIPATY, 2002a; MONGE; ELKAN, 1997); ou identificação de entidades como *entity name matching* (COHEN; RICHMAN, 2002), *entity resolution* (BENJELLOUN et al., 2006), *hardening soft databases* (COHEN; KAUTZ; MCALLESTER, 2000), *reference matching* (MCCALLUM; NIGAM; UNGAR, 2000); ou identificação de instâncias *object matching* (DOAN et al., 2003), *object identification* (TEJADA; KNOBLOCK; MINTON, 2001), *instance matching* (RAHM; DO, 2000), e assim por diante. Outra aplicação de consultas por abrangência é sobre os algoritmos de junção por similaridade. Um limiar adequado sobre o resultado de uma função de similaridade permite ao operador de junção identificar quais elementos re referem ao mesmo objeto real, para então fazer a junção propriamente dita.

Os exemplos citados caracterizam as consultas por abrangência. Outro tipo de consulta por similaridade é denominado consulta por quantidade, conhecidos como *top-k queries*. O problema é justamente que o valor de k não é conhecido durante a especificação da consulta. As aplicações citadas nos exemplos acima pressupõem que seja determinado o número de objetos que representam o mesmo elemento usado como objeto consulta somente após a execução da consulta. Por isso, determinar um valor de n ao acaso implica em dizer que os primeiros n elementos são relevantes, o que pode ser totalmente incorreto.

As consultas por abrangência têm outro desafio, que é como determinar o limiar. Para determinar o limiar adequado, é necessário analisar vários *rankings* resultantes da função de similaridade. Então, a partir da identificação dos elementos relevantes e irrelevantes, um especialista humano determina um limiar que possa restringir o escore para que somente os elementos façam parte do resultado. Justamente a dependência de intervenção humana dificulta o uso de consultas por abrangência na prática.

Atribuir um valor de limiar ao acaso, por tentativa e erro acaba trazendo frustração no processo de consulta. Uma estratégia adotada nos sistemas de Recuperação de Informações (RI) é avaliar a qualidade do resultado da consulta em vários limiares. Tradicionalmente, a avaliação da qualidade de funções de similaridade em sistemas RI é medido através de curvas de revocação e precisão (R&P) e outras medidas derivadas (SALTON; MCGILL, 1983; BAEZA-YATES; RIBEIRO-NETO,

1999). O problema com o uso de tais medidas é a alta dependência da intervenção de um especialista humano. Para cada consulta realizada é necessário contar o número de elementos retornados e destes, quantos são relevantes, em cada limiar.

Portanto, minimizar a intervenção humana na avaliação da qualidade de funções de similaridade para estimar valores de R&P em vários limiares é o objetivo do método apresentado nesta tese. A estimativa dos valores de R&P em vários limiares permite avaliar a qualidade de funções de similaridade. Por conseqüência, obtendo um indicativo da qualidade desejada é possível utilizar o limiar correspondente, que é uma saída do método proposto nesta tese.

De maneira geral, pode-se observar a respeito das funções de similaridade que: (i) são imperfeitas por construção, pois dependem dos dados existentes na coleção (COHEN; RAVIKUMAR; FIENBERG, 2003); e (ii) têm qualidade diferente, uma vez que a qualidade depende do domínio do conjunto de dados utilizado.

Referente à primeira observação, durante o estudo, pesquisa e validação através de variados experimentos, foi possível comprovar que um limiar usado em uma coleção pode ser totalmente inadequado para outra. Da mesma forma, em certos casos, somente um especialista humano consegue determinar se duas representações referem-se ao mesmo objeto do mundo real. Isto ocorre porque uma determinada função de similaridade pode atribuir um escore maior para representações de objetos distintos do mundo real comparado às representações do mesmo objeto. Por exemplo, uma função de similaridade pode atribuir um escore maior para "Porto Alegre" e "Pouso Alegre", que são duas cidades diferentes, do que para "Porto Alegre" e "P. Alegre", onde parte no nome da mesma cidade aparece abreviado.

As duas observações indicam que antes de usar uma função de similaridade, é necessário avaliar a qualidade do resultado produzido pela mesma sobre um conjunto de dados da coleção. Entretanto, os experimentos desenvolvidos nesta tese mostraram que a avaliação da qualidade de consultas por abrangência baseada em medidas como R&P não é adequada para capturar certas variações sobre o grau de similaridade, determinado pelo escore. R&P utilizam somente a ordem e a quantidade de elementos relevantes em certas posições do *ranking*.

As medidas baseadas em R&P, por não considerarem o valor do escore, favorecem os elementos relevantes que são encontrados no topo do *ranking*, o que é certamente adequado para consultas por quantidade. Para avaliar a qualidade de funções de similaridade aplicadas no contexto de consultas por abrangência, esta tese apresenta uma nova medida denominada discernibilidade.

O objetivo da discernibilidade é medir a habilidade da função de similaridade de separar os elementos relevantes dos irrelevantes. Comparando com medidas baseadas em R&P, a discernibilidade mostrou-se apropriada para consultas por abrangência, pois consegue perceber variações no *ranking* que medidas tradicionais consideram. Tais variações são decorrentes do uso do escore no cálculo da discernibilidade e na premissa de que uma função tem melhor qualidade quando separa com os elementos relevantes dos irrelevantes uma distância maior. Seguindo este raciocínio significa que uma função ideal deveria atribuir um escore igual a 1 para elementos relevantes e 0 para os irrelevantes.

Conforme exposto, esta tese investiga estratégias de avaliação da qualidade de consultas por abrangência, apresentando novas soluções para contribuir com o estado da arte da literatura assim como os resultados dos experimentos realizados. Portanto, as principais contribuições desta tese podem ser resumidas em três tópicos:

1. um método semi-automático para estimar R&P em vários valores de limiar com baixa intervenção de um especialista humano;

2. uma medida para avaliar a qualidade de funções de similaridade no contexto de consultas por abrangência;

3. experimentos que corroboram o método de estimativa e a medida de qualidade propostos, no processo de definição do limiar mais apropriado.

O trabalho está organizado da seguinte forma. O Capítulo 2 apresenta o estado da arte, referenciando o tipo de funções de similaridade consideradas ao longo deste trabalho, as consultas por similaridade e suas aplicações relacionadas com a consulta por abrangência. Alguns trabalhos relevantes na literatura são rapidamente abordados. O Capítulo 3 descreve o método de estimativa de revocação e precisão para vários limiares. O Capítulo 4 apresenta a nova medida de avaliação da qualidade de funções de similaridade denominada discernibilidade. O Capítulo 5 refere-se à avaliação experimental. Refere-se à forma de validação dos experimentos, assim como à explicação do modo de execução, além da discussão dos resultados obtidos. Os experimentos relatados são referentes ao método apresentado no Capítulo 3 e à medida de discernibilidade introduzida no Capítulo 4. O Capítulo 6 apresenta as conclusões, publicações e trabalhos futuros referentes ao trabalho desenvolvido.

2 TRABALHOS RELACIONADOS

Neste capítulo, são analisadas as principais estratégias encontradas na literatura para a avaliação da qualidade de funções de similaridade aplicadas ao contexto de consultas por abrangência.

A Seção 2.1 inicia discutindo alguns conceitos importantes a respeito de métricas, conceitos e tipos de funções de similaridade, bem como, a respeito do significado de escore, relevância e limiar. Tais conceitos são empregados no desenvolvimento desta tese. A Seção 2.2 classifica de forma didática os tipos de consulta por similaridade de acordo com o problema adotado e com o tipo de delimitador. Vale salientar que essa classificação não é formal, mas apenas uma forma de agrupamento dos trabalhos relacionados. Em seguida, a Seção 2.3 apresenta trabalhos da literatura que abordam diferentes cenários onde as consultas por abrangência são aplicadas e a avaliação da qualidade abordada nesta pode ser aplicada. A Seção 2.4 apresenta o estado da arte em termos de avaliação da qualidade de funções de similaridade. Estão incluídos os fundamentos e exemplos do uso de precisão e revocação como medidas de avaliação da qualidade de funções de similaridade. E finalmente, embora não esteja necessariamente se referindo aos trabalhos relacionados, a Seção 2.5 apresenta, de forma resumida, os principais conceitos sobre agrupamento por similaridade, para facilitar o entendimento do método apresentado no Capitulo 3.

2.1 Funções de similaridade

Neste trabalho utiliza-se o termo de função de similaridade de forma genérica, referindo-se também às métricas de similaridade. De forma simplificada, pode-se dizer que uma função de similaridade calcula o grau de semelhança entre dois objetos. Primeiramente, é apresentada a diferença entre métrica e função de similaridade. Em seguida, são comentados alguns aspectos relativos ao tipo de funções utilizadas, ao significado do escore, bem como algumas considerações a respeito de limiar e relevância.

2.1.1 Métricas vs. funções de similaridade

As funções de similaridade, assim como as funções de distância, têm por finalidade atribuir uma medida do grau de semelhança entre dois objetos. De forma simplificada, medidas de similaridade podem ser definidas como: sejam x e y objetos de um determinado universo U, a medida de similaridade é a função $sim(x, y) \rightarrow [0, 1]$. Alternativamente a medida de distância $d(x, y) \rightarrow [0, 1]$ pode ser usada. Todas as distâncias podem ser transformadas em medidas de similaridade, usando uma

transformação simples como $sim(x, y) = 1 - d(x, y)$.

Uma medida amplamente usada em um espaço n-dimensional é a Distância Euclidiana, que pode ser representada por uma distância d entre os pontos de dados s_1, s_2, e s_3 em um espaço métrico S. Para receber a denominação de `métrica` (BIMBO, 1999), uma função de distância precisa atender as seguintes condições:

- identidade: $d(s_1,s_2) = d(s_1,s_2) = 0$;
- minimalidade : $d(s_1,s_2) \geq\ d(s_1,s_2) \geq\ 0$;
- simetria : $d(s_1,s_2) = d(s_2,s_1)$;
- desigualdade triangular : $d(s_1,s_2) + d(s_2,s_3) \geq d(s_1,\ s_3)$.

Para ser considerada uma métrica, a função de similaridade deve atender às propriedades citadas acima. Neste trabalho, optou-se por manter o termo função de similaridade como se não fossem coisas diferentes.

Um predicado de similaridade como $sim(x, y) \mapsto [0, 1]$, significando "x é semelhante a y com um certo grau dentro do intervalo [0,1]", pode ser derivado de medidas de similaridade ou distância.

Uma métrica de distância bem conhecida para representações de caracteres do tipo texto é Levenshtein (LEVENSHTEIN, 1966) ou *edit distance* $ed(p, w)$. Certos custos são associados com operações como inserção, exclusão ou substituição para transformar uma seqüência de caracteres original p em uma comparação w, e a distância mínima é computada. Por exemplo, assumindo um valor constante 1 para as operações citadas, a distância entre "flowers" e "flowoes" é 2, porque o menor conjunto de operações aplicáveis é $substituir(\#5, \text{“}e''), substituir(\#6, \text{“}r'')$, ou $excluir(\#5), inserir(\#6, \text{“}r'')$. Existem diversas variações e outras métricas específicas descritas por Navarro (NAVARRO, 2001).

De maneira geral, uma função de similaridade implementa um algoritmo para definir um escore, que quantifica a semelhança entre dois ou mais objetos. Embora seja dependente da implementação da função de similaridade, neste trabalho optou-se por manter o valor do escore dentro do intervalo [0,1] como forma de uniformizar os valores. Um escore igual a 0 significa que os dois objetos analisados são totalmente diferentes, e um escore igual a 1 indica que são iguais.

Uma função de similaridade pode ser implementada em um mecanismo de consulta sobre uma determinada coleção. Quando uma determinada consulta é submetida, um objeto consulta é comparado com todos os elementos da coleção, e para cada um dos elementos da coleção, a função de similaridade determina um escore. O escore representa o grau de similaridade entre os elementos da coleção e o objeto consulta. O resultado é apresentado em forma de uma lista em ordem decrescente do escore, denominada *ranking*.

Portanto, o escore determina o valor de similaridade entre os objetos analisados, diferente do conceito de relevância. Então, cabe ao usuário identificar se o elemento é relevante ou não para a consulta realizada. Entretanto, existe uma relação entre escore e relevância. Para um elemento a com escore x e um elemento b com escore y, e se $x > y$ é possível afirmar que a é mais relevante que b.

2.1.2 Tipos de funções de similaridade usadas

No contexto deste trabalho foram usadas tanto funções de similaridade que envolvem aspectos genéricos do domínio como aspectos específicos. Por aspectos genéricos (HALL; DOWLING, 1980; NAVARRO, 2001; COHEN; RAVIKUMAR; FIENBERG, 2003), entende-se que função de similaridade considera as propriedades relacionadas ao tipo de dado do domínio do atributo, como por exemplo, se é do tipo caracter, ou possui texto curto ou texto longo, ou se é to tipo numérico, etc. Já as funções de similaridade que consideram aspectos específicos do domínio atributo estão relacionadas com a semântica do conteúdo significativa para o usuário. Compreendem funções definidas pelo usuário, as quais podem usar características como formato e apresentação dos valores como parte da definição de objetos similares.

Exemplos desses aspectos específicos podem ser encontrados em atributos como nome de pessoas (pode ter abreviações, inversões de nome/sobrenome, etc.), siglas/acrônimos (formação através das primeiras letras, abreviações, etc.), email (apresenta um caracter "@" e pelo menos um "."), endereços (possuem abreviações de logradouros), data entre outros(LIMA, 2002; DORNELES et al., 2004; GUTH, 1976).

As funções de similaridade genéricas para dados do tipo texto podem ser classificadas em dois tipos (COHEN; RAVIKUMAR; FIENBERG, 2003): (i) baseadas em edição de distância, caracter a caracter, e (ii) baseadas em token (ou freqüência de termo). Como exemplo do primeiro grupo pode ser citado *Edit distance* (LEVENSHTEIN, 1966; HALL; DOWLING, 1980), N-gram (NAVARRO, 2001; GRAVANO et al., 2001), *Jaccard* (JACCARD, 1912), *Jaro* (JARO, 1989), *JaroWinkler* (WINKLER, 1999), entre outros. Como exemplo do segundo grupo, tem-se TF-IDF (*Term Frequency-Inverse Document Frequency*) (SALTON; MCGILL, 1983) como uma função bem conhecida na área de RI.

O critério de escolha para as funções de similaridade usadas foi utilizar tanto funções de similaridade genéricas já conhecidas e utilizadas em ferramentas e implementações como o Projeto *SecondString*, assim como funções de similaridade mais específicas, desenvolvidas pelo usuário com conhecimento sobre o contexto semântico dos dados armazenados. As implementações específicas foram utilizadas certas funções de similaridade desenvolvidas no próprio grupo de pesquisa ao qual este trabalho está vinculado (LIMA, 2002; DORNELES et al., 2004). Uma breve descrição das funções utilizadas nos experimentos é descrita na Seção 5.1, do Capítulo 5.

2.1.3 Significado do escore, limiar e relevância

Embora o cálculo do valor do escore seja peculiar às implementações das funções de similaridade, e conseqüentemente o valor numérico obtido possa ser variado de função para função, neste trabalho utiliza-se o valor normalizado dentro do intervalo [0,1]. O uso dos valores de escore normalizados tem por objetivo facilitar a comparação entre os valores de diferentes funções de similaridade. Portanto, o escore igual a 1 dentro do intervalo [0,1] equivale a dizer que o elemento armazenado e o consultado são iguais.

O uso de diferentes funções de similaridade para os diferentes atributos que compõe o objeto resulta em diferentes escores, o que implica na necessidade de um mecanismo de combinação desses escores em um valor único. A combinação de escores é o objetivo de estudo apresentado por Dorneles(DORNELES, 2006), que mostra

que certos atributos podem ter pesos diferentes, influenciando significativamente no escore final. O referido trabalho investiga meios alternativos para combinar cada escore obtido a partir das diferentes funções de similaridade utilizadas em cada atributo e produzir um "escore combinado". O resultado final que determina o *ranking* da consulta utiliza esse valor combinado como critério de ordenação. Existem diversas técnicas e estratégias de combinação de escores como pode ser observado na literatura (TEJADA; KNOBLOCK; MINTON, 2001; MOTRO, 1988). Geralmente tais combinações envolvem pesos diferenciados para os atributos, apresentando resultados satisfatórios na identificação de instâncias duplicadas.

Utilizando atributos combinados ou não, é fato que cada função de similaridade produz um escore entre um elemento consultado e cada um dos elementos da coleção, e que esse escore varia conforme os dados da coleção. Por razões de simplicidade, bem como, pelo fato desta tese não abordar a combinação de escores de cada atributo, optou-se por utilizar coleções contendo apenas elementos compostos por um atributo nos experimentos. Portanto, cada elemento é atômico, e pode ter seu respectivo escore calculado por diferentes funções em um contexto de consulta por similaridade.

Uma função de similaridade é mais apropriada que outra quando atribui valores de escore maiores para os elementos mais relevantes, de forma que os mesmos se encontrem nas posições iniciais do *ranking*. Embora o termo relevante possa ser encontrado como sinônimo de similar na literatura, no contexto deste trabalho, **relevância** refere-se ao grau de concordância entre o objeto consultado e cada elemento da coleção no sentido de que ambos representam o mesmo objeto real.

Entretanto, o valor expresso pelo **escore** representa o grau de similaridade entre o objeto consulta e cada elemento da coleção. É o valor usado como **limiar** que determina separação dos elementos do *ranking* que são relevantes dos irrelevantes. Por esta razão, o valor usado com o **limiar** tem um papel fundamental na qualidade do resultado produzido por uma função de similaridade. Uma vez separados os elementos do *ranking* em grupos de relevantes e irrelevantes, o valor do escore em si não é um valor significativo, pois é dependente da implementação da função de similaridade. Funções diferentes atribuem diferentes valores de escore para os mesmos pares de objeto consulta e elementos, de acordo com suas características internas.

Funções de similaridade que consideram particularidades, como os aspectos sintáticos e semânticos do domínio dos dados, permitem maior autenticidade e consistência ao *ranking* produzido, identificando os elementos mais relevantes no topo do *ranking* por atribuir um escore de valor mais alto. Pode-se intuir que uma função perfeita atribuiria escore máximo para todos os elementos relevantes. A questão chave se concentra em torno do valor apropriado para o limiar, pois é o fator principal para atingir o objetivo da consulta por similaridade: minimizar falsos positivos e falsos negativos.

Considera-se falso positivo um elemento retornado como parte do resultado de uma consulta processada por uma função de similaridade, que não representa uma variação do objeto consultado. Por analogia, falso negativo, ocorre quando a função de similaridade deixa de incluir no resultado certos elementos que deveriam ser considerados relevantes, geralmente devido ao uso de um limiar inadequado ou devido ao baixo escore de similaridade produzido pela função. Em aplicações como os sistemas de RI(BAEZA-YATES; RIBEIRO-NETO, 1999), os termos falso positivo referem-se ao retorno de um documento que não é relevante à consulta, e falso negativo,

refere-se à ausência de certo documento no resultado, mesmo ausente na coleção.

O resultado produzido por uma função de similaridade geralmente consiste em ordenar pelo valor do escore os elementos da coleção, resultando em um volume razoável de respostas que não interessa ao usuário. Para determinar que somente elementos relevantes sejam retornados, ou que pelo menos somente os `mais relevantes` sejam recuperados, é necessário o uso de um limiar. Objetivo de um limiar é definir um critério de corte, determinando um subconjunto dos elementos pertencentes ao *ranking* produzido pela função de similaridade que é apresentado como resposta.

2.2 Consultas por similaridade

Este trabalho refere-se ao uso de consultas por abrangência, que é um tipo de consulta por similaridade. Uma consulta por abrangência é submetida sobre uma coleção e processada por um mecanismo de consulta que implementa uma função de similaridade retorna um *ranking* com os elementos relevantes para a consulta. Uma coleção de dados refere-se a um conjunto de instâncias com duplicidades ou não, porém que representem entidades atômicas, ou seja, cada instância refere-se a um objeto do mundo real. As diversas representações podem ser decorrentes de erros de grafia, abreviações, inversões na ordem das palavras, formato diferente, entre outros. Por exemplo, em uma coleção de instituições de ensino pode-se encontrar "UFRGS" ou "Univ. Federal do Rio Grande do Sul" ou "Universidade F. do RGS" ou ou "Univesidade do Rio Grnde do Sul" além de diversas outras variações, porém todas representando a mesma instituição. Em uma consulta por abrangência, considera-se que todas as representações do exemplo acima como elementos relevantes e que correspondem a mesma instituição de ensino.

Considerando consultas por similaridade, é possível utilizar uma função de similaridade específica para cada atributo (COHEN; RAVIKUMAR; FIENBERG, 2003), que melhor se adapte ao domínio e considere aspectos sintáticos e/ou semânticos dos dados armazenados. Por exemplo, uma coluna de um banco de dados de `referências bibliográficas` contendo `nome de autores` e outra `datas de publicação`, requerem diferentes funções de similaridade, que poderiam estar associadas ao esquema do banco de dados. Um processador de consulta por similaridade executa a implementação associada para cada coluna ao avaliar a consulta para considerar as variações do objeto como representações do mesmo objeto do mundo real. Da mesma forma, é possível usar uma função de similaridade que considere aspectos semânticos característicos do atributo, como o uso de abreviação, siglas ou acrônimos, formatos de data, padrões de caracteres (como email, CEP, etc.), inversão de nome e sobrenome, e várias outras situações.

A seguir, a Seção 2.2.1 classifica as consultas por similaridade de acordo com os trabalhos encontrados na literatura. Esta classificação é meramente didática, não se trata de uma classificação formal. O objetivo é contextualizar o tipo de consulta definido como escopo desta tese.

2.2.1 Tipos de consulta por similaridade

De acordo com os trabalhos encontrados na literatura (ARANTES et al., 2004; KOUDAS; SARAWAGI; SRIVASTAVA, 2006; GAO et al., 2004; COHEN, 2000; ZHONG et al., 2006; SCHALLEHN; SATTLER, 2003), diversas são as aplicações das funções de similaridade para produzir resultados mais adequados que o proces-

samento de consulta por coincidência exata. Observando os trabalhos na literatura sobre consulta por similaridade de forma geral, nota-se que existem duas características significativas no contexto de consulta por similaridade:

1. Semântica do problema; e
2. Delimitação do escopo do resultado.

A primeira tem um caráter de classificação, que correspondem aos tipos de consulta por similaridade encontrados na literatura. Classificar de acordo com a semântica do problema significa atribuir um contexto do que se espera como resultado da consulta por similaridade. Embora não seja uma classificação formal, de acordo com os trabalhos encontrados na literatura, a consulta por similaridade pode ser usada para atender dois tipos (ou categorias) de problemas:

1. Encontrar **elementos semelhantes ao objeto consultado** - Tipicamente, refere-se às consultas dos sistemas de Recuperação de Informações (RI), como acontece em sistemas de busca de informações na Internet, máquinas de busca em documentos XML, em banco de dados multimídia, em banco de imagens, entre outros. A principal característica desse tipo de consulta é que os argumentos informados na especificação da consulta possuem caráter informativo.

   Por exemplo, nas consultas sobre dados multimídia, o usuário fornece uma idéia aproximada do que deve ser recuperado. Propriedades do objeto consulta indicam que o resultado deve conter argumentos semelhantes em diversos aspectos (cor, intensidade, forma, entre outros), ou ainda referem-se a exemplos do que se espera como resposta. Através da interação com o usuário ocorre o refinamento da consulta para ajuste do resultado (ORTEGA-BINDERBERGER, 2002). A combinação do resultado de funções de similaridade específicas permite a recuperação de objetos semelhantes, pois o usuário geralmente não tem conhecimento do conteúdo exato armazenado. QBIC (*Query by Image Content*) (FLICKNER et al., 1995) e *Query by Visual Example* (HIRATA; KATO, 1992) são exemplos conhecidos de consulta sobre dados multimídia, que utilizam similaridade.

   Considerando busca textual, WHIRL (*Word-based Heterogeneous Information Representation Language* (COHEN, 1998) é um sistema de integração de informações, que utiliza uma linguagem baseada em lógica, pois é um subconjunto não recursivo da linguagem *Datalog* (GALLAIRE; MINKER; NICOLAS, 1984), acrescida de operadores como $\sim$ para incluir suporte à consulta sobre documentos estruturados. Para permitir suporte a documentos XML foi desenvolvida outra linguagem, a XML-QL (CHINENYANGA; KUSHMERICK, 2002), que faz um mapeamento dos dados XML para o modelo relacional e as condições de similaridade são traduzidas em sub-consultas, que são avaliadas usando WHIRL.

   Nos trabalhos que permitem busca por similaridade sobre dados contidos em arquivos XML, o que diferencia é o grau de flexibilidade permitido. Geralmente permitem três tipos de consulta: (i) somente por conteúdo; (ii) conteúdo vago e estrutura exata e (iii) tanto conteúdo quanto estrutura vagos. A determinação de conteúdo ou estrutura vaga é feita por uma função de similaridade. Pode-se destacar três exemplos de consulta sobre dados XML: Tijah,

TopX e TReX. *Tijah* (LIST et al., 2003, 2005) utiliza a álgebra por regiões (SRA (*Score Region Algebra*) (CONSENS; MILO, 1998)), onde os documentos da coleção são representados por regiões e armazenados em um banco de dados XML nativo, chamado *MonetDB* (BONCZ, 2002). Cada região possui descritores para que possam ser comparados com o termo da consulta e um escore, definido pela freqüência dos elementos desta região em relação às demais regiões baseado no modelo vetorial (BAEZA-YATES; RIBEIRO-NETO, 1999). TopX (THEOBALD; SCHENKEL; WEIKUM, 2005) combina diferentes técnicas como o *Threshold Algorithm* (FAGIN; LOTEM; NAOR, 2001) e probabilidade estatística para permitir o suporte de consultas por similaridade sobre arquivos XML, assim como aumentar o desempenho e otimizar os recursos disponíveis. TopX usa o modelo Okapi BM25 (ROBERTSON; WALKER, 1994) para calcular o escore, com parâmetros obtidos através do processamento prévio da coleção. Utilizando listas invertidas como forma de indexação, TopX mantém algoritmos de navegação em pré e pós-ordem da estrutura dos arquivos XML acessando os dados diretamente na coleção. Com o auxílio de tabelas de acesso seqüencial, TopX calcula os escores e demais dados estatísticos que permitem otimização da consulta em tempo de execução. TReX (ALI et al., 2006) é semelhante ao TopX para calcular o escore usando o modelo Okapi BM25 (ROBERTSON; WALKER, 1994). Entretanto, TReX utiliza persistência em banco de dados, usando *Berkeley Database*[1] para o armazenamento do conteúdo separado da estrutura do documento. A estratégia de indexação também é diferente, uma vez que TReX utiliza o conceito de *Path Summaries* (BARTA; CONSENS; MENDELZON, 2005) para indexar a estrutura. *Path summaries* fornecem um identificador padrão para todos os caminhos iguais (da raiz até a folhas) da estrutura dos arquivos XML que permite uma forma compacta de armazenamento e facilidade de acesso ao conteúdo durante o processamento da consulta.

2. Encontrar **representações do mesmo objeto** - Refere-se ao estilo tradicional de consultas sobre banco de dados (BD), denominadas de consultas complexas ou do tipo SPJ (select-project-join). O argumento da consulta tem caráter restritivo através do predicado da consulta. Somente elementos (ou tuplas) que satisfazem o predicado da consulta devem fazer parte do resultado da consulta. Durante a decomposição da consulta em partes, onde cada parte é avaliada por um tipo de operador, é o operador que implementa o critério de similaridade. Da mesma forma que no processamento da consulta tradicional, o resultado de um operador é usado como entrada para outro operador. Por esta razão, a restrição imposta pelo predicado deve ser mantida durante todo o processo de avaliação da consulta, inclusive com o uso de funções de similaridade pelos operadores intermediários. Para esta categoria de consultas, onde as funções de similaridade inseridas nos operadores tem a finalidade de encontrar variações dos mesmos objetos reais. E isto somente é possível com o uso de delimitadores. Exemplos desse tipo de consulta são apresentados na Seção 2.3.

[1]Berkeley Database mantém licença de código aberto, com versão proprietária (*http* : *//en.wikipedia.org/wiki/Berkeley*$_D B$ e *http* : *//www.oracle.com/database/berkeley − db/index.html*).

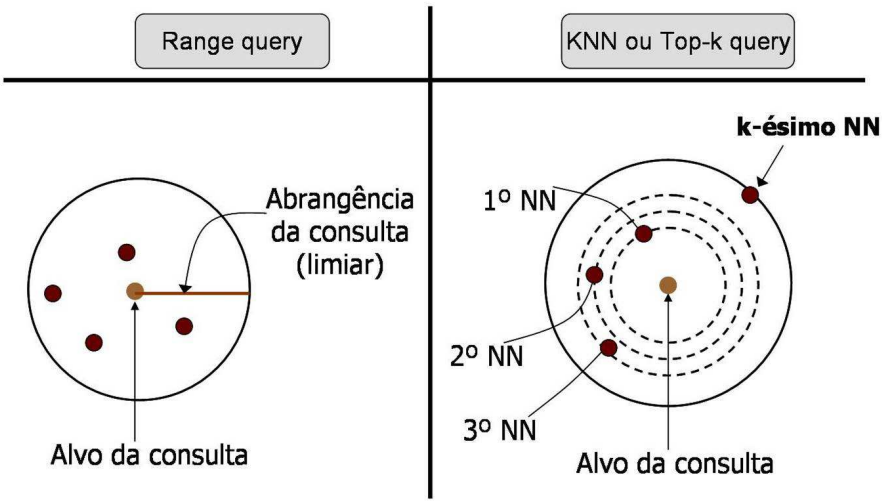

Figura 2.1: Representação de uma consulta delimitada por abrangência (*Range query*) e por quantidade (*Top-k* ou *KNN query*).

A segunda característica está relacionada à forma de restrição do resultado, uma vez que a função de similaridade produz um *ranking* de todos os elementos da coleção. Então, um critério de corte delimitando o escopo precisa ser estabelecido para que o resultado possa ser utilizado de forma viável em ambientes reais. Existem duas formas de delimitação do escopo, conforme mostrado na Figura 2.1, que podem ser aplicadas aos dois tipos de consulta por similaridade mencionados acima:

1. **Abrangência**: Refere-se ao raio ou escopo da consulta. É determinado por um limiar, i.é, um valor numérico que representa o valor mínimo do escore de similaridade aceito. As consultas que utilizam essa estratégia de delimitação do resultado são denominadas consultas por abrangência, ou ***range queries***. A consulta por abrangência recupera todos os elementos cuja similaridade com o objeto consulta estão dentro do escopo do limiar. O resultado é ordenado pelo escore de similaridade, que também determinada o número de elementos recuperados. Portanto, o número de elementos não é conhecido antes da execução da consulta. Portanto, uma `consulta por similaridade por abrangência` pode ser definida de acordo com o critério de delimitação da seguinte forma: seja τ o limiar (*threshold*), P um elemento qualquer de uma coleção DB que determina o critério de corte do resultado, $\mathcal{F}$ uma função de similaridade e Q um objeto a ser consultado:

   $$\{P \in DB; \mathcal{F}(P,Q) \leq \tau\}$$

   Os trabalhos que utilizam operadores por similaridade geralmente se baseiam em uma álgebra que agrega suporte às funções de similaridade, como por exemplo, a álgebra baseada em escore (LI et al., 2004).

   Em situações onde não se tem definido o número de resposta, ou o usuário especifica o grau de qualidade mínima necessário para o resultado da consulta, é a aplicação de operadores delimitados por abrangência, ou *range queries*. A característica principal desses operadores é a definição de um limite que estabelece a abrangência do operador de acordo com o escore de similaridade. Schal-

lenhn (SCHALLEHN; SATTLER; SAAKE, 2004; SCHALLEHN; SATTLER, 2003) apresenta operadores relacionais que foram redefinidos para processar dados que não podem ser processados por igualdade. Os operadores apresentados por Schallenhn utilizam a função de similaridade *edit distance* (BAEZA-YATES; RIBEIRO-NETO, 1999).

Um predicado por similaridade p consiste de uma função de distância ou similaridade e um limiar (*threshold*) t. Na especificação da consulta é definida a função que calcula as distâncias entre os atributos envolvidos na operação de junção. O escore de similaridade é limitado por t para definir quais elementos devem fazer parte do resultado, desde que atendam o predicado p. Por exemplo, tem-se uma consulta especificada em uma linguagem do tipo SQL da seguinte forma:

```
select *
from r1 similarity join r2
where edDist(r1.a1,r2.a1)
threshold ≥ 0.9
```

O exemplo acima mostra o uso de um operador de junção que utiliza a função de distância, *edit distance*, para estabelecer o critério de similaridade do atributo de ligação (a1). Dessa forma, tuplas que correspondam em 90% de similaridade passam a fazer parte do resultado. É aceito que exista uma variação de 10% uma vez que um limiar de 1.0 seria correspondente ao critério de igualdade.

2. **Quantidade**: Refere-se ao número de elementos requerido no resultado. O número k de elementos é especificado pelo usuário na definição da consulta. Assume-se que os k elementos mais relevantes devem ser retornados. As consultas que determinam o número dos elementos mais relevantes que devem ser retornados são denominadas consultas por quantidade, ou ***top-k queries***. As consultas limitadas por quantidade de elementos que devem ser retornados, representam as consultas por vizinhança, conhecidas como ***KNN (K-Nearest Neighbor) queries*** e amplamente utilizadas na área de busca por similaridade sobre domínios mais complexos, como por exemplo nas consultas multimídia (FALOUTSOS, 1996). *KNN queries* têm por objetivo retornar os k vizinhos mais próximos determinados por uma função de distância, que corresponde ao valor inverso de similaridade, pois quanto mais próximo o valor de distância é menor, porém os objetos são mais similares. Portanto, como pode ser observado na Figura 2.1 a proximidade dos objetos em relação ao objeto consulta é que determina a ordem do *ranking*. O valor de k, que determina quantos elementos serão retornados, é informado na especificação da consulta. Portanto, uma `consulta por similaridade por quantidade` pode ser definida de acordo com o critério de delimitação da seguinte forma: seja A um subconjunto dos K elementos mais relevantes da coleção DB, P um elemento qualquer da coleção DB, $\mathcal{F}$ uma função de similaridade, Q um objeto a ser consultado:

$$K = |A|, A \subseteq DB, \forall P \in A, P' \in (DB - A); \mathcal{F}(P,Q) \leq \mathcal{F}(P',Q)$$

Utilizando a estratégia de combinar condições no predicado da consulta através de uma extensão da linguagem SQL, o referido trabalho utiliza como base para

a avaliação da consulta a álgebra baseada em escore (LI et al., 2004). Um exemplo de consulta utilizando operadores limitados por quantidade (ILYAS; AREF; ELMAGARMID, 2003) pode ser descrito em uma consulta sobre uma base de dados que auxilia os usuários a encontrarem hotéis, restaurantes e museus em determinada cidade. Um consulta especificada em linguagem tipo SQL pode ser descrita como:

select *
from hotel h, restaurante r, museu m
where C_1 `AND` C_2 `AND` C_3 **order by** $média(p_1, p_2, p_3)$
`limit` k

Supondo que o usuário esteja interessado em um hotel com categoria 3 estrelas, almoçar em um restaurante com comida italiana, e não gastar mais de 100 reais no restaurante e na diária do hotel. Os predicados possuem as condições C_1 : *h.nr_ estrela = 3*, C_2 : *r.tipo_ cozinha = italiana* e C_3 : *(h.preço + r.preço)* < 100. Os critérios de classificação do resultado são especificados por funções específicas de conhecimento semântico do domínio, definidos pela combinação média de predicados como p_1 : *barato(h.preço)*, p_2 : *ambiente(h.tipo_ambiente)*, p_3 : *perto(r.endereço,m.endereço)*. Alguns predicados funcionam como filtro e podem ser implementados como seleção booleana (C_1 e C_2) e como junção booleana (C_3), suportados pelos operadores tradicionais. Predicados como p_1 podem ter implementações com métrica específica por domínio, aceitando formatos diferentes (COHEN; RAVIKUMAR; FIENBERG, 2003). Um predicado como p_2 pode ser implementado por operadores como seleção por similaridade (LI et al., 2004) e o p_3 requer junção por similaridade (ILYAS; AREF; ELMAGARMID, 2003). O limite definido em k produz a quantidade de respostas esperada pelo usuário.

Conforme exposto, o tipo de consulta nesta tese considera uma consulta por similaridade com um delimitador por abrangência, utilizada nos mecanismos de consulta por similaridade. Entretanto, a estratégia adotada para no desenvolvimento desta tese, pode ser aplicada para qualquer contexto onde seja necessário estimar um limiar adequado e avaliar o resultado do *ranking* obtido. Por esta razão, diferentes aplicações de consultas por abrangência encontrados na literatura são apresentados na Seção 2.3.

2.2.2 Discussão sobre consultas por similaridade

Embora normalmente sejam tratados como operadores independentes, os operadores por abrangência e por quantidade podem ser combinados na mesma consulta. A combinação dos dois critérios de delimitação (ARANTES et al., 2003, 2004) tem por objetivo a otimização do processamento da consulta envolvendo conjunções e disjunções de resultados provenientes de operadores intermediários como acontece nos processadores relacionais. Experimentos realizados com os operadores combinados (ARANTES et al., 2004) mostram ganhos em termos de desempenho e eficiência que podem ser usados na otimização do processamento da consulta.

Geralmente, os esforços encontrados na literatura concentram-se em desenvolver algoritmos que possam implementar o critério de corte de duas formas: (i) especificado na consulta, à escolha do usuário, ou (ii) especificado ou determinado no

operador, como seleção ou junção, de acordo com a decomposição da consulta pelo processador para obter o resultado.

Observou-se que a literatura é vasta no uso e aplicações de funções de similaridade. Tanto aplicados à consulta por similaridade, como foi apresentado neste capítulo, como em outras áreas como Recuperação de Informações, dados multimídia, imagens, entre outras. Entretanto, buscando por estratégias para avaliar e medir a qualidade das funções de similaridade nota-se a escassez de pesquisas mais aprofundadas.

2.3 Aplicações de consultas por abrangência

O problema tratado no contexto desta tese tem por objetivo utilizar funções de similaridade para encontrar representações distintas do mesmo objeto. Embora tenha sido optado pelo termo consulta vaga, esse mesmo problema é encontrado em diversas aplicações, mesmo com outros nomes. Alguns exemplos são: *record linkage* (WINKLER, 1999; FELLEGI; SUNTER, 1969), *merge/purge problem* (HERNANDEZ; STOLFO, 1995), *duplicate detection* (BILENKO; MOONEY, 2003; SARAWAGI; BHAMIDIPATY, 2002a; MONGE; ELKAN, 1997), *vague queries* (MOTRO, 1988), *hardening soft databases* (COHEN; KAUTZ; MCALLESTER, 2000), *reference matching* (MCCALLUM; NIGAM; UNGAR, 2000), *entity name matching* (COHEN; RICHMAN, 2002), *object matching* (DOAN et al., 2003), *object identification* (TEJADA; KNOBLOCK; MINTON, 2001), *instance matching* (RAHM; DO, 2000), consultas imprecisas (NAMBIAR; KAMBHAMPATI, 2003) e assim por diante. Devido à amplitude e diversidade do estudo deste problema na literatura, foram selecionados alguns trabalhos mais representativos.

De acordo com as características em comum, foram selecionados os trabalhos mais representativos e agrupados em três grupos relacionados com esta tese. Inicialmente, na Seção 2.3.1, são apresentados trabalhos que descrevem mecanismos de consulta por similaridade sobre banco de dados. Em seguida, estratégias de casamento aproximado de instâncias são discutidas na Seção 2.3.2, que englobam *record linkage*, *instance matching*, *deduplication* e *entity resolution*. Também foram incluídos trabalhos que exemplificam o problema de encontrar representações do mesmo objeto em coleções ou banco de dados resultantes de integração, durante o processo denominado *data cleaning*. O terceiro grupo de aplicações apresentado na Seção 2.3.3, refere-se às soluções e algoritmos de junção por similaridade.

2.3.1 Consultas por similaridade

Considerando mecanismos de processamento de consulta por similaridade, existem diversos trabalhos na literatura. Foram selecionados alguns mais representativos de acordo com o tipo de problema abordado nesta tese.

O uso de consultas vagas é apresentado por Motro no sistema *Vague* (MOTRO, 1988), que estende a abordagem relacional para permitir o suporte para consulta por similaridade. Motro apresenta um modelo probabilístico para calcular a similaridade, assim como um mecanismo de avaliação da qualidade o resultado de funções de similaridade. Cada domínio de atributo possui uma ou mais funções de similaridade específicas para calcular a distância entre os valores dos atributos. De forma geral, as consultas são redefinidas utilizando-se um operador de similaridade definido como "*similar-to*" representado por $\sim$, incluído na linguagem de consulta.

Valores de distância individuais de cada atributo são combinados em um único valor de relevância que representa a tupla. Este valor é usado para ordenar o resultado produzido.

O modelo de consulta é dependente da qualidade da função de similaridade associada a cada domínio de atributo definida pelo usuário. O sistema *Vague* permite ao projetista do banco de dados quatro escolhas: usar funções de similaridade pré-definidas embutidas no próprio mecanismo de consulta; fornecer um procedimento para calcular as distâncias entre cada par de elementos de um domínio; fornecer uma relação que armazene a distância entre cada par de elementos; ou usar uma relação de referência, ou seja, uma relação do banco de dados que contém as chaves apontando para as distâncias entre os valores dos atributos. As distâncias entre os valores dos atributos são combinadas para gerar distâncias entre tuplas. Cabe ao projetista também definir a fórmula para a combinação das distâncias de cada atributo, para definir o critério de ordenação do resultado.

O sistema *Vague* é limitado por abrangência através de dois parâmetros associados a cada métrica e domínio correspondente: diâmetro e raio. O `diâmetro` é o limite superior entre todas as distâncias do domínio, usado para estimar distâncias desconhecidas e representa a distância máxima entre os valores dos atributos. O `raio` estabelece uma vizinhança padrão, i.é, um valor que define o limiar. Portanto, sendo x e y valores de atributos de um mesmo domínio, $\mathcal{F}$ a função de similaridade para o domínio e r o raio, respectivamente, o operador de similaridade é definido como $x \sim y$ se $\mathcal{F}(x, y) \leq r$.

O sistema *Vague* possui um interpretador de consultas que auxilia e direciona o usuário para selecionar a função de similaridade mais adequada para processar a consulta. Por exemplo, uma consulta sobre filmes, onde o usuário especifica um determinado nome de filme, o interpretador solicita intervenção para indicar se a similaridade deve ser: (i) por filmes com títulos parecidos; (ii) por filmes com atributos parecidos; (iii) por comparação exata. Assim, através de perguntas com intervenção do usuário, o interpretador procura obter dados para processar a consulta e produzir o resultado de forma mais satisfatória.

Nambiar (NAMBIAR; KAMBHAMPATI, 2003) apresenta o sistema IQE (*Imprecise Query Engine*), que implementado como um *middleware*, o sistema IQE permite o processamento de consultas imprecisas sobre banco de dados. A partir de uma consulta Q sobre uma relação R, o resultado é um conjunto de tuplas ordenadas de acordo com o valor de similaridade com Q. A estratégia de consulta consiste em extrair tuplas adicionais de R de forma que sejam similares a Q. Essas tuplas formam um repositório, pois, quando outra consulta for submetida, primeiro é verificado se é similar às consultas já realizadas.

A similaridade (estimada através da função de similaridade *Jaccard* (HAVELIWALA et al., 2002)) entre duas consultas é definida por dois parâmetros: (i) pelos dados compartilhados no predicado da própria consulta; e/ou (ii) pela semântica compreendida pelo usuário, através dos valores dos atributos envolvidos. Para cada par de consultas armazenadas é calculado o valor de similaridade, formando uma matriz de similaridade. As consultas que ocorrem com freqüência são armazenadas para reduzir o custo de processamento, evitando recalcular a similaridade de dados que estão no repositório. Quando uma consulta é especificada, primeiro é verificado se existe uma consulta similar que possa ser mapeada para uma consulta precisa e produzir o resultado diretamente, senão é calculado o valor de similaridade e armaze-

nado no repositório de consultas. O conceito de precisão embutido em uma consulta no modelo IQE refere-se ao grau de similaridade obtido (i) entre os argumentos definidos na consulta e os argumentos das consultas previamente armazenadas; e (ii) entre os valores dos atributos armazenados e os valores dos atributos especificados no predicado da consulta.

Uma aplicação de consulta por similaridade aplicada no contexto de sistemas ponto a ponto (*peer-to-peer systems*) é apresentado por Zhong (ZHONG et al., 2006). Como recuperar todos os objetos em uma rede ponto a ponto é impraticável, então através de sucessivos refinamentos a partir das respostas mais relevantes identificadas pelo usuário, é feita a avaliação estatística da qualidade das respostas às consultas submetidas. De forma simplificada, a estratégia proposta por Zhong é usar amostras de consultas avaliando as respostas de um conjunto de pontos selecionados como uma amostra inicial. Um modelo de qualidade baseado em probabilidade estatística estabelece os critérios para determinar o escore de das respostas obtidas para as consultas aproximadas comparado com o critério de qualidade definido pelo usuário. O resultado de consultas sobre uma amostra é propagado para os demais pontos da rede sem a necessidade de acesso real a todos os pontos. Em cada ponto avaliado, somente as consultas por similaridade que apresentam um escore acima de um limiar previamente determinado são propagadas para outros pontos, o que evita tráfego de rede desnecessário. O limiar é definido como um parâmetro definido de forma experimental. O escore é determinado entre o objeto consulta e os elementos da amostra, através do uso de descritores de imagens como o histograma.

2.3.2 Identificação de instâncias duplicadas

O problema de identificação de instâncias duplicadas, conhecido como deduplicação ou *record linkage*, é estudado na literatura sob três aspectos: (i) definição de funções de similaridade apropriadas para detectar instâncias ou registros duplicados; (ii) desenvolvimento de ferramentas amigáveis para melhorar a qualidade dos dados e (iii) estudo de técnicas de escalabilidade para fusão de grandes conjuntos de dados. O primeiro está diretamente relacionado com esta tese, uma vez que o uso de funções de similaridade apropriadas para o domínio restá diretamente relacionado com a qualidade do resultado produzido pela função de similaridade. Quanto a escalabilidade, a forma de processamento utilizada em banco de dados é a implementação de operadores de junção, os quais, acrescidos de funções de similaridade permitem maior flexibilidade no processamento, desde que delimitados por um limiar, conforme discutido na Seção 2.3.3.

Uma extensa e aprofundada revisão sobre métodos de detecção de duplicidades pode ser encontrado em (ELMAGARMID; IPEIROTIS; VERYKIOS, 2007). De maneira genérica, as soluções adotadas para o problema de deduplicação atuam em duas fases: (i) casamento (*match*) – identificam a similaridade entre os atributos, gerando pares de registros duplicados com o respectivo escore; e (ii) (*merge*) – aplicam um conjunto de regras de fusão, conectando os registros duplicados, geralmente baseando-se na similaridade de múltiplos atributos ou em probabilidade. Em certos casos, após a fusão, os registros duplicados são removidos automaticamente, técnica conhecida como *merge-purge* (HERNANDEZ; STOLFO, 1995).

As soluções adotadas na literatura para identificar instâncias duplicadas variam como através do uso de operadores de junção por similaridade (GRAVANO et al., 2001; KOUDAS; SARAWAGI; SRIVASTAVA, 2006), ou através do uso de funções de

similaridade diretamente no processamento de texto como *edit distance* (GRAVANO et al., 2001) ou no processamento de documentos, como TF/IDF (CHAUDHURI et al., 2003).

É intuitivo que a fase mais complexa é decidir quais os parâmetros que definem a fusão, i.é, qual é o limiar que determina que dois objetos podem ser unidos. Outros trabalhos (BILENKO et al., 2003; SARAWAGI; BHAMIDIPATY, 2002b) priorizam o desenvolvimento de técnicas de fusão eficientes, utilizando mecanismos de aprendizado de máquina. Tais técnicas têm por finalidade gerar as regras de fusão dos registros duplicados, utilizando-se dos dados históricos obtidos através de coleções de treinamento.

Usando uma abordagem baseada em amostragem para estabelecer um limiar inicial, o trabalho apresentado por Li (LI; JIN; MEHROTRA, 2006) recalcula o limiar para cada nova amostra e estabelece quais instâncias estão duplicadas. O algoritmo *StringMap* apresentado, seleciona pares entre as amostras e calcula a distância euclidiana dentro de um limite máximo de distância, que é alterado de acordo com o limiar obtido na amostra inicial. Diversas amostras são geradas e novos limiares são calculados a medida que o método é melhorado e nos valores limites são obtidos de forma que possam ser capturados o número máximo de pares similares possíveis. As amostras são compostas por pares de elementos, geradas de três formas: (i) aleatória simples, onde os elementos são obtidos de um único conjunto de dados; (ii) aleatória composta, cujos elementos são obtidos de dois conjuntos dados distintos; e (iii) ordenação lexicográfica, onde os pares são sorteados baseados na semelhança de com um conjunto inicial de elementos. De acordo com os experimentos apresentados em (LI; JIN; MEHROTRA, 2006), a amostragem por ordenação lexicográfica apresenta os melhores resultados comparados com os outros dois métodos, pelo fato de estar mais próximo ao limiar real definido por um usuário humano. A ordenação lexicográfica aumenta a chance dos pares representarem instâncias duplicadas. A estratégia de mapeamento das instâncias duplicadas é usada para o processamento de consultas. Não é mencionado nada a respeito de avaliação da qualidade dos resultados obtidos em função da mudança do limiar.

Um estudo semelhante, denominado Resolução de Entidades (*Entity Resolution*) é apresentado por Molina (GARCIA-MOLINA, 2006), como parte do Projeto SERF (BENJELLOUN et al., 2006), cujo objetivo é desenvolver um mecanismo genérico de consulta que permita a identificação e a fusão de representações diferentes do mesmo objeto do mundo real. A estratégia adotada considera as operações de casamento e fusão como funções que se adaptam e se encaixam em qualquer aplicação, funcionando como caixa-preta. Dessa forma, os autores propõe o uso de funções de similaridade para a fase de casamento combinadas com um limiar para estabelecer a fusão. Entretanto, não é especificado como definir o limiar. Em um trabalho mais recente (MENESTRINA; BENJELLOUN; GARCIA-MOLINA, 2006), é usado um modelo probabilístico que estima valores de confiança a partir dos dados armazenados. A fase de fusão adiciona um novo registro referenciado os registros duplicados a que se refere e registra um determinado grau de confiança estabelecido pelo modelo probabilístico.

Dados duplicados resultantes de integração para formação de datawarehouses, podem contribuir significativamente para fornecer informações equivocadas, que não refletem a realidade dos dados armazenados. De maneira geral, técnicas conhecidas como *data cleaning* procuram identificar instâncias repetidas, o que caracteriza o

mesmo problema de encontrar diferentes representações de um mesmo objeto. Existem diversos trabalhos nesta linha, entre eles o algoritmo de casamento aproximado, denominado *fuzzy match algorithm* (CHAUDHURI et al., 2003). Utilizando uma função de similaridade, o referido algoritmo retorna tuplas mais próximas dentro de um limiar de similaridade estabelecido pelo usuário. A similaridade é medida pela função *edit distance* e por TF-IDF (BAEZA-YATES; RIBEIRO-NETO, 1999), conhecido método de pesos baseado na freqüência de termos usado nos sistemas de RI. Mais recentemente, Chaudhuri (CHAUDHURI; GANTI; KAUSHIK, 2006) apresenta um operador específico para identificar e remover duplicidades.

Outra abordagem apresentada em (GUHA et al., 2004) gera *rankings* individuais para cada atributo presente na consulta e combina esses *rankings* para ter um *ranking* global. Usando coeficiente de Spearman como forma de correlação dos *rankings* individuais, o *ranking* global é gerado através de operações de casamento aproximado das instâncias. Os algoritmos propostos para o casamento aproximado visam a minimizar o custo de processamento para selecionar os k elementos mais relevantes, onde k é determinado pelo usuário na especificação da consulta. Infelizmente, o referido trabalho não apresenta experimentos para avaliar a qualidade do *ranking* criado pelos algoritmos de casamento aproximado.

2.3.3 Junções por similaridade

De forma genérica, uma junção por similaridade de duas relações R e S, onde ambas contém a coluna A corresponde à junção R $JOIN_\theta$ S e θ corresponde ao predicado $\mathcal{F}(R.A, S.A) > \alpha$, para uma determinada função de similaridade $\mathcal{F}$ e um limiar α. Geralmente a junção por similaridade é expressa em SQL através da implementação das funções de similaridade como UDFs(*user-defined functions*), que são funções definidas pelo usuário, suportadas pela maioria dos sistemas gerenciadores de banco de dados modernos (ULLMAN; GARCIA-MOLINA; WIDOM, 2002). Conseqüentemente, técnicas especializadas permitem maior eficiência no processamento de junção por similaridade. Entre estas técnicas especializadas, certos métodos visam a personalização para determinadas funções de similaridade (GRAVANO et al., 2003, 2001; ANANTHAKRISHNA; CHAUDHURI; GANTI, 2002).

Gravano (GRAVANO et al., 2003, 2001) apresenta algoritmos de junção por similaridade utilizando atributos do tipo texto, ocorrendo junção de cadeias de caracteres. O primeiro trabalho é baseado na identificação de todos os pares similares usando similaridade textual calculada por cosseno, de acordo com o modelo vetorial (BAEZA-YATES; RIBEIRO-NETO, 1999). Utiliza pesos derivados do cálculo de freqüência com que os termos (ou seja, as palavras) ocorrem no texto, conhecida como TF-IDF. A idéia apresentada no segundo trabalho é parecida, porém com porções de texto de tamanho fixo agrupadas primeiramente por igualdade, com intuito de bonificar palavras com radicais iguais.

2.4 Avaliação da qualidade dos resultados obtidos por funções de similaridade

As medidas de avaliação de funções de similaridade mais utilizadas em consultas por similaridade são baseadas em R&P. Na Seção 2.4.1 são apresentados os fundamentos e exemplos necessários para o entendimento desta tese. A Seção 2.4.2 apresenta os conceitos relativos à precisão média. Na Seção 2.4.3, descreve-se um

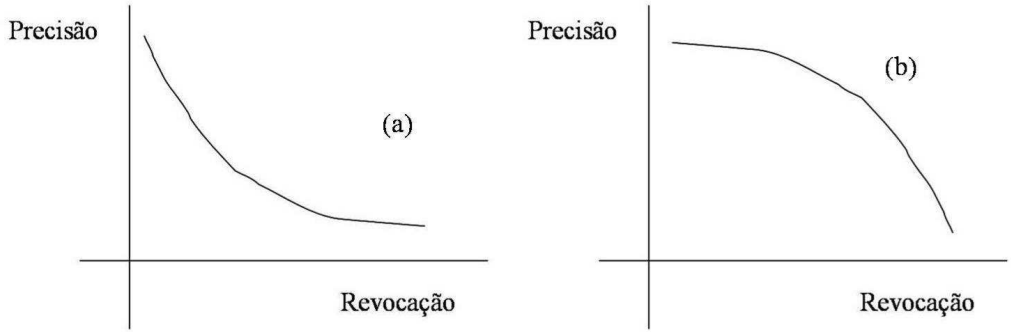

Figura 2.2: Exemplos de curvas de Revocação e Precisão.

exemplo de aplicação de R&P para avaliação da qualidade de funções de similaridade.

2.4.1 Medidas baseadas em revocação e precisão

Medidas de revocação (*recall*) e precisão (*precision*) (R&P) são amplamente conhecidas em sistemas de Recuperação de Informações (RI) como forma de avaliar a qualidade do resultado obtido por um mecanismo de busca. Revocação representa o número de elementos recuperados. Precisão é a medida do quanto esses objetos recuperados correspondem ao predicado informado pelo usuário. Definindo esses valores de acordo com os sistemas de RI (BAEZA-YATES; RIBEIRO-NETO, 1999), pode-se considerar uma consulta q sobre uma coleção de dados C. Esta consulta q possui um conjunto de elementos relevantes R que devem ser retornados. Portanto, $|R|$ é o número de elementos que devem ser retornados para determinada consulta. Supondo que uma consulta q é submetida sobre uma coleção C e produz um conjunto de instâncias A como resultado. Portanto, $|A|$ é o número de elementos recuperados e Ra é a intersecção entre R e A. Ou seja, $|Ra|$ é o número de elementos retornados corretamente. Conforme apresentado por Baeza (BAEZA-YATES; RIBEIRO-NETO, 1999), a revocação e a precisão são medidas definidas por:

$$\text{Revocação} = \frac{|Ra|}{|R|} \tag{2.1}$$

$$\text{Precisão} = \frac{|Ra|}{|A|} \tag{2.2}$$

Revocação é a fração de elementos relevantes que foram recuperados e a precisão, indica quantos desses recuperados estão corretos em função da consulta especificada pelo usuário.

As medidas definidas nas equações 3.2 e 3.1 assumem que todas as respostas do conjunto A foram examinadas. Revocação e precisão são medidas ortogonais, pois a medida que uma medida aumenta a outra tende a diminuir. Uma representação gráfica chamada de Curva de Revocação e Precisão permite avaliar a qualidade do resultado recuperado. Como pode ser observado na Figura 2.2, a medida que aumenta a revocação, o valor de precisão cai, e vice-versa. O eixo da revocação é geralmente determinado por 11 níveis de recuperação determinados por 11 pontos de corte (0.0, 0.1, 0.2, ..., 1.0). Quando não se tem o valor correspondente de revocação para um dos 11 pontos, o novo valor é interpolado pela média entre os valores mais

Limiar	R1	R	P
0.9	1.00	0.67	1.00
	0.90		
0.8	0.82	0.75	0.98
	0.80		
	0.80		
0.6	0.68	0.92	0.73
	0.65		
	0.65		
	0.63		
	0.61		
	0.60		
0.5	0.55	0.95	0.65
	0.55		
	0.53		
	0.52		
0.4	0.45	0.98	0.49
	0.43		
	0.42		
0.2	0.20	1.00	0.20

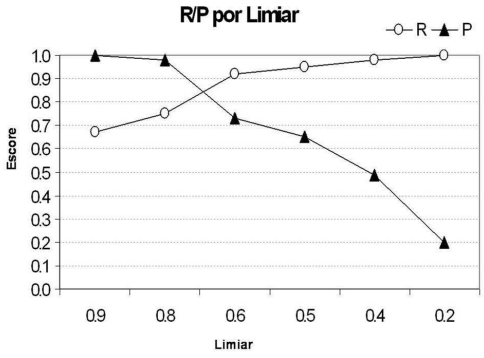

Figura 2.3: Exemplos de representação dos valores de R&P em vários limiares.

próximos. Geralmente, os limites são maximizados, para que possam estar próximos dos eixos.

A informação obtida com a representação gráfica da curva de revocação e precisão é a variação do escore a medida que mais elementos vão sendo recuperados. As variações mostradas na Figura 2.2 indicam que uma curva similar a (b) indica que a função mantém a precisão a medida que recupera mais elementos. Portanto, se comparada com a curva (a) na mesma Figura, é possível inferir que a função de similaridade que produziu a curva (b) é mais adequada, ou apresenta melhor qualidade.

Embora a representação da curva de R&P conforme a ilustração da Figura 2.2 seja mais conhecida como forma de avaliação da qualidade de sistemas de RI, optou-se por representar os valores de R&P por limiar. Esta escolha justifica-se pelo fato de que é possível informar um conjunto de limiares e dentre eles o método apresentado no Capítulo 3 seleciona o limiar mais apropriado ou analisar a qualidade que pode ser obtida em cada limiar. A Figura 2.3 mostra um exemplo com valores artificiais de Figura 2.3 mostrados em cada limiar na tabela à esquerda, e a respectiva curva de R&P por limiar, à esquerda, para os limiares definidos no eixo x. A conversão de uma curva de R&P por limiar para uma representação tradicional da curva de 11 pontos de R&P é trivial. A Figura 2.4 mostra o mesmo exemplo da Figura 2.3 no formato da tradicional curva de 11 pontos de revocação. Note-se que foi adicionada uma coluna na tabela da esquerda para a precisão interpolada (P_i), onde os valores destacados indicam que não haviam sido calculados os valores de R&P para este limiar. Então, o valor da precisão é interpolado pela média simples entre os valores próximos, como no caso do $P_i = 0.86$ ou por valores máximos ou mínimos entre os valores próximos.

Embora revocação e precisão sejam medidas bem conhecidas para avaliar a qualidade em sistemas de RI, existem algumas alternativas que combinam essas duas medidas. Conforme apresentado por Baeza (BAEZA-YATES; RIBEIRO-NETO, 1999) essas alternativas podem ser usadas no contexto desse trabalho:

Média Harmônica ou Medida F Combina revocação e precisão em um único valor. A média harmônica é calculada por:

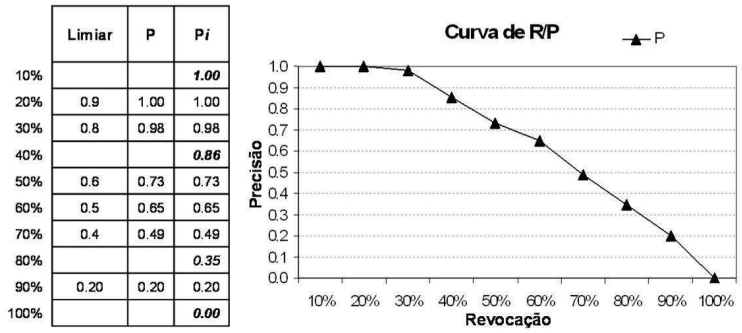

Figura 2.4: Exemplos de curva de revocação em 11 pontos com valores interpolados.

Tabela 2.1: Exemplo do resultado produzido por uma função de similaridade x.

Escore	**Elemento**	**Relevância**	**Limiar**
1.0000	Ranking in Databases	$*$	
0.9581	Ranking on Databases	$*$	0.9
0.7023	Relational Databases	$\circ$	0.7
0.6789	Ranking Correlation	$\circ$	
0.6767	Ranking on DBs	$*$	
0.6089	Rankin on DBs	$*$	
0.5543	Ranking and DBs	$*$	0.5
0.4412	Ranking on IR	$\circ$	
		$*$ - Relevante	
		$\circ$ - Irrelevante	

$$F(j) = \frac{2}{\frac{1}{r(j)} + \frac{1}{p(j)}} \tag{2.3}$$

onde $r(j)$ é a revocação do j-ésimo elemento do resultado no *ranking*, $p(j)$ é a precisão do j-ésimo elemento do resultado do *ranking* e $F(j)$ é a média harmônica de $r(j)$ e $p(j)$.

A função F assume valores no intervalo $[0, 1]$. O valor é 0 quando nenhum elemento relevante foi recuperado e é 1 quando todos os elementos recuperados são relevantes. A média harmônica F assume um valor alto somente quando ambos os valores de revocação e precisão são altos. Esta medida pode ser usada quando se espera a melhor combinação de valores altos, tanto para revocação quanto para precisão. A `Medida F` possui variações e é conhecida como `Medida` F_1 quando atribui um peso igual tanto para revocação quanto para precisão.

Medida E também combina os valores de revocação e precisão, porém atribuindo um peso para um ou para outro. Existem situações em que se faz necessário um peso maior em precisão, por exemplo, mesmo não recuperando todas as

instâncias. A medida E é definida por:

$$E(j) = 1 - \frac{1 + b^2}{\frac{b^2}{r(j)} + \frac{1}{p(j)}} \tag{2.4}$$

onde $r(j)$ é a revocação do j-ésimo elemento do resultado, $p(j)$ é a precisão do j-ésimo elemento do resultado e $E(j)$ é a medida de avaliação relativa a $r(j)$ e $p(j)$ e b é um parâmetro especificado pelo usuário que reflete a relativa importância de revocação e precisão. Para $b = 1$ a medida $E(j)$ funciona como o complemento da média harmônica $F(j)$. Valores de b maiores que 1 indicam que o usuário está mais interessado em precisão; por outro lado, valores de b menores que 1 indicam maior interesse em revocação.

2.4.2 Precisão média

A precisão média AP (*Average Precision*) (SALTON; LESK, 1968) é conhecida e referenciada como MAP (*Mean Average Precision*) porque geralmente representa os valores de precisão médios referem-se à várias consultas. Se várias consultas são avaliadas, o que geralmente acontece, primeiramente, cada *ranking* é obtido separadamente. Em seguida, o valor de MAP, média das precisões médias, pode ser obtido de duas formas (SALTON; LESK, 1968): (i) média-macro – onde primeiro é calculada a precisão para cada consulta em um ponto fixo de revocação comum. Em seguida, é calculado o valor médio das precisões médias (MAP) para todas as consultas realizadas. Geralmente esse tipo de comparação é usado para sistemas de RI, para comparar máquinas de busca, por exemplo, MAP@100, significa precisão média no retorno de 100 elementos; ou (ii) micro-média – em cada ponto de revocação, é calculada a precisão de cada consulta. Em seguida, são calculados os valores de precisão média considerando todas as consultas, no mesmo ponto de revocação. A micro-média é a mais utilizada quando se deseja fazer comparações entre sistemas distintos através da curva de 11 pontos de revocação, pois o MAP é calculado para cada valor de revocação, como foi feito nos experimentos desta tese.

As medidas de R&P são baseadas em um *ranking* de documentos retornadas por um sistema de busca, ou no contexto desta tese, por uma função de similaridade. A precisão média favorece documentos relevantes que retornam no início do *ranking*. A Equação 2.5 define o cálculo da média das precisões:

$$\text{MAP} = \frac{\sum_{r=1}^{N}(P(r) \times \text{rel}(r))}{\text{nr. de documentos relevantes}}, \tag{2.5}$$

onde r é um *ranking*, N o número de documentos recuperados , `rel()` uma função binária que determina relevância sobre um determinado *ranking*, e `P()` é a precisão no ponto de corte do *ranking*.

Supondo que se deseja avaliar um sistema de recuperação através de uma coleção de teste, que consiste de um conjunto de documentos, um conjunto de requisições de busca (tópicos ou consultas) e os resultados julgados relevantes. Assumindo x_k uma variável representando o grau de relevância do $k_{ésimo}$ documento na lista de *rankings* que o sistema gerou para um determinado tópico. Assumimos um relevância binária de julgamento.

$$x_k = \begin{cases} 1 & \text{se } k_{ésimo} \text{ documento é relevante} \\ \\ 0 & \text{se } k_{ésimo} \text{ documento é irrelevante} \end{cases} \tag{2.6}$$

Selecionando os m documentos nas posições iniciais do *ranking*, o valor de precisão deste conjunto de documentos por ser expresso por:

$$p_m = \frac{1}{m} \sum_{k=1}^{m} x_k \tag{2.7}$$

Pode-se notar que p_m pode ser interpretado como a média dos valores, $x_1, ..., x_m$, denotado por $\overline{x}_m$.

A precisão média é definida como "a média da precisão dos escores obtidos depois que cada documento relevante é recuperado, usando zero para o valor de precisão dos documentos relevantes não recuperados", que pode ser representado matematicamente usando p_m ou $\overline{x}_m$:

$$v = \frac{1}{R} \sum_{i=1}^{n} I(x_i) p_i = \frac{1}{R} \sum_{i=1}^{n} I(x_i) \overline{x}_i \tag{2.8}$$

onde R é o número total de documentos relevantes e n é o número de documentos incluídos na list (usualmente $n = 1000$). $I(x_i)$ é a função:

$$I(x_i) = \begin{cases} 0 & \text{se} x_i = 0 \\ \\ 1 & \text{outras situações} \end{cases} \tag{2.9}$$

No caso onde x_i seja uma variável binária, é possível simplificar para $I(x_i) = x_i$. Então, a precisão média pode ser representada por:

$$v = \frac{1}{R} \sum_{i=1}^{n} x_i p_i = \frac{1}{R} \sum_{i=1}^{n} \frac{x_i}{i} \sum_{k=1}^{i} x_k \tag{2.10}$$

Outro indicador, `Precisão em R`, é algumas vezes usado em experimentos de recuperação. Ele é definido como "precisão após R documentos que tenham sido recuperados", onde R é o número de documentos relevantes do tópico corrente. Pode ser expresso por:

$$v = \frac{1}{R} \sum_{i=1}^{R} x_i \tag{2.11}$$

Também existem outros indicadores clássicos para a montagem do *ranking* de saída. A soma do *ranking* de R documentos relevantes pode ser expresso como $\Sigma_{i=1}^{N} i I(x_i)$ onde N é o número total de documentos existentes na coleção. Estes documentos relevantes podem ser ordenados do 1º até a $R_{ésima}$ posição na lista de ordenação "ideal" onde a soma dos documentos ordenados é expressa por $\Sigma_{i=1}^{R} i$. A revocação normalizada (SALTON; LESK, 1968) é baseada na diferença das duas somas do *ranking*, e é definido formalmente por:

$$z_r = 1 - \frac{\sum_{i=1}^{N} i I(x_i) - \sum_{i=1}^{R} i}{R(N - R)} \tag{2.12}$$

O valor máximo da diferença de duas somas é representado por $R(N-R)$ porque a diferença é calculada através de $(N-R+1)+(N-R+2)+...+N-(1+2+...+R) = R(N-R)$. Conseqüentemente, $0 \leqslant z_r \leqslant 1$, onde 1 indica o melhor *ranking*.

Quando convertemos o indexador i, representando cada posição do documento, em $\log i$, o indicador é chamado de "precisão normalizada", sendo expressado matematicamente por:

$$z_p = 1 - \frac{\sum_{i=1}^{N} I(x_i) \log i - \sum_{i=1}^{R} \log i}{\log_N C_R} \tag{2.13}$$

onde $\log_n CR = \log(N-R+1) + \log(N-R+2) + \ldots + \log N - \log(\log 1 + \log 2 + \ldots + \log R)$.

2.4.3 Exemplo do uso de R&P como medidas de qualidade

A avaliação da qualidade de uma função de similaridade através de medidas de R&P requer a execução de várias consultas. Avaliar somente um *ranking* não reflete o comportamento de uma função de similaridade sobre uma coleção por expressar a avaliação em termos de R&P de apenas um elemento. Naturalmente, executar diversas consultas, de forma iterativa e mantendo-se a representatividade dos objetos consulta em relação à coleção, é uma forma de avaliar o comportamento de uma função de similaridade sobre uma determinada coleção. Um processo iterativo produz um conjunto de *rankings* que são avaliados em termos de R&P, em diferentes limiares previamente definidos. Em cada limiar, a média dos valores de R&P resultante do conjunto de *rankings* avaliados é o valor usado para estimar a qualidade da função de similaridade.

Tabela 2.2: Exemplo do resultado produzido por uma função de similaridade x.

Escore	**Elemento**	**Relevância**	**Limiar**
1.0000	Ranking in Databases	$*$	
0.9581	Ranking on Databases	$*$	0.9
0.7023	Relational Databases	$\circ$	0.7
0.6789	Ranking Correlation	$\circ$	
0.6767	Ranking on DBs	$*$	
0.6089	Rankin on DBs	$*$	
0.5543	Ranking and DBs	$*$	0.5
0.4412	Ranking on IR	$\circ$	
		$*$ - Relevante	
		$\circ$ - Irrelevante	

Por exemplo, a Tabela 2.2 mostra o exemplo de um *ranking* produzido por uma função de similaridade x, para um objeto consulta igual a "*`Ranking in Databases`*". Na primeira coluna é apresentado o escore, na segunda o elemento recuperado da coleção, na terceira a identificação de relevância e na quarta coluna, alguns limiares são previamente definidos. Lembrando que a indicação de relevância é determinada por um usuário, representando uma variação do mesmo objeto real.

Para medir a qualidade de um *ranking* através de R&P é necessária a identificação de quais elementos deveriam ser retornados para cada consulta realizada. Cabe ao usuário identificar os elementos retornados que são relevantes para a consulta

definida. Então, através da contagem de acertos e erros, é calculado o valor de R&P para certos limiares previamente definidos, geralmente dentro do intervalo [0,1]. O limiar representa um ou mais critérios de corte no *ranking*, delimitando o escopo da consulta e conseqüentemente os valores de R&P correspondentes. Embora o limiar mais adequado possa variar de acordo com a aplicação conforme exemplificado acima, o consenso adotado neste exemplo considera-se "ótimo" o limiar que busca maximizar R&P juntos.

Conforme apresentado no Capítulo 2, na Seção 2.4.1, uma medida de combinação de R&P é a `Medida F`, que permite atribuir pesos distintos para situações onde se deseja uma busca em largura, recuperando um maior número de elementos, ou para situações onde se deseja uma busca em profundidade, priorizando somente elementos relevantes. Portanto, o uso da `Medida F` permite o ajuste ponderado para combinar R&P e recebe a denominação de `Medida` F_1 quando são estabelecidos pesos iguais tanto para revocação quanto para precisão.

Tabela 2.3: Valores de R&P para os limiares da Tabela 2.2.

Limiar	**% Revocação**	**% Precisão**	**Medida F_1**
0.9	40%	100%	0.57
0.7	40%	66%	0.49
0.5	100%	71%	0.83

A Tabela 2.3 apresenta o cálculo de R&P para o *ranking* da consulta mostrado na Tabela 2.2. Além de R&P, a Tabela 2.3 mostra os valores obtidos pelo cálculo da `Medida` F_1. Analisando os valores obtidos pela `Medida` F_1 pode-se dizer que usando um limiar de 0.5 é mais apropriado neste exemplo para se obter o percentual máximo de R&P combinado. Se forem verificados os dados e o escore mostrados na Tabela 2.2, pode-se constatar que um limiar de 0.5 determina a recuperação dos elementos relevantes, com poucos elementos irrelevantes. Entretanto, esse valor é significativo somente para este exemplo uma vez que é obtido de um único *ranking* resultante da consulta "`Ranking in Databases`", conforme apresentado na Tabela 2.2.

Para cada um dos limiares determinados (0.9, 0.7 e 0.5, por exemplo) são calculados os respectivos valores de R&P como mostra a Tabela 2.3. Para obter uma estimativa mais confiável, é importante avaliar diversos *rankings*. Conseqüentemente, diversos valores de R&P são obtidos, um para cada *ranking*, em cada limiar. Calculando-se a média aritmética dos valores obtidos em cada limiar, tanto para revocação quanto para precisão, obtém-se a média dos valores de R&P obtidos para cada limiar: a média da revocação e a média da precisão.

Intuitivamente, pode-se inferir que quanto maior o número de *rankings* utilizados para obter a média dos valores de R&P, mais confiável é a estimativa. Considerando situações práticas é em sistemas reais, dificilmente consegue-se obter uma avaliação em termos de R&P devida a alta dependência de interação com o usuário. O método apresentado a seguir, mostra como o processo de amostragem pode ser usado para evitar a necessidade de avaliação exaustiva de todos os elementos da coleção.

2.5 Algoritmos de agrupamento por similaridade

Como o agrupamento por similaridade possui um papel importante para tornar o processo proposto semi-automático, nesta seção discutiremos alguns pontos importantes sobre os algoritmos de agrupamento por similaridade.

O objetivo dos algoritmos de agrupamento[2](*clustering*) (ALDENDERFER; BLASHFIELD, 1984) por similaridade é identificar grupos de elementos que são semelhantes, ou seja, possuem um grau de semelhança que faz com que pertençam ao mesmo grupo. Embora existam formas diferentes de classificação, os métodos de agrupamento podem ser classificados em duas categorias, como apresentado por Hartingan (HARTIGAN, 1975): hierárquicos, e não-hierárquicos (também conhecidos na literatura como divisivos ou particionadores). De maneira simplista, uma abordagem "*bottom-up*" agrupa elementos um a um enquanto que uma abordagem "*top-down*" inicia com um grande grupo e começa a separar e sub-grupos.

Uma revisão completa sobre o processo de agrupamento pode ser encontrada no trabalho desenvolvido por Jain (JAIN; MURTY; FLYNN, 1999), onde é apresentada uma taxonomia das diferentes abordagens para agrupamento de dados (*data clustering*). De um lado estão os métodos baseados em hierarquia, que permitem o uso de funções de similaridade ou medidas de distância, e do outro estão os métodos baseados em particionamento, baseados em estatística. Métodos hierárquicos produzem um aninhamento de uma série de partições (divisões) enquanto que os métodos particionadores produzem somente uma. Por definição, os métodos hierárquicos são aglomerativos, possuem uma abordagem "*bottom-up*", i.e., iniciam de forma que cada elemento é considerado um grupo e seguem através de combinações sucessivas entre os grupos, mesclando-os, até que certo critério seja satisfeito. Já os métodos divisivos poderiam ser considerados como uma abordagem "*top-down*", pois iniciam com um único grupo de todos os elementos e executam sucessivas divisões até que certo critério de parada seja encontrado.

Os hierárquicos são agrupamentos cuja forma se parece com uma estrutura de árvore. Por esse motivo, tais algoritmos demonstram relações de hierarquia entre os grupos formados. Já os processos não-hierárquicos constituem-se de partições isoladas ou disjuntas. Conforme indicado por Wives (WIVES, 2004), a hierarquia é extremamente interessante quando se deseja analisar relações de abrangência ou especificidade entre objetos, mas os usuários devem percorrer toda a estrutura para compreender as inúmeras relações entre eles. Já os agrupamentos não hierárquicos podem ser visualizados e compreendidos mais facilmente, desde que o número total de grupos formados não seja muito grande. Independente de serem hierárquicos ou não, os grupos podem estar completamente isolados (disjuntos) ou podem estar sobrepostos, i.e., o mesmo objeto pode estar em mais de um cluster.

Aldenderfer e BlashField (ALDENDERFER; BLASHFIELD, 1984) classificam os algoritmos levando em conta as seguintes categorias: hierárquicos aglomerativos (*hierarchical aglomerative*), hierárquicos divisivos (*hierarchical divisive*), de particionamento iterativo (*iterative partitioning*), de busca em profundidade (*density search*), fator-analíticos (*factor analytic*), de amontoamento (*clumping*) e grafo-teoréticos (*graph-theoretic*). Cada um desses, quando aplicado ao mesmo conjunto de dados, pode gerar resultados completamente diferentes. Uma apresentação detalhada de cada método pode ser encontrada no trabalho de Wives (WIVES, 2004).

[2]Métodos de agrupamento são sinônimos de conglomerados, segundo (WIVES, 2004)

Aqui são apresentados de forma superficial os três primeiros grupos por estarem mais próximos do contexto em estudo neste trabalho.

2.5.1 Hierárquicos aglomerativos

Os métodos hierárquicos aglomerativos (este termo é sumarizado na literatura especializada pelo acrônimo HACM - *Hierarchical Agglomerative Clustering Methods*) são os mais populares, inclusive em áreas como banco de dados e RI, e mais simples de serem implementados e compreendidos. Os grupos são formados através da inclusão de objetos em agrupamentos cada vez maiores, em forma de estruturas de árvore.

De forma geral os métodos de agrupamento aglomerativo baseiam-se em uma matriz de similaridade, a qual contém os valores de similaridade entre os pares de elementos do conjunto. O agrupamento inicia selecionando-se os dois elementos mais similares, formando um grupo. Após, novos elementos são adicionados ao grupo formado, utilizando regras de ligação que dependem do algoritmo utilizado. O processo continua até que todos os elementos sejam agrupados.

Existem quatro algoritmos principais (ALDENDERFER; BLASHFIELD, 1984) que implementam as regras de ligação para incluir novos elementos em cada grupo. São eles:

- ligação simples (*single linkage*) - adiciona elementos a um grupo somente se este elemento possuir alguma ligação com qualquer um dos elementos já presentes no grupo. Esse método utiliza a regra do vizinho mais próximo para juntar dois elementos. O algoritmo mais comum desse grupo é o SLINK (SIBSON, 1973);

- ligação completa (*complete linkage*) - cria agrupamentos fortemente acoplados, pois adiciona elementos somente se eles possuírem um nível de ligação específico com todos os elementos do grupo. Tem a tendência de gerar agrupamentos menores, compactos e hiper-esféricos, contendo elementos altamente similares;

- ligação mediana ou pelo valor médio (*average linkage*) - é calculada a similaridade média entre o elemento a ser adicionado e todos os elementos já presentes em um grupo. O novo elemento é adicionado ao agrupamento cuja média de similaridade for maior do que a dos outros. Essa similaridade média pode ser avaliada de diversas formas, sendo a mais comum a média aritmética;

- método de Ward - é uma variação dos anteriores, que busca otimizar a variância mínima entre os grupos, juntando os elementos cuja soma dos quadrados entre eles seja mínima ou que o erro desta soma (denominada ESS ou *Error Sum of Squares*) seja mínimo.

Qualquer que seja o algoritmo de ligação utilizado, o resultado final do processo de agrupamento é uma árvore, denominada de Dendrograma (dendrogram) (ALDENDERFER; BLASHFIELD, 1984), como exemplificado na Figura 2.5.

No Dendrograma as folhas (as letras A, B, C, D, E e F, encontradas na Figura 2.5) representam elementos completamente isolados. Caminhando-se em direção à raiz (parte superior da figura), cada bifurcação representa uma união entre elementos. Os valores existentes à esquerda do diagrama representam graus de similaridade,

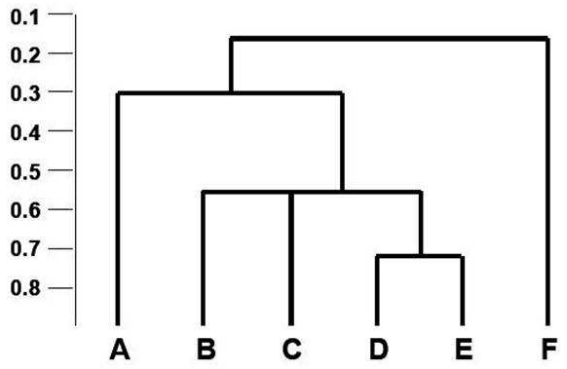

Figura 2.5: Ilustração de um Dendograma, onde as letras representam os objetos e as linhas indicam os grupos formados de acordo com o valor de escore (eixo y).

indicando em que nível de similaridade os elementos foram reunidos. A definição do número de grupos resultante ocorre quando um certo grau de similaridade é atingido, definido por um escore de similaridade mínimo que deve existir entre os membros de cada grupo.

No Dendograma pode-se notar que os métodos hierárquicos aninham os objetos, destacando relações de composição, abrangência e especificidade entre os elementos. Além disso, os métodos hierárquicos aglomerativos costumam ser rápidos, necessitando de $n-1$ passos para gerar o diagrama, onde n corresponde ao número de elementos. O maior problema é não poder mudar os grupos já constituídos, ou seja, após juntar dois elementos, eles não podem mais ser separados. Se essa junção, por algum motivo, estiver incorreta, ela permanece assim até o final do processamento, que tem que ser reiniciado (com outros parâmetros), para que se possa (tentar) corrigir o problema. Por esta razão, usar uma métrica adequada para processar a similaridade é fundamental para agrupar adequadamente.

De forma genérica, um algoritmo que define um agrupamento por similaridade funciona da seguinte forma: um algoritmo que implementa uma função de similaridade $\mathcal{F}_{sim}$ sobre um conjunto de dados D que calcula o escore entre todos os pares de elementos de um conjunto D denotado por $\mathcal{F}_{sim} : D \times D \mapsto \mathcal{R}_S \subseteq [0,1]$, i.é, $\mathcal{F}_{sim}(x,y) \mapsto s | (x \in D) \wedge (y \in D)$, onde $0 \leq s \leq 1$, formando uma `matriz de similaridade`, denotada por $\mathcal{M} = \bigcup_{i=1}^{z} x_i, y_i, s_i$, onde $z = (|D|^2)$. A partir de $\mathcal{M}$ é construído o processo de agrupamento de acordo com uma estrutura em forma de árvore, ou `dendograma`. Os elementos da matriz com maior escore são agrupados e a similaridade entre os elementos do grupo é recalculada usando $\mathcal{F}_{sim}$. Este processo se repete até que todos os elementos estejam em um único grupo. Então, o critério de corte é aplicado, determinando o número de grupos resultante.

2.5.2 Hierárquicos divisivos

Nos métodos hierárquicos divisivos todos os objetos são inicialmente alocados a um único grupo e este vai sendo dividido (ou partido) em grupos menores até que cada objeto esteja em um grupo separado (ALDENDERFER; BLASHFIELD, 1984). Para que possam constituir os grupos, os métodos divisivos testam todas as possibilidades de divisão de cada conglomerado existente, tornando-os computacionalmente ineficientes. Por esse motivo, eles são mais recomendados para dados que sejam descritos por variáveis binárias, pois sua complexidade de análise é menor, já

que, nesse caso, testa-se somente a presença ou não de um valor no objeto. Também são conhecidos como métodos de classificação.

2.5.3 Particionamento iterativo

Os métodos de partição iterativos são mais utilizados em áreas como mineração de texto (*text mining*).

Os métodos de particionamento iterativo dividem o conjunto de dados, ou seja, criam grupos através de diversas iterações. O algoritmo mais conhecido dessa categoria é o k-médias (k-*means*). No k-médias, o usuário indica o número de grupos desejado e o algoritmo de particionamento cria (de forma aleatória ou por outro processo) um conjunto inicial de partições (agrupamentos). A seguir, é definido um elemento representativo, também conhecido como centróide, de cada um desses grupos. O algoritmo analisa a distância ou similaridade desses com todos os elementos a serem agrupados. Após, cada elemento é alocado ao grupo cujo centróide esteja mais próximo e, ao ser incluído, este centróide é recalculado para refletir (e representar) esse novo elemento. O processo é repetido até que os centróides não mudem mais de posição (ALDENDERFER; BLASHFIELD, 1984).

Embora tais métodos permitam a correção de eventuais problemas pelas sucessivas iterações, são mais demorados que os hierárquicos e, dependendo da implementação, podem não suportar muitos elementos ou tornar o processo inviável do ponto de vista computacional.

O usuário deve informar o valor de k, o que muitas vezes se torna um problema, pois não se tem o conhecimento dos dados. Existem trabalhos mais recentes (WIVES, 2004) que apresentam estratégias para auxiliar na definição do valor de k.

2.5.4 Validação do processo de agrupamento

Existem diversos métodos para se medir a qualidade dos resultados produzidos por algoritmos de agrupamento por similaridade. Halkidi et. al (HALKIDI; BATISTAKIS; VAZIRGIANNIS, 2001) discutem critérios para avaliar os grupos criados, que podem ser resumidos em:

Critérios Externos – os grupos resultantes são avaliados comparando-os com um agrupamento pré-definido, construído por especialistas, definida como uma estrutura de teste. A validação é feita calculando-se índices de similaridade ou percentuais de acertos e erros entre o resultado obtido pelo agrupamento e o conjunto de teste, por métodos como a *Estatística de Rand*, *Coeficiente de Jaccard* e *Índice de Folkes & Mallows* ;

Critérios Internos – utilizam os próprios dados para realizar a validação dos resultados. Para tanto, são utilizados geradores de números aleatórios, baseados em técnicas denominadas de *Procedimentos de Monte Carlo*, com o intuito de gerar um conjunto de dados (artificial), cujas características sejam as mesmas dos objetos originais (reais), porém, sem os grupos. Ambos os conjuntos de dados (real e artificial) são submetidos ao algoritmo de identificação de agrupamentos e seus resultados são analisados por métodos estatísticos de validação, como a `Estatística` Γ `de Huberts` apropriados ao tipo de dado sendo processado;

Critérios Relativos – a validação é feita comparando a estrutura de clusters com

outras obtidas através da aplicação exaustiva do mesmo algoritmo, porém com diferenças nos parâmetros de entrada. Os melhores valores para cada parâmetro podem ser identificados pela `Estatística Γ de Huberts` ou por um índice de Dunn, que mede a dispersão de cada agrupamento.

Seja $C^* = C_1, C_2, \ldots, C_m$ um `Agrupamento Resultante` conforme a Definição 6 a partir de uma amostra $\mathcal{X}$, onde $C_{i=1}^m$ representa cada um dos grupos criados e C^* representa o conjunto do todos os grupos. De forma análoga, seja $P^* = P_1, P_2, \ldots, P_s$, onde $P_{i=1}^s$ é cada um dos grupos definidos por um usuário com conhecimento do domínio dos dados, sobre a mesma amostra $\mathcal{X}$ e P^* representa o conjunto do todos os grupos; seja $m \neq s$; sejam (x, y) dois elementos da amostra $\mathcal{X}$, de forma que $(x, y) \in X$, é possível obter os seguintes contadores para os pares (x, y):

- α - se os dois elementos pertencem ao `mesmo` grupo em C e ao `mesmo` grupo em P;
- β - se os dois elementos pertencem ao `mesmo` grupo em C e a `diferentes` grupos em P;
- γ - se os dois elementos pertencem a `diferentes` grupos em C e ao `mesmo` grupo em P;
- δ - se os dois elementos pertencem a `diferentes` grupos em C e a `diferentes` grupos em P.

Os valores de $\alpha, \beta, \gamma, \delta$ correspondem ao número de pares em cada situação acima. Então, é possível afirmar que $\alpha + \beta + \gamma + \delta = M$, onde M é o número máximo de pares possíveis em $\mathcal{X}$, i.é, $M = n(n-1)/2$, onde n é o número de elementos em $\mathcal{X}$ definido por $|\mathcal{X}|$.

Os testes estatísticos usados na validação dos grupos formados, como *Estatística de Rand*, *Coeficiente de Jaccard* e *Índice de Folkes & Mallows* são usados para medir a grau de similaridade entre C e P. Considerando os termos definidos acima, os índices citados podem ser descritos como:

Estatística de Rand - Definida por R, onde:

$$R = (\alpha + \delta)/M \tag{2.14}$$

Coeficiente de Jaccard - Definida por J, onde:

$$J = \alpha/(\alpha + \beta + \gamma) \tag{2.15}$$

Índice de Folkes & Mallows - Definida por FM, onde:

$$FM = \alpha/\sqrt{m_1 \cdot m_2} = \sqrt{\frac{\alpha}{\alpha+\beta} \cdot \frac{\alpha}{\alpha+\gamma}}\,, \tag{2.16}$$

onde $m_1 = \alpha/(\alpha + \beta)$ e $m_2 = \alpha/(\alpha + \gamma)$.

Rand e *Jaccard* apresentam valores entre 0 e 1 e são maximizados quando $m = s$. Já o *Índice de Folkes & Mallows* pode variar. Considerando os três índices descritos, conforme apresentado por Halkidi et al. (HALKIDI; BATISTAKIS; VAZIRGIANNIS, 2001), quanto mais altos os valores dos índices, maior é a semelhança entre C e P é maior. E por conseqüência, a qualidade do agrupamento resultante também é maior.

3 MÉTODO SEMI-AUTOMÁTICO DE ESTIMATIVA DA QUALIDADE DE FUNÇÕES DE SIMILARIDADE

Neste capítulo, é apresentado o processo de estimativa de qualidade de funções de similaridade apresentado neste capítulo propõe uma abordagem semi-automática de estimativa de revocação e precisão (R&P) com pouca dependência do especialista humano. O método requer que o usuário informe somente quantos objetos distintos são representados em uma amostra. As consultas são avaliadas através de um processador de consulta que implementa uma ou mais funções de similaridade sobre uma coleção de dados[1].

Estimar os valores de R&P é uma forma de avaliar a qualidade de uma função de similaridade aplicada sobe uma coleção. A partir de um conjunto de limiares previamente estabelecido, o método proposto seleciona o limiar mais apropriado para a função de similaridade aplicada a uma determinada coleção. De maneira geral, pode-se considerar como principais aspectos do método proposto:

- Entrada – como entrada o método recebe uma coleção e uma função de similaridade para ser avaliada;

- Intervenção do especialista humano – a partir de uma amostra extraída da coleção, o especialista humano informa quantos elementos distintos estão presentes na amostra;

- Saída – para cada função de similaridade analisada, o método apresenta duas saídas: (i) uma tabela contendo os limiares com os respectivos valores de R&P estimados; e (ii) os limiares selecionados. De acordo com as características da aplicação é possível definir o grau de qualidade desejada, através de definições como revocação alta, ou precisão alta, ou maximizando ambos. Tendo esta definição, o método permite selecionar um ou mais limiares considerados "ótimos", combinando os valores estimados de R&P.

O método proposto e apresentado neste capítulo tem por objetivo avaliar a qualidade da função de similaridade, através da estimativa de R&P para vários limiares. De acordo com a saída do método, é possível estimar o limiar mais apropriado para uma determinada função de similaridade. As consultas por abrangência dependem do limiar para permitir o processamento adequado da consulta por similaridade.

[1] O termo coleção está sendo usado como sinônimo de base de dados no contexto desta tese.

Caso o tipo de consulta usado fosse por quantidade não precisaria do limiar, por ter a definição direta no número de objetos desejados como respostas.

A seguir, um rápido comentário a respeito do método clássico de estimativa de R&P é apresentado na Seção 3.1. Na Seção 3.2 método é explicado de forma genérica, através de um exemplo, e em seguida, cada passo é detalhado e formalizado na Seção 3.3.

3.1 Considerações sobre o método clássico

Conforme exposto no Capítulo 2, o processo clássico para calcular R&P requer que um especialista humano determine quais objetos são relevantes e quais são irrelevantes. Obviamente esta é uma tarefa trabalhosa, pois é necessária para todas as consultas executadas. Ainda mais que, para se obter uma avaliação da qualidade das funções de similaridade é preciso executar uma série de consultas e obter valores médios. Caso contrário, os valores podem estar totalmente distorcidos.

Relembrando, de forma simplificada, a abordagem tradicional para calcular R&P envolve os passos citados a seguir. Para cada função de similaridade a ser analisada, é necessário:

1. Executar uma consulta processada por uma determinada função de similaridade para todos os elementos da coleção, obtendo um *ranking* para cada elemento;

2. Identificar os elementos relevantes e irrelevantes de cada *ranking*;

3. Calcular os valores de R&P;

4. Definir o limiar mais adequado baseado nos valores de R&P obtidos.

Os passos citados precisam ser executados para cada função de similaridade, de forma exaustiva para o maior número possível de consultas. Da mesma forma, tais passos sugerem um ciclo repetitivo que permite o registro dos valores obtidos de forma exaustiva para várias consultas com diferentes funções de similaridade. Baseando-se no registro desses dados, um ou mais limiares podem ser escolhidos como apropriados para a maioria das consultas, de acordo com a aplicação.

Entretanto, existem algumas dificuldades neste processo para o uso prático com coleções reais. Considerar todos os elementos armazenados para serem processados pela função de similaridade resulta em um grande número de *rankings*. Acrescentando que não é possível ter uma coleção de referência, um especialista humano precisa indicar quais são os elementos relevantes. De forma prática não é viável, pois não se pode esperar que alguém conte os valores corretos e incorretos avaliando todos os *rankings* decorrentes das consultas processadas como elementos da coleção, sem uma incidência significativa de falhas. Por esta razão, é necessário um método que minimize a intervenção do especialista humano.

3.2 Visão geral do método proposto

O método proposto recebe como entrada do especialista humano o número de objeto distintos que aparecem em cada amostra. Esta é a única intervenção necessária do especialista humano. Duas estratégias reforçam a independência entre o método

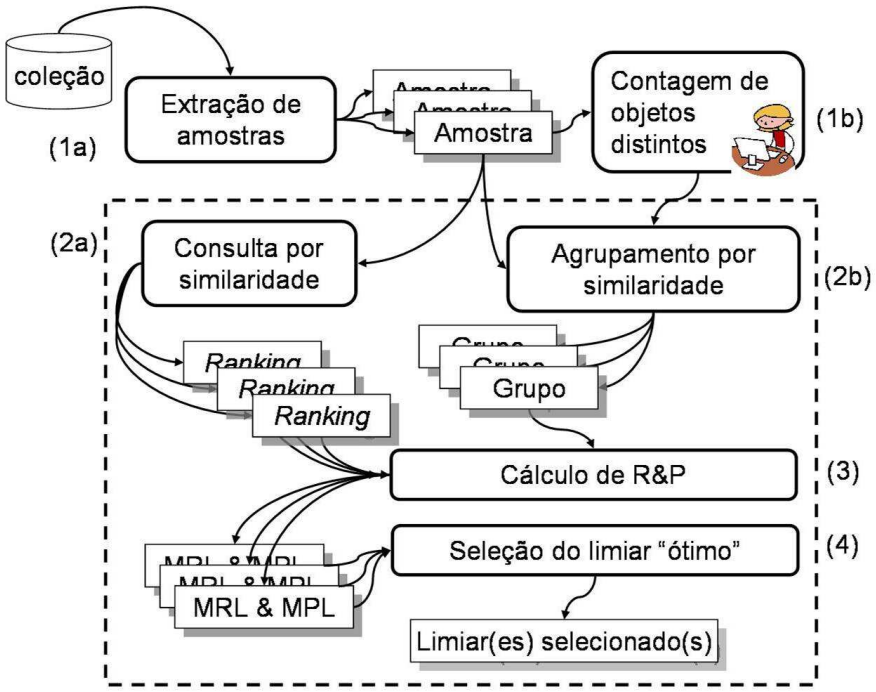

Figura 3.1: Método semi-automático de estimativa de R&P para vários limiares.

de estimativa e o especialista humano e tornam o método semi-automático: (i) o uso de um processo de amostragem, onde as amostras são extraídas da própria coleção; e (ii) a implementação de algoritmos de agrupamento por similaridade, conhecidos como *clustering*, descritos na Seção 2.5. O agrupamento por similaridade, através do uso de funções de similaridade durante a formação dos grupos, é um fator que permite reduzir a intervenção do especialista humano significativamente. Os grupos formados são usados no cálculo automático de R&P, como forma de identificar os elementos relevantes e irrelevantes.

A Figura 3.1 mostra uma representação gráfica dos passos de execução do método semi-automático de estimativa de valores de R&P calculados para vários limiares. Cada atividade (representada por um retângulo com cantos arredondados) gera um ou mais artefatos como resultado (representados por retângulos sombreados). Um amostra pode ser avaliada por mais de uma função de similaridade. Neste caso, os passos dentro da área pontilhada devem ser repetidos para cada função de similaridade avaliada.

A seguir, cada atividade e o correspondente resultado produzido pela atividade é brevemente explicado, de acordo com a seqüência mostrada na Figura 3.1:

- O passo 1 do método mostrado na Figura 3.1 corresponde ao **processo de amostragem**. Uma ou mais `amostras` são geradas a partir de elementos extraídos da coleção (passo 1a). O especialista humano analisa a amostra gerada e informa o `número de objetos distintos` que a amostra contém (passo 1b). Este número será usado como parâmetro para a etapa seguinte, descrita no passo 2b. As amostras geradas são utilizadas no passo 2. A criação das amostras foi derivada dos procedimentos estatísticos tradicionais, resumidos na Seção 3.3.1.

Por exemplo, supondo que foi extraída uma amostra de uma coleção de nomes de cidades, como mostram os dados artificiais apresentados na Figura 3.2. Pode-se observar que esta amostra contém 20 nomes de cidades. O especialista humano conta os objetos diferentes e informa o número de cidades distintas observadas na amostra. Neste caso, 6 cidades. Esse número é usado como parâmetro para o processo de agrupamento por similaridade, descrito no passo 2b.

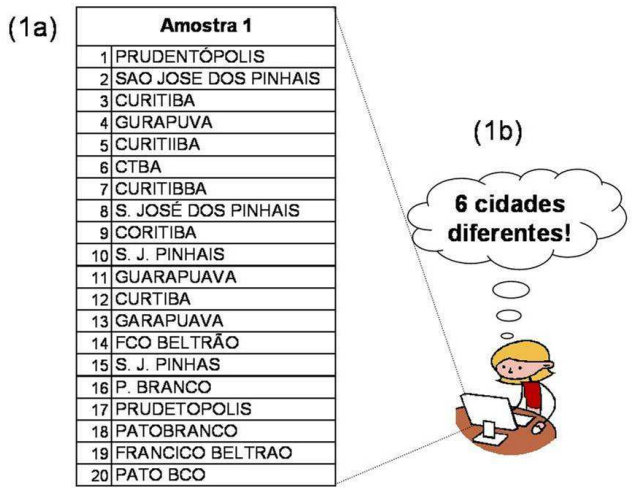

Figura 3.2: Exemplificando os passos (1a) amostra e (1b) contagem de objetos na amostra.

- O passo 2 compreende duas atividades independentes, e que podem ser executadas em paralelo: a consulta (passo 2a) e o agrupamento (passo 2b) por similaridade. Neste passo, utiliza-se a mesma implementação da função de similaridade nas duas atividades:

  - **Consultas por similaridade** – Cada elemento da amostra é utilizado como um objeto consulta sobre os dados da amostra, produzindo o respectivo *ranking* para o objeto consulta através de um mecanismo de consulta por similaridade.

    Seguindo o exemplo do passo anterior, a Figura 3.3 mostra o exemplo de vários *rankings* gerados a partir cada um dos elementos da amostra utilizado como objeto consulta. A primeira cidade, "Prudentópolis" é utilizada como objeto consulta e processada por uma função de similaridade A que resulta em um escore do objeto consulta com cada um dos elementos da amostra. O mesmo acontece com os demais elementos da amostra.

    Portanto, uma amostra gera um número de *rankings* correspondente ao número de elementos que a amostra contém. Cada *ranking* gerado inclui o próprio objeto consulta, com escore igual a 1. Definições e maiores detalhes sobre o mecanismo de consulta por similaridade são apresentados na Seção 3.3.2.

(2a)

Ranking 1

PRUDENTÓPOLIS	
1.00	PRUDENTÓPOLIS
0.90	PRUDETOPOLIS
0.82	PATOBRANCO
0.80	PATO BCO
0.80	P. BRANCO
0.71	SAO JOSE DOS PINHAIS
0.68	S. JOSÉ DOS PINHAIS
0.65	S. J. PINHAS
0.65	S. J. PINHAIS
0.63	GUARAPUAVA
0.61	GARAPUAVA
0.60	GURAPUVA
0.55	CURITIBA
0.55	CURITIIBA
0.53	CTBA
0.52	CURITIBBA
0.45	CORITIBA
0.43	CURTIBA
0.42	FCO BELTRÃO
0.34	FRANCICO BELTRAO

Ranking 2

SAO JOSE DOS PINHAIS	
1.00	SAO JOSE DOS PINHAIS
0.99	S. JOSÉ DOS PINHAIS
0.88	S. J. PINHAIS
0.88	S. J. PINHAS
0.75	PATOBRANCO
0.70	PATO BCO
0.70	P. BRANCO
0.60	PRUDENTÓPOLIS
0.59	PRUDETOPOLIS
0.56	GUARAPUAVA
0.55	GARAPUAVA
0.54	GURAPUVA
0.45	CURITIBA
0.45	CURITIIBA
0.44	CURITIBBA
0.42	CORITIBA
0.43	CURTIBA
0.31	CTBA
0.23	FRANCICO BELTRAO
0.20	FCO BELTRÃO

Ranking 3

CURITIBA	
1.00	CURITIBA
0.92	CURITIIBA
0.92	CURITIBBA
0.88	CURTIBA
0.75	CORITIBA
0.70	CTBA
0.68	FCO BELTRÃO
0.65	FRANCICO BELTRAO
0.62	GUARAPUAVA
0.60	GARAPUAVA
0.55	GURAPUVA
0.54	PATOBRANCO
0.45	PATO BCO
0.45	P. BRANCO
0.44	PRUDENTÓPOLIS
0.42	PRUDETOPOLIS
0.43	SAO JOSE DOS PINHAIS
0.31	S. JOSÉ DOS PINHAIS
0.23	S. J. PINHAIS
0.20	S. J. PINHAS

Ranking ...

Figura 3.3: Exemplificando a criação do passo (2a), onde é gerada a consulta por similaridade com cada elemento da amostra.

– **Agrupamento por similaridade** – O objetivo desta etapa é gerar grupos de elementos de forma que cada grupo contenha somente as representações do mesmo objeto do mundo real. O método utiliza o valor informado pelo especialista humano no passo 1b como critério de quantos grupos devem ser formados. Todos os elementos da amostra são comparados através da função de similaridade que está sendo avaliada, e os elementos com escore mais alto são agrupados até formar o número de grupos necessário de acordo com o método de agrupamento adotado conforme descrito no Capítulo 2. Dessa forma, cada `grupo` concentra os elementos não sobrepostos, i.é, um elemento pertence a um único grupo, o qual contém as variações do mesmo objeto real. Mais detalhes sobre mecanismos de agrupamento por similaridade são apresentados na Seção 3.3.3.

  A Figura 3.4 mostra os 6 grupos gerados da amostra gerada no passo 1a, conforme o número de objetos diferentes informados pelo especialista humano no passo 1b. O número de elementos em cada grupo será utilizado no passo 3, para o cálculo de R&P.

- O passo 3, corresponde ao **cálculo de R&P**. Combina dois resultados produzidos pelo passo anterior: (i) os diversos *rankings* gerados pela atividade de consulta por similaridade (passo 2a), e (ii) os grupos formados pelo algoritmo de agrupamento por similaridade (passo 2b), contendo as variações do mesmo objeto. O número de elementos em cada grupo corresponde ao número de variações do mesmo objeto que devem ser retornados na consulta. Utilizando limiares previamente definidos, é calculado o valor de R&P para cada *ranking* e depois obtida a média aritmética em cada limiar, produzindo de forma automática os MRLs e MPLs, i.é, os valores médios de R&P para cada limiar. Mais detalhes são apresentados na Seção 3.3.4.

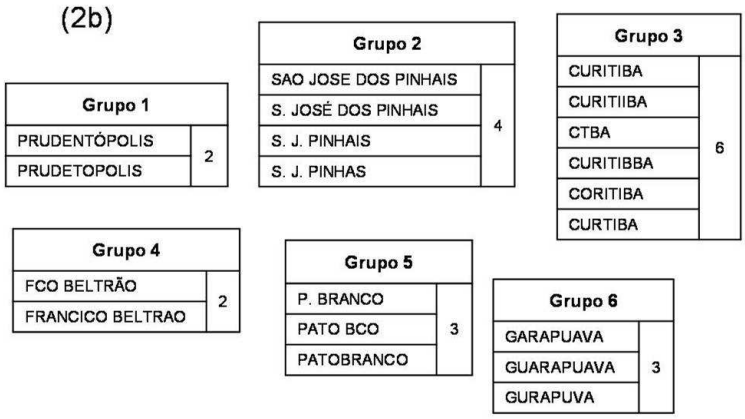

Figura 3.4: Exemplificando o agrupamento por similaridade do passo (2b), onde cada grupo contém as representações do mesmo objeto.

Seguindo com o exemplo, os limiares podem ser definidos como: 0.9, 0.8, 0.7, 0.6, 0.5, 0.4 e 0.3. Podem ser utilizados quaisquer valores, desde que estejam dentro da faixa de valores compreendido pelo escore produzido pela função de similaridade. Em cada *ranking* obtido no passo 2a, são calculados os valores de R&P em cada limiar, utilizando o número de elementos do grupo ao qual o objeto consulta pertence como número de elementos relevantes que deveriam ter retornado. Como exemplificado na Figura 3.5, para cada limiar é calculada a média dos valores de revocação, denominado MRL (Média da Revocação por Limiar) e a média dos valores de precisão, denominado MPL (Média da Precisão por Limiar) considerando todos os *rankings* obtidos da amostra.

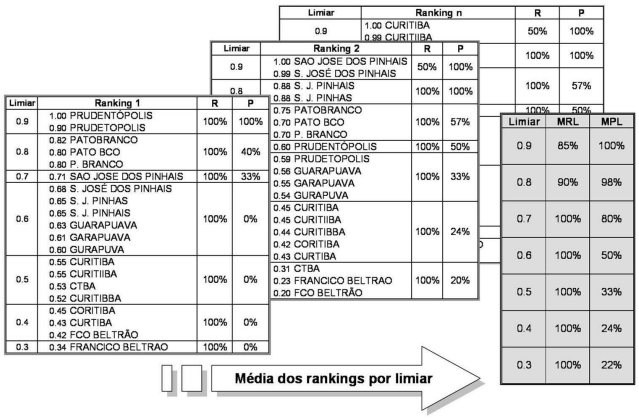

Figura 3.5: Exemplo dos valores médios de R&P estimados para vários limiares.

- O passo 4 refere-se à etapa de **selecionar o limiar "ótimo"** entre os limiares previamente determinados no passo 3. O conceito de limiar `ótimo` depende da aplicação, como exemplificado anteriormente, pois existem situações onde é necessária revocação alta, e outras precisão alta. Dessa forma, o critério de seleção do limiar é dependente da aplicação. Lembrando que, o método já tem como saída os valores estimados para os diversos limiares (MRL e MPL), o

que pode ser mostrado em forma de uma tabela, por exemplo, para determinar qual a qualidade desejada e então selecionar o limiar "ótimo".

Por exemplo, analisando o resultado apresentado na Figura 3.5, caso seja necessário obter alta revocação, o limiar adequado deve ser menor ou igual a 0.7. Esse limiar indica que em média a função de similaridade consegue recuperar todos os elementos relevantes para esta amostra. Por outro lado, se a necessidade da aplicação requer alta precisão, um limiar de 0.9 garante a alta precisão estimada. Uma combinação como a Medida F_1 pode ser usada, combinando as médias referentes aos valores de R&P, conforme explicado no exemplo no início deste capítulo.

No contexto deste trabalho, é adotado um critério cujo objetivo é maximizar ambos, revocação e precisão, como forma de minimizar falsos positivos e falsos negativos no resultado. Maiores detalhes sobre como selecionar os limiares mais apropriados é apresentado na Seção 3.3.5, onde é descrito o algoritmo para selecionar um ou mais limiares visando maximizar tanto a revocação quanto a precisão.

3.3 Definição formal do método proposto

Nesta seção, o método de estimativa de R&P para vários limiares é detalhado e formalizado. Uma vez que o processo foi explicado e exemplificado na Seção 3.2, as sub-seções foram agrupadas de acordo com as principais atividades descritas na Figura 3.1: (i) processo de amostragem; (ii) consulta por similaridade; (iii) agrupamento por similaridade; (iv) cálculo de R&P e (v) seleção do limiar, descritos nas seções a seguir.

3.3.1 Processo de amostragem

Embora o processo de amostragem seja bastante difundido e amplamente estudado na literatura estatística, certos conceitos utilizados no contexto desta tese são brevemente definidos nesta seção como forma de facilitar o entendimento das demais definições utilizadas. As amostras são utilizadas em duas etapas, conforme exposto anteriormente, a consulta e o agrupamento por similaridade. Por esta razão, alguns aspectos a respeito da geração da amostra são considerados e discutidos a seguir.

Primeiramente, é necessário definir a amostra extraída de uma coleção. Um amostra é um subconjunto de uma coleção de dados, composta por um conjunto de elementos atômicos e independentes. Por elemento atômico entende-se que cada elemento representa diretamente uma entidade ou objeto do mundo real.

De maneira mais formal, uma amostra $\mathcal{Q}$ é composta por elementos extraídos de uma coleção V pré-existente, conforme a Definição 1.

***Definição* 1** *(Amostra) Seja uma coleção V, composta de elementos atômicos e representada por $V = \{e_1, e_2, \ldots, e_n\}$, de forma que e_i e e_{i+1} são independentes, sendo $0 < i < n$ e $n = |V|$; seja $\mathcal{Q} \subseteq V$ e $|\mathcal{Q}| \leq |V|$; seja $\mathcal{Q} = \{q_1, q_2, \ldots, q_m\}$, sendo $m = |\mathcal{Q}|$ e $0 < i < m$; $\mathcal{Q}$ é uma* `amostra` *de V produzida por $F_{ga}(V, m)$ de forma que $F_{ga}(V, m) \mapsto \mathcal{Q}$, onde F_{ga} é função conforme a Definição 2.*

***Definição* 2** *(Função de Geração de Amostra) Seja uma coleção V, composta de elementos atômicos w representada por $V = \{w_1, w_2, \ldots, w_n\}$, tal que $n = |V|$ e*

w_i e w_{i+1} são independentes, sendo $0 < i < n$; a **`Função de Geração de Amostra`** *$F_{ga}(V, m)$ é definida por um dos métodos abaixo:*

- ***Aleatório*** *– seja $F_{random}(V) \mapsto w_i$ uma função aleatória de amostragem discreta sobre a coleção V, onde um elemento $w_i \in V$ é escolhido aleatoriamente, com distribuição de probabilidade uniforme definida por $p(w_i) = 1/|V|$ (SPIEGEL, 1992); seja $Q = \bigcup_{i=1}^{m} F_{random}(V)$; então $F_{ga}(V, m) \mapsto Q$.*
- ***Catação*** *– seja $z = |V|/m$, com $m > 0$; $rand(z) \mapsto j$, onde $rand()$ é uma função de geração de números inteiros aleatórios entre 1 e z, de distribuição de probabilidade uniforme(SPIEGEL, 1992); com e $Q = w_j \bigcup_{i=1}^{z-1} w_{j+i*z}$; então $F_{ga}(V, m) \mapsto Q$.*

Uma amostra, conforme apresentada na Definição 1, representa um subconjunto da coleção. Conforme apresentado na literatura sobre estatística (GUERRA; DONAIRE, 1944), pode-se obter uma amostra através de diversos meios. Neste trabalho foram adotadas as duas formas, aleatório e catação, conforme apresentado pela Definição 2. Ambos os métodos permitem uma distribuição representativa da coleção.

3.3.1.1 Geração das amostras

Durante a geração das amostras, ou seja, durante o processo de extração dos elementos para criar as amostras, duas características importantes devem ser consideradas a respeito dos elementos de cada amostra: (i) repetição e (ii) sobreposição.

As coleções reais podem conter elementos idênticos, o que em uma amostragem representa a característica de repetição. Neste trabalho foram utilizadas amostras com e sem repetição. As amostras sem repetição, conforme a Definição 3, descartam os elementos idênticos aos já selecionados na mesma amostra.

Definição 3 *(Amostra Sem Repetição) Seja $\mathcal{Q}$ uma amostra gerada conforme a Definição 1; seja $q_i \in \mathcal{Q}$; uma amostra $\mathcal{Q}$ é dita* **`Amostra Sem Repetição`** *se $(q_i \cap Q) = q_i$.*

De maneira geral, as amostras visam representar o mais fielmente possível os dados que a coleção contém. Permitir repetição de elementos significa que uma mesma amostra pode possuir elementos exatamente iguais. Mantendo-se a mesma proporção de elementos repetidos na amostra e na coleção é possível usar amostras com repetições. Entretanto, é importante salientar que um número elevado de elementos iguais pode distorcer os resultados. Como o método apresentado neste capítulo utiliza a média aritmética dos valores de R&P estimados em cada limiar, o fato de ter elementos iguais influencia de forma positiva ou negativa sobre a média em cada limiar.

A sobreposição permite que o mesmo elemento apareça em mais de uma amostra. Neste trabalho, são utilizadas somente amostras não-sobrepostas, conforme apresentando na Definição 4. Portanto, um elemento da coleção usado em uma amostra, não aparece em outra amostra. Esta estratégia tem a finalidade de garantir maior representatividade das amostras sob dois aspectos. O primeiro refere-se à diversidade de elementos extraídos da coleção, uma vez que sem repetição o número de elementos diferentes é maior. O segundo aspecto refere-se à iteração, utilizando várias amostras sobre a mesma coleção, o processo repetitivo pode se tornar viciado, extraindo o mesmo elemento.

***Definição* 4** *(Amostra Não Sobreposta) Seja $\mathcal{Q}_i$ uma amostra gerada conforme a Definição 1; seja $\mathcal{A} = \bigcup_{i=1}^{j}(Q_i)$ o conjunto das amostras geradas, j é o número de amostras já criadas; $\mathcal{Q}'$ é uma* `Amostra Não Sobreposta`*, se e somente se, $\mathcal{Q}' \cap \mathcal{A} = \emptyset$.*

3.3.1.2 Análise das propriedades da amostra

A amostra é apresentada ao especialista humano para que o mesmo identifique o número de objetos distintos que a mesma possui. Portanto, duas propriedades merecem destaque: o tamanho da amostra e a representatividade.

O tamanho é uma característica a ser observada em função do número de elementos que devem compor a amostra. Uma amostra muito grande dificulta a contagem e identificação do número de objetos diferentes. Uma amostra muito pequena não é significativa e pode não representar claramente as características dos dados armazenados na coleção. Uma amostra que não tenha alto grau de representatividade da coleção implica em estimativas proporcionalmente incorretas.

Para determinar o tamanho adequado da amostra, para que mantenha suas propriedades de representatividade e facilidade de visualização, foram realizados experimentos experimentos variados. Em geral, uma amostra com tamanho entre 30 e 50 elementos é adequada. Os resultados dos experimentos foram analisados de forma estatística para comprovar a representatividade da coleção. Os experimentos estão descritos no Capítulo 5.

3.3.2 Consulta por similaridade

Cada elemento de uma amostra é usado como objeto consulta. Utilizando um mecanismo de processamento desta consulta implementado com uma função de similaridade, o resultado é uma lista dos elementos da amostra, ordenada pelo grau de semelhança com o objeto. Esta lista é conhecida como *ranking*. Portanto, cada elemento da amostra tem seu respectivo *ranking* conforme formalizado pela Definição 5.

***Definição* 5** *(*Ranking*) Seja uma amostra $\mathcal{Q}$ conforme a Definição 1 composta por elementos $q_1, q_2, \ldots, q_n$, onde $n = |\mathcal{Q}|$ e $n > 0$; seja ω um objeto consulta de forma que $\omega \in \mathcal{Q}$; seja f uma função de similaridade de forma que $f(\omega, q_i) \mapsto \varepsilon$, para $1 \leq i \leq n$ e $0 \leq \varepsilon \leq 1$. Um* `Ranking` *$\mathcal{R}_\omega = \sum_{i=1}^{n} < \omega, q_i, \varepsilon_i$ e ε representa o grau de similaridade (escore) entre ω e cada elemento q_i. Um* `Ranking` *$\mathcal{R}_\omega$ possui uma ordenação simples f' de forma que $\mathcal{R}_\omega f'(\varepsilon_i) \geq f(\varepsilon_{i+1})$ para $i = 1, 2, \ldots, n-1$.*

O fato de cada elemento da amostra ser usado como objeto consulta, determina que o número de *rankings* produzidos para cada amostra analisada é igual ao número de elementos da amostra. O objeto consulta faz parte do *ranking*, portanto, em todas as consultas pelo menos um elemento é recuperado com o escore igual a 1 em cada *ranking*.

3.3.3 Agrupamento por similaridade

A estratégia de usar algoritmos de agrupamento por similaridade tem por objetivo criar grupos, cujos componentes de cada grupo sejam representações de um mesmo objeto do mundo real. O critério de agrupamento é determinado por uma função de similaridade. Embora existam diversos tipos de agrupamento, neste

trabalho, optou-se por utilizar a abordagem *bottom-up*, que define a categoria de algoritmos de agrupamento hierárquico aglomerativo (definida na literatura como *hierarchical clustering*). Conforme descrito no Capítulo 2, Seção 2.5, a abordagem *bottom-up* inicia com a construção da `matriz de similaridade` $\mathcal{M}$. O processo de agrupamento dos elementos de $\mathcal{M}$ utilizando uma estrutura em forma de árvore, ou `dendograma`, gera novos grupos, recalculando o valor de similaridade entre os elementos do grupo recém formado. Este processo se repete até que todos os elementos estejam em um único grupo. Então, o critério de corte é aplicado, determinando o número de grupos resultante.

O critério de parada do processo de agrupamento por similaridade adotado neste trabalho é informar como parâmetro o número de grupos que deve ser criado em cada amostra. A razão desta escolha se deve pelo fato de que o valor de similaridade varia em cada grupo, de acordo com os elementos de agrupados. Portanto, definir um critério baseado no grau de similaridade entre os componentes do grupo gerado implica em intervenção humana, analisando cada grupo gerado. Então, optou-se por utilizar o número de objetos diferentes na amostra que foi informado pelo especialista humano (correspondente ao passo (1a) na Figura 3.1) para determinar o número de grupos que deve ser gerado em cada amostra pelo algoritmo de agrupamento por similaridade.

O agrupamento por similaridade, conforme a Definição 7, consiste em aplicar o algoritmo de agrupamento por similaridade sobre cada amostra extraída da coleção, utilizando o número de objetos distintos informado pelo especialista humano como critério de parada do algoritmo. O resultado, conforme a Definição 6 representa o conjunto dos grupos formados pelo algoritmo. Portanto, os elementos da amostra são agrupados, onde cada grupo corresponde às representações do mesmo objeto do mundo real.

Definição 6 *(Agrupamento Resultante) Seja uma amostra $\mathcal{Q}$ conforme a Definição 1 composta por elementos $q_1, q_2, \ldots, q_n$, onde $n = |\mathcal{Q}|$; seja γ um número inteiro positivo $0 \leq \gamma \leq n$; $\mathcal{B}$ é um* ***Agrupamento Resultante*** *definido por $\mathcal{B} = \{C_1, C_2, \ldots, C_\gamma\}$, onde $(C_i \neq \emptyset) \wedge (C_i \subseteq \mathcal{Q}) \wedge C_i = \{c_1, c_2, \ldots, c_w\}$, sendo $1 \leq i \leq \gamma$ e $1 \leq w \leq n$, se e somente se, $\bigcap_{i=1}^{\gamma}(C_i) = \emptyset$. Portanto, $\mathcal{B} = \bigcup_{i=1}^{\gamma}(C_i) \equiv \mathcal{Q}$.*

Definição 7 *(Algoritmo de Agrupamento por Similaridade) Seja uma amostra $\mathcal{Q}$ conforme a Definição 1 composta por elementos $q_1, q_2, \ldots, q_n$, onde $n = |\mathcal{Q}|$ e $n > 0$; seja γ um número inteiro de forma que $1 \leq \gamma \leq n$; um* ***Algoritmo de Agrupamento por Similaridade*** *é uma função definida por $\mathcal{F}_\theta(\gamma) \mapsto \mathcal{B}$, onde $1 \leq \gamma \leq n$ e $\mathcal{B}$ é o* ***Agrupamento Resultante*** *conforme a Definição 6 e $|\mathcal{B}| = \gamma$.*

Conforme apresentado pela Definição 6, o processo de agrupamento por similaridade gera o número de grupos que corresponde ao número de objetos distintos disponíveis na amostra. Cabe à função de similaridade implementada no algoritmo de agrupamento (Definição 7) determinar quais elementos de uma amostra devem pertencer ao mesmo grupo. Cada elemento da amostra é incluído em um único grupo, pois os grupos não são sobrepostos. A união dos elementos contidos nos grupos criados, é equivalente aos elementos da amostra. Portanto, cada grupo contém os elementos da amostra que representam o mesmo objeto do mundo real.

Para facilitar a identificação dos objetos relevantes e irrelevantes para cada consulta, é utilizado um algoritmo de agrupamento que utiliza a mesma função de similaridade que é usada na etapa de consulta, no passo 2a da Figura 3.1. Os elementos que pertencem ao mesmo grupo correspondem às representações do mesmo objeto real. Por esta razão, o número de elementos que o grupo ao qual o objeto consultado pertence possui determina quantos elementos deveriam ser retornados como relevantes no *ranking* da consulta. Dessa forma, sem que o especialista humano necessite contar e indicar quais são os elementos relevantes, é possível calcular os valores de R&P.

Quando a consulta é executada sobre os elementos da amostra, os elementos que pertencem ao mesmo grupo representam o número de elementos que a consulta deveria retornar. Dessa forma é possível utilizar o número de elementos que o grupo ao qual o objeto consultado pertence como informação sobre quantos elementos a função de similaridade deve retornar. Porém, como os grupos são formados de acordo com o critério de similaridade

Embora tenha sido utilizado método de agrupamento hierárquico, outras abordagens de agrupamento podem ser utilizadas. Uma alternativa é o uso de métodos divisivos, como por exemplo k-médias(WIVES, 2004). O valor inicial de k pode ser considerado o valor η informado pelo especialista humano. Dessa forma, o mecanismo de estimativa de limiar se torna flexível, permitindo que diferentes algoritmos de agrupamento possam ser usados, considerando questões de desempenho, volume de dados e característica dos dados analisados. O importante é que cada grupo deve conter as representações do mesmo objeto real, o que é determinado pela função de similaridade utilizada.

A etapa de agrupamento por similaridade assume que: (i) cada grupo gerado pela função de similaridade contém somente representações do mesmo objeto do mundo real e (ii) o número de grupos obtidos para a amostra é aplicado proporcionalmente para a coleção toda. Com a finalidade de verificar tais afirmações, e conseqüentemente corroborar o uso de algoritmos de agrupamento por similaridade neste trabalho, a validação foi feita através de análise estatística dos grupos gerados.

Para validar a grau de correção da formação dos grupos, uma técnica que pode ser utilizada é a Estatística de Teste t-Student(GUERRA; DONAIRE, 1944). O objetivo é medir o grau de confiança em que um determinado elemento está alocado corretamente no grupo em que deveria estar. Outra forma de validação dos grupos formados é através da intervenção do especialista humano. Para verificar que os grupos formados na amostra são representativos para a coleção, é possível comparar os grupos formados em diversas amostras. Representando graficamente através de histograma, espera-se um comportamento semelhante na representação das amostras e na representação da coleção completa. Os experimentos e os respectivos resultados destas análises estão apresentados e discutidos no Capítulo 5.

É importante verificar se a função de similaridade gerou adequadamente os grupos. Caso contrário, o cálculo de R&P produz resultados incorretos, pois o número de elementos de cada grupo é usado para determinar o número de representações do mesmo objeto. Conforme já mencionado, o método semi-automático de estimativa de limiar assume que cada grupo contém somente as variações do mesmo objeto real. Porém, como a qualidade da função de similaridade é o fator que determina a qualidade do agrupamento, podem existir casos que somente um especialista humano pode agrupar corretamente.

3.3.4 Cálculo de R&P

Cada *ranking* obtido pela consulta por similaridade é avaliado em termos de R&P, conforme a Definição 8. Avaliar o *ranking* significa calcular o valor de R&P em cada limiar. Os valores definidos como limiares devem estar dentro da faixa de valores de escore calculados pela função de similaridade. Embora os valores de limiar pré-definidos possam ser escolhidos pelo especialista humano, é comum usar o padrão adotado em sistemas de RI como foi feito neste trabalho. Em sistemas de RI, utilizam-se 11 pontos de revocação, dentro do intervalo $[0.0, 1.0]$, formando o conjunto de limares $L = \{0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0\}$.

***Definição* 8** *(Revocação e precisão) Seja um* ranking $\mathcal{R}_\omega$ *conforme a Definição 5; seja* L *um conjunto de limiares previamente definidos na forma de* $L = \{\tau_1, \tau_2, \ldots, \tau_z\})$*, onde* $z = |L|$ *e* $\tau_i \in \mathbb{R}^+$*; seja* C_ω *tal que* $(\omega \in C)$*, onde* C *é o grupo ao qual* ω *pertence e que corresponde ao número de variações de* ω*, conforme a Definição 6 e* C_ω *corresponde aos* `elementos relevantes` *de* ω*; seja* $Ra = \bigcup_{j=1}^{i}(q_i)$*, onde* $0 \leq j \leq i$*,* Ra *corresponde ao número de* `elementos recuperados` *no* ranking $\mathcal{R}_\omega$ *até a posição* i*. De acordo com Baeza (BAEZA-YATES; RIBEIRO-NETO, 1999), a* `revocação` r *e a* `precisão` p *são calculadas pelas Equações 3.2 e 3.1, respectivamente, para cada limiar* i*. Portanto,* $\forall\tau_i\exists(p_i \wedge r_i)$*, onde* $1 \leq i \leq z$*.*

$$p_i = (C_\omega \cap Ra)/Ra \tag{3.1}$$

$$r_i = (C_\omega \cap Ra)/C_\omega \tag{3.2}$$

Portanto, para cada *ranking* e em cada limiar previamente definido, são calculados os valores de R&P, conforme apresentado na Definição 8. Entretanto, conforme já mencionado, um único ranking não pode ser considerado como representativo da avaliação da qualidade de uma função de similaridade sobre uma amostra. Por esta razão, em cada limiar, são calculados os MRLs e MPLs, i.é, médias da revocação e da precisão por limiar, conforme as Definições 9 e 10, respectivamente. Os valores de MRL e MPL são obtidos em cada limiar, por amostra, de forma a expressar o comportamento da função de similaridade sobre a amostra.

***Definição* 9** *(Média da Revocação por Limiar (MRL)) Seja* $\mathcal{R}$ *o conjunto de* rankings*, onde* $\mathcal{R} = \{R_1, ..., R_x\}$*; seja* $\mathcal{L}$ *o conjunto de limiares previamente definidos, onde* $\mathcal{L} = \{L_1, ..., L_m\}$*; seja* $\mathcal{F}_r(R_j, L_i) \mapsto [0, 1]$ *a função que calcula a revocação em cada* ranking R_j *com limiar* L_i *resultando em* $R_jL_i = \mathcal{F}_r(R_j, L_i)$*; sejam* $i = 1 \leq m$ *e* $j = 1 \leq x$*; então* $\overline{MRL}_i = \sum_{j=1}^{x} R_jL_i$ *é chamado de* `Média da Revocação por Limiar - MRL`*, onde* i *representa cada limiar definido.*

***Definição* 10** *(Média da Precisão por Limiar (MPL)) Seja* $\mathcal{R}$ *o conjunto de* rankings*, onde* $\mathcal{R} = \{R_1, ..., R_x\}$*; seja* $\mathcal{L}$ *o conjunto de limiares previamente definidos, onde* $\mathcal{L} = \{L_1, ..., L_m\}$*; seja* $\mathcal{F}_p(R_j, L_i) \mapsto [0, 1]$ *a função que calcula a precisão em cada* ranking R_j *com limiar* L_i *resultando em* $R_jL_i = \mathcal{F}_p(R_j, L_i)$*; sejam* $i = 1 \leq m$ *e* $j = 1 \leq x$*; então* $\overline{MPL}_i = \sum_{j=1}^{x} R_jL_i$ *é chamado de* `Média da Precisão por Limiar - MPL`*, onde* i *representa cada limiar definido.*

3.3.5 Seleção do limiar "ótimo"

A seleção do limiar é dependente da coleção de dados utilizada e da função de similaridade implementada. Portanto, a etapa de consulta por similaridade assume que os valores de R&P, em média, correspondem aos limiares previamente definidos, conforme apresentado na Definição 8. Selecionar o limiar mais adequado, ou seja, o limiar considerado "ótimo", implica em selecionar um ou mais limiares que resultam em valores máximos de R&P. Um limiar "ótimo" tem por finalidade minimizar falsos positivos e falsos negativos, ou, em outras palavras, o ponto de corte no *ranking* que melhor separe os elementos relevantes dos irrelevantes e dessa forma recupere somente as variações do objeto consultado.

Lembrando que, de maneira geral, R&P são medidas ortogonais, pois a medida que aumenta o valor de revocação diminui o valor de precisão, é necessário uma representação que permita a combinação de R&P. A `Medida` F_1, conforme apresentada no Capítulo 2 é utilizada em sistemas de RI quando se trata de combinar valores de R&P com pesos iguais, embora tais combinações possam variar de acordo com as características da aplicação. Portanto, a Definição 11 corresponde a dizer que um limiar é considerado "`ótimo`" quando F_1 é maximizado.

***Definição* 11** *(Limiar "ótimo") Sejam* $\overline{P_i}$ *e* $\overline{R_i}$ *as* `médias aritméticas dos valores de precisão e revocação`*, respectivamente, para o limiar* i *obtido a partir de um conjunto finito de* rankings $\mathcal{R}^*$*, conforme a Definições 10 e 9; seja* $L = \bigsqcup_{i=1}^{j} \tau_i$ *um conjuntos de limiares conforme definido em Definição 8; onde* $0 \leq i \leq j$*; é possível definir um* `limiar "ótimo"` *por uma função* $max(F_1)$*, onde* F_1 *é obtido pelo maior valor retornado pela Equação 3.3 (BAEZA-YATES; RIBEIRO-NETO, 1999):*

$$F_1(i) = \frac{2}{\frac{1}{R_i} + \frac{1}{P_i}} \tag{3.3}$$

Conforme apresentado pela Definição 11, um limiar considerado ótimo neste trabalho refere-se a melhor combinação de R&P com o objetivo de minimizar falsos positivos e falsos negativos, o que implica na escolha de valores máximos tanto de revocação quanto de precisão. Conforme apresentado no Capítulo 2, na Seção 2.4, a `Medida` F_1 representa o valor combinado de R&P, calculando um único valor para cada limiar. Portanto, uma forma de obter o limiar é escolher o limiar correspondente ao maior valor que resulta do cálculo da `Medida` F_1.

Uma alternativa ao valor combinado de R&P, é seleciona-los separadamente, usando diretamente a média obtida para cada limiar representado por R_j para revocação e P_j para precisão, onde j corresponde ao limiar, conforme descrito na Definição 11. Uma solução para selecionar os limiares utilizando os valores separados de R&P é apresentada no Algoritmo 1, cujas notações utilizadas são descritas como:

- $\langle Th \rangle_v$ - conjunto de z limiares previamente definidos e representados por $L_{j=1}^{z}$, onde L_j corresponde a cada limiar como descrito na Definição 8;
- $\langle S \rangle_r$ - conjunto de valores médios de revocação representados por R, conforme a Definição 9;
- $\langle S \rangle_p$ - conjunto de valores médios de precisão representados por P, conforme a Definição 10;

- $f_{dist}(S_r, S_p)$ - valor obtido pelo cálculo da distância absoluta entre S_r e S_p, obtida por $|S_r - S_p|$, para cada par de valores definidos por $\langle S \rangle_r$ e $\langle S \rangle_p$;
- $\langle S \rangle_{out}$ - conjunto de limiares selecionados $\leq \langle C \rangle_v$;
- C - valor constante atribuído para limitar a distância máxima dos valores de R&P.

Algoritmo 1 ***SelectThreshold***

1: **Input**:$\langle S \rangle_r, \langle S \rangle_p, \langle Th \rangle_v, C$
2: **Output**: $\langle S \rangle_{out}$
3: $w =$ `null`
4: $\langle S \rangle_{temp} = \emptyset$
5: **for** i from 1 to $|\langle Th \rangle_v|$ **do**
6: compute $w = f_{dist}(\langle S \rangle_r, \langle S \rangle_p)$
7: $\langle S \rangle_{temp}[i] = w$
8: **end for**
9: $w =$ `null`
10: **for** i from 1 to $|\langle Th \rangle_v|$ **do**
11: $w =$ `min` from $\langle S \rangle_{temp}$
12: **if** $w \leq C$ **then**
13: $\langle S \rangle_{out} = \langle S \rangle_{out} \bigcup \{w\}$
14: $\langle S \rangle_{temp} = \langle S \rangle_{temp} - \{w\}$
15: **end if**
16: **end for**
17: return $\langle S \rangle_{out}$

O Algoritmo 1 define uma função baseada em divisão na linha 6 para calcular a distância entre os valores médios de R&P para cada limiar previamente definido em $\langle Th \rangle_v$. Embora o cálculo de distância tenha sido definido pela operação de divisão, usando a operação de diferença apresenta o mesmo resultado. Caso seja necessário estabelecer um peso para revocação ou precisão, esta função poderia ser facilmente adaptada, atribuindo um valor constante multiplicado pelo valor da revocação ou da precisão. As linhas 5-7 produzem o conjunto de valores temporários $\langle S \rangle_{temp}$, representando as distância entre os valores médios de R&P em cada limiar.

Tabela 3.1: Exemplo do Algoritmo 1.

$\langle Th \rangle_v$	$\langle S \rangle_r$	$\langle S \rangle_p$	$f_{dist}(S_r, S_p)$
0.9	0.65	1	0.35
0.8	0.82	0.99	0.17
0.7	0.84	0.98	0.14
0.6	0.87	0.85	0.02
0.5	0.93	0.75	0.18
0.3	1	0.59	0.41
	$C = 0.20$	$\langle S \rangle_{out} = 0.8\ 0.7\ 0.6\ 0.5$	
	$C = 0.15$	$\langle S \rangle_{out} = 0.7\ 0.6$	
	$C = 0.10$	$\langle S \rangle_{out} = 0.6$	

A partir das distâncias calculadas, o algoritmo seleciona os valores mínimos, i.é, com menor distância entre R&P, de acordo com um limite pré-fixado por um valor constante C. O uso da constante C como limite de distância, tem por objetivo selecionar mais de um limiar onde os resultados obtidos pelas médias de R&P cujas distâncias são curtas e, portanto, apresentariam os mesmos resultados mesmo variando o limiar. Quanto maior o valor de C, maiores são as chances de ocorrer falsos positivos. A Tabela 3.1 mostra valores artificiais para exemplificar a execução do Algoritmo 1, usando diferentes valores para a constante C.

3.4 Resumo do capítulo

O método apresentado baseia-se na extração de amostras de uma coleção e no uso de algoritmos de agrupamento por similaridade para identificar as variações de cada objeto, devidamente alocados ao mesmo grupo. Cada amostra representa um conjunto de consultas que produzem *rankings* compostos por todos os elementos da amostra. A avaliação de cada *ranking* através de R&P em vários limiares, permite estimar valores médios de R&P mais adequados para determinada função de similaridade e coleção. O método proposto apresenta a estratégia para verificar se os resultados obtidos nas amostras são suficientemente representativos da coleção.

O método semi-automático de estimativa de valores de R&P procura minimizar a interação com o especialista humano, permitindo que a estimativa possa ser aplicada para grandes coleções. Existem duas premissas nas quais o método se baseia: (i) que o algoritmo de agrupamento por similaridade definiu corretamente as representações do mesmo objeto e (ii) que a estimativa feita para a amostra é válida para a coleção completa. A avaliação experimental comprova a viabilidade do método mostrando que tais premissas são verdadeiras com o uso de dados reais.

4 DISCERNIBILIDADE: MEDIDA DE QUALIDADE DE FUNÇÕES DE SIMILARIDADE

Avaliação da qualidade de *rankings* produzidos por uma função de similaridade tradicionalmente é feita em termos de revocação e precisão (R&P). Medidas baseadas em R&P mostram a habilidade da função de similaridade em recuperar elementos relevantes de uma coleção. Ambos medem a qualidade do ponto de vista numérico, pois a revocação representa o número de elementos recuperados, enquanto a precisão representa a relevância dos elementos recuperados para uma determinada consulta. R&P apresentam resultados satisfatórios para avaliar a habilidade da função de similaridade em recuperar elementos mais relevantes à consulta com valor de escore mais alto.

Entretanto, o tipo de consulta considerada neste trabalho, denominada como `consulta vaga por abrangência`, assume que um determinado limiar separa corretamente elementos relevantes à consulta dos irrelevantes. É importante lembrar que por relevante, consideram-se somente as variações do mesmo objeto consultado. Por esse ponto de vista, uma função ideal deveria atribuir um grau máximo (totalmente relevante) aos elementos que representam o objeto consultado e um grau mínimo (totalmente irrelevante) para os elementos que não são variações do mesmo objeto consulta.

Este capítulo introduz uma nova métrica, chamada discernibilidade, cujo objetivo é medir o resultado de uma função de similaridade quanto à capacidade de separar os elementos relevantes dos irrelevantes. Com a medida de discernibilidade é possível diferenciar funções de similaridade mais apropriadas para determinadas coleções através de propriedades que não são perceptíveis com o uso de R&P.

Este capítulo inicia com a explicação da diferença entre a avaliação da qualidade do resultado produzido por funções de similaridade em termos de R&P e de discernibilidade. Em seguida, é apresentado o método de definição da discernibilidade, através das duas etapas (i) definição do limiar e (ii) cálculo da discernibilidade. O processo de definição do limar pode ser feito através de duas abordagens distintas, uma algorítmica e outra estatística. O cálculo da discernibilidade formaliza a métrica definida, utilizado o limiar obtido na primeira etapa. Em seguida, é apresentada uma comparação da discernibilidade com a precisão média, através de um exemplo onde as duas medidas são utilizadas para avaliar um *ranking* obtido por uma função de similaridade.

4.1 Avaliação da qualidade de funções de similaridade por discernibilidade

O uso de funções de similaridade implica em dois problemas. O primeiro consiste em determinar qual o limiar (*threshold*) mais adequado para separar os elementos relevantes dos irrelevantes. Isto ocorre porque os valores de escore produzidos por uma função de similaridade podem ser muito diferentes daqueles produzidos por outra função, mesmo considerando-se o mesmo domínio. Da mesma forma, utilizando-se conjuntos de dados diferentes com as mesmas funções de similaridade, o limiar pode variar significativamente.

O segundo problema consiste em definir uma medida do quanto uma função de similaridade é mais adequada para um determinado conjunto de dados que outra. Uma forma de avaliação da qualidade das funções de similaridade adotada por Cohen (COHEN; RAVIKUMAR; FIENBERG, 2003) e Bilenko (BILENKO et al., 2003) é baseada na abordagem clássica de RI, conhecida como Curva de Revocação & Precisão (R&P) (BAEZA-YATES; RIBEIRO-NETO, 1999; SALTON, 1989). Curvas de R&P expressam a habilidade da função de similaridade de ordenar por escore os resultados obtidos. Contudo, revocação e precisão são medidas que avaliam a capacidade de recuperação de elementos relevantes, não considerando a variação do escore atribuído aos elementos para definir o *ranking*. Por outro lado, o valor escore pode ser uma propriedade importante para medir a qualidade da função de similaridade, já que através do mesmo, é possível avaliar a capacidade da função de similaridade em separar os elementos relevantes dos irrelevantes.

Investigando vários *rankings* obtidos por consultas usando funções de similaridade, observou-se que certas funções apresentam uma separação distinta dos elementos relevantes e irrelevantes, enquanto que outras funções, embora projetadas para o mesmo domínio apresentam valores relevantes e irrelevantes misturados no *ranking*. Lembrando que um elemento relevante é aquele que é aceito pelo especialista humano como uma variação da representação do mesmo objeto consulta, independente do escore atribuído. Portanto, uma função de similaridade é mais adequada que outra quando consegue atribuir um escore alto para os elementos relevantes e um escore baixo para os elementos irrelevantes.

Por exemplo, é possível considerar duas funções projetadas para o mesmo domínio. Considera-se que a função de similaridade A, produz um resultado como apresentado na Tabela 4.1. Na primeira coluna é mostrado o escore obtido, na segunda o elemento correspondente e na terceira coluna, um especialista humano definiu se cada elemento obtido é relevante ou não. De forma parecida, nas Tabelas 4.2 e 4.3, é apresentado o resultado obtido por outras funções de similaridade B e C, respectivamente. Considerando que todos os elementos da coleção foram retornados para a mesma consulta usando as três funções de similaridade A, B e C, analisando a qualidade dos resultados apresentados nas respectivas tabelas, é possível concluir:

- A função de similaridade A é considerada mais adequada, pois usando um limiar com valor maior que 0.3312 e menor que 0.7720 somente os valores relevantes são retornados. Também pode-se observar que nenhum elemento relevante deixaria de ser retornado com a referida faixa de valores. Pode-se dizer que a função A tem maior capacidade de separar os valores relevantes dos irrelevantes, o que permite maior variação do limiar.

- Já a função de similaridade B, embora tenha ordenado primeiro todos os elementos relevantes, apresenta um intervalo de limiar entre o último elemento relevante e o maior irrelevante mais restrito. É intuitivo que a função de similaridade B, embora apresente um grau de discernibilidade, com certeza esse grau de discernibilidade é menor comparado a função de similaridade A.
- Como a função de similaridade C não atribui um escore significativo para elementos relevantes e irrelevantes, pode-se verificar no resultado apresentado na Tabela 4.3 que elementos considerados por especialista humano como irrelevantes apresentam um escore mais alto que outros elementos definidos como irrelevantes. Como a função de similaridade C não distingue adequadamente os relevantes dos irrelevantes, é dito que esta função não é adequada.

Tabela 4.1: Resultado obtido de acordo com a função de similaridade A.

Escore	**Elemento**	**Relevância**
1.0000	Ranking in Databases	Relevante
0.9581	Ranking on Databases	Relevante
0.8753	Ranking on DBs	Relevante
0.8391	Rankin on DBs	Relevante
0.7720	Ranking and DBs	Relevante
0.3312	Relational Databases	Irrelevante
0.3040	Ranking on IR	Irrelevante
0.2871	Ranking Correlation	Irrelevante

Tabela 4.2: Resultado obtido de acordo com a função de similaridade B.

Escore	**Elemento**	**Relevância**
1.0000	Ranking in Databases	Relevante
0.9873	Ranking on Databases	Relevante
0.9040	Ranking on DBs	Relevante
0.8755	Rankin on DBs	Relevante
0.7341	Ranking and DBs	Relevante
0.7221	Relational Databases	Irrelevante
0.7044	Ranking on IR	Irrelevante
0.7025	Ranking Correlation	Irrelevante

Portanto, com o intuito de quantificar a capacidade de uma função de similaridade em discernir elementos relevantes dos irrelevantes, foi definida uma medida de "discernibilidade". No contexto deste trabalho, é possível verificar que quanto maior for a discernibilidade de uma função de similaridade, maior é a qualidade do *ranking* obtido. Relembrando que por qualidade, busca-se minimizar falsos positivos e falsos negativos.

A avaliação da qualidade de uma função de similaridade através da discernibilidade proposta neste trabalho é realizada em duas etapas:

1. Definir o limiar adequado para separar os elementos relevantes dos irrelevantes. Pode ser um valor único ou um intervalo de limiares, compreendido pelo menor escore do elemento relevante e pelo maior escore do elemento irrelevante;

Tabela 4.3: Resultado obtido de acordo com a função de similaridade C.

Escore	Elemento	Relevância
1.0000	Ranking in Databases	Relevante
0.9581	Ranking on Databases	Relevante
0.7023	Relational Databases	Irrelevante
0.6789	Ranking Correlation	Irrelevante
0.6767	Ranking on DBs	Relevante
0.6089	Rankin on DBs	Relevante
0.5543	Ranking and DBs	Relevante
0.4412	Ranking on IR	Irrelevante

2. Calcular a discernibilidade. Essa medida estabelece um valor numérico para mensurar a qualidade de uma função de similaridade quanto à habilidade em separar elementos relevantes dos irrelevantes.

4.2 Processo de definição do limiar

Para muitas aplicações, o processo de definição do limiar é atribuído ao especialista humano, o qual determina um valor de corte arbitrário para uma ou mais consultas. Considerando que o limiar refere-se ao escore que determina o grau de similaridade entre o objeto consultado e os dados armazenados, a definição de um limiar muito alto implica no risco de não recuperar nenhum resultado. Por outro lado, utilizando um valor muito baixo pode permitir que um número excessivo de elementos irrelevantes sejam recuperados.

O método de definição do limiar apresentado neste trabalho utiliza um processo de amostragem, onde cada amostra $\mathcal{Q}$ é composta por elementos de uma coleção V pré-existente, semelhante a Definição 1. Cada elemento da amostra q é usado como objeto consulta sobre $\mathcal{Q}$ e o resultado produzido por uma função de similaridade L é composto por cada um dos elementos q da amostra e o respectivo valor de escore s_q, o qual determina o grau de similaridade entre o objeto consultado e os elementos da amostra.

Relembrando, uma amostra $\mathcal{Q}$ é definida a partir da coleção V, i.e., $\mathcal{Q} \subseteq V$. Cada elemento q da amostra, i.e., é usado como objeto consulta sobre $\mathcal{Q}$. De forma semelhante ao processo explicado no passo 2a da Figura 3.1 apresentada na Seção 3.2, para cada elemento da amostra é gerado um *ranking*, conforme a Definição 5, pois cada elemento da amostra é utilizado como objeto consulta.

Em cada *ranking* $\mathcal{R}_L(q)$ gerado para cada elemento da amostra q, um especialista humano identifica cada elemento v do *ranking* como relevante (rel), se v refere-se ao mesmo objeto do mundo real representado por q ou como irrelevante (irrel), caso contrário.

***Definição* 12** *(s_{rel} e s_{irrel}) Seja $\mathcal{R}_L(q)$ um* ranking *gerado por uma função de similaridade L conforme a Definição 5; seja v elemento de $\mathcal{R}_L(q)$ e v_s o respectivo escore; seja a Equação 4.1:*

$$\begin{aligned} s_{rel}(q) &= \min\{v_s \mid v \text{ é } \textit{relevante} \}, \\ s_{irrel}(q) &= \max\{v_s \mid v \text{ é } \textit{irrelevante} \}, \end{aligned} \tag{4.1}$$

, *então* s_{rel} *é o* ***menor escore do elemento relevante;*** s_{irrel} *é o* ***maior escore do elemento irrelevante.***

Portanto, a Definição 12 apresenta dois valores para uma determinada consulta k usando uma função de similaridade L: (i) $s^L_{rel}(k)$, que é o menor escore de um elemento relevante e (ii) $s^L_{irrel}(k)$, que corresponde ao maior escore associado a um elemento irrelevante. Este dois valores são utilizados pelos dois métodos de definição do limiar apresentados neste capítulo. É importante notar que para algumas consultas $s^L_{irrel}(k)$ pode ser maior que $s^L_{rel}(k)$. Tal situação indica que a função de similaridade falhou na separação de elementos relevantes e irrelevantes.

Exemplo: Considerando uma base de dados que contém títulos de periódicos em ciência da computação. O objeto "Journal of Informetrics" é representado em cinco formas diferentes: "Journal of Informetrics", "J. of Informetrics", "JOI", "Informetrics Journal", "Jrnl of Infometrics". Supondo que essa base de dados possui nove títulos, o resultado produzido pela função de similaridade *edit distance* é mostrado na Tabela 4.4. Analisando o resultado, o menor escore de um elemento relevante é $s_{rel} = 0.1304$ e o maior escore de um elemento irrelevante é $s_{irrel} = 0.1250$. Então, a discernibilidade é calculada com base no intervalo formado por s_{rel} e s_{irrel}, o qual separa elementos relevantes dos irrelevantes, respectivamente.

Tabela 4.4: Exemplo do resultado de uma função de similaridade.

Escore	**Elemento**	**Relevância**
1.0000	Journal of Informetrics	Relevante
0.8636	Jrnl of Infometrics	Relevante
0.7391	J. of Informetrics	Relevante
0.1304	Informetrics Journal	Relevante
0.1304	JOI	Relevante
0.1250	Decision Support Systems	Irrelevante
0.0869	TODS	Irrelevante
0.0869	SIGMOD	Irrelevante
0.0434	TKDE	Irrelevante

Dois métodos semi-automáticos para o cálculo do limiar de uma determinada função de similaridade são propostos no contexto desta tese. O primeiro deles, apresentado na Seção 4.2.1 é algorítmico e o segundo, apresentado na Seção 4.2.2 é estatístico. Ambos os métodos baseiam-se nos valores de s_{rel} e s_{irrel} como entrada. Os valores de s_{rel} e s_{irrel} podem apresentar duas situações como: (i) s_{rel} é maior que s_{irrel}, que é situação ideal, onde a função de similaridade separou corretamente relevantes dos irrelevantes; ou (ii) s_{rel} é menor s_{irrel}, ocorre o inverso, um elemento irrelevante teve um escore maior que o elemento relevante. Observando os valores de s_{rel} e s_{irrel} obtém-se um intervalo na forma de $[s_{rel}, s_{irrel}]$.

Por exemplo, considerando uma série de consultas onde são marcados os pontos de s_{rel} e s_{irrel} de cada *ranking*, como a representação gráfica na Figura 4.1. Conectando-se os pontos de s_{rel} e s_{irrel} em duas linhas, respectivamente. É possível observar que em certos pontos as linhas estão distantes, em outros estão próximas, e em outros podem estar cruzadas. Esta ilustração tem por objetivo mostrar que quanto maior a distância entre as duas linhas, melhor é a discernibilidade da função. Quanto mais estiverem próximas, a qualidade piora e se as linhas de cruzam, significa que a função não conseguiu identificar corretamente os elementos relevantes e separa-los dos irrelevantes, atribuindo escores inconsistentes.

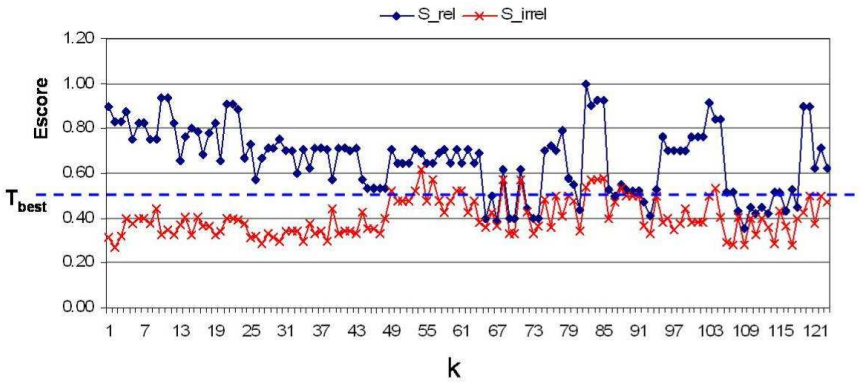

Figura 4.1: Exemplo ilustrativo dos valores de s_{rel} e s_{irrel}.

A linha tracejada representada por t_{best} na Figura 4.1 mostra uma separação ilustrativa do limiar. Como saída, ambos os métodos produzem um intervalo denominado $[t_{best}^{\min}, t_{best}^{\max}]$, os quais representam valores de escore que visam limiares "ótimos". Por "ótimo" é entendido um valor de limiar que maximiza o número de casos onde $s_{irrel} \leq t_{best} \leq s_{rel}$. O processo de definição do limiar representado pelo intervalo $[t_{best}^{\min}, t_{best}^{\max}]$ é considerado adequado quando se encontra dentro dos limites definidos pelo especialista humano. Lembrando que tais limites são representados por s_{rel} e s_{irrel}, conforme a Definição12. As duas premissas que guiam o processo de definição do limiar em ambos os métodos são: (i) minimizar falsos positivos e (ii) minimizar falsos negativos.

4.2.1 Abordagem baseada em uma função de recompensa

O primeiro método visa medir o quanto um limiar é capaz de separar relevantes de irrelevantes. Foi definida uma função baseada em um critério de recompensa, que permite uma implementação descrita pelo Algoritmo 2. De maneira geral, o Algoritmo 2, incrementa (ou decrementa) pontos se o escore obtido é acima (ou abaixo) dos valores identificados pelo especialista humano como s_{rel} e s_{irrel}.

A função baseada nessa idéia de recompensa pode ser definida pela fórmula abaixo:

$$f^L(n,t) = \sum_{k=1}^{n} d(s_{rel}^L(k), s_{irrel}^L(k)) \tag{4.2}$$

onde:

L é a função de similaridade usada;

n é o número de objetos usados para consulta (tamanho da amostra);

t é o limiar que está sendo analisado;

$d(\cdot,\cdot)$ mede o quanto $s_{rel}^L(k)$ e $s_{irrel}^L(k)$ são adequados com o limiar t, de forma que:

$$d(s_{rel}^L(k), s_{irrel}^L(k)) = R_{rel}^t(k) + R_{irrel}^t(k) \tag{4.3}$$

com

$$R^t_{rel}(k) = \begin{cases} 1 & \text{se} \quad s^L_{rel}(k) > t \\ -1 & \text{senão} \quad s^L_{rel}(k) \leq t \end{cases} \text{ e } R^t_{irrel}(k) = \begin{cases} -1 & \text{se} \quad s^L_{irrel}(k) \geq t \\ 1 & \text{senão} \quad s^L_{irrel}(k) < t \end{cases} \tag{4.4}$$

De acordo com as equações acima, o limiar "ótimo" t_{best} (ou mais precisamente o intervalo para o limiar "ótimo") é aquele que alcança o valor máximo na função $f^L(n,t)$ e pode ser definido como:

$$f^L_{\max} = \max_{t \in [t_{\min}, t_{\max}]} \left\{ f^L(n,t) \right\}, \tag{4.5}$$

onde t_{min} e t_{max} representa os limites do intervalo a ser testado.

O Algoritmo 2 mostra a descrição do *BestThresh*, o qual determina t_{best}. As entradas para o algoritmo são: (i) o número de consultas(n), (ii) os limites do intervalo para ser testado(t_{min} e t_{max}), (iii) o menor escore de similaridade obtido por um elemento relevante para a consulta k, denotado por $s^L_{rel}(k)$, (iv) o maior escore obtido por um elemento irrelevante para a mesma consulta k, denotado por $s^L_{irrel}(k)$, e (v) a precisão numérica (h) sobre a qual o algoritmo deve operar. O algoritmo produz duas saídas: o intervalo $[t^{\min}_{best}, t^{\max}_{best}]$ no qual o limiar ótimo (t_{best}) se encontra; e seu respectivo $f_{\max}$, que é o número de pontos obtidos pelo intervalo. A razão de manter a saída do algoritmo como um intervalo ao invés de um valor único é que mais de um valor de limiar, em ordem seqüencial, pode atingir $f_{\max}$. Portanto, o menor e o maior valor de limiar são utilizados como limites do intervalo. Como utilizar $f_{\max}$ na avaliação da qualidade da função de similaridade é discutido na Seção 5.4.4.

Os limites $t_{\min}$ e $t_{\max}$ são, respectivamente, o menor e o maior escore de similaridade do resultado gerado pela função de similaridade. A precisão numérica denotada por h é calculada pela fórmula $h = (t_{\max} - t_{\min})/n_{div}$, onde n_{div} é o número de divisões necessárias para definir o intervalo $[t_{min}, t_{max}]$. Dessa forma, cada limiar t para ser testado pelo algoritmo é obtido por $t_i = t_{\min} + ih$, onde $i = 0, ..., n_{div}$.

O algoritmo trabalha da seguinte forma: cada limiar t, de acordo com a precisão numérica definida entre t_{min} e t_{max}, é testado para cada consulta. O teste consiste em comparar t com s_{rel} e s_{irrel}. O número de pontos obtido por cada limiar t é calculado de acordo com as Equações 4.4 e 4.5. O limiar t é inicializado com o menor valor possível no início do algoritmo. O maior número de pontos obtido por um limiar ($f_{\max}$) é então encontrado. Uma vez que $f_{\max}$ é estabilizado, o Algoritmo 2 encontra o intervalo no qual todos os valores obtém $f_{\max}$.

4.2.2 Abordagem baseada em análise estatística

Uma alternativa ao método baseado na função de recompensa definida na Seção 4.2.1 é a abordagem estatística utilizando uma distribuição normal bivariada (SPIEGEL, 1992)[1] para determinar t_{best}. O método estatístico procura maximizar a probabilidade de encontrar um limiar que minimize falsos positivos e falsos negativos.

A intuição por trás dessa abordagem encontra-se no ajuste de duas curvas, que representam a distribuição da freqüência dos valores de s_{irrel} e s_{rel} resultantes de

[1] Weisstein,E.W.(2004) Bivariate Normal Distribution, From MathWorld - A Wolfram Web Resource, disponível em: http://mathworld.wolfram.com/BivariateNormalDistribution.html

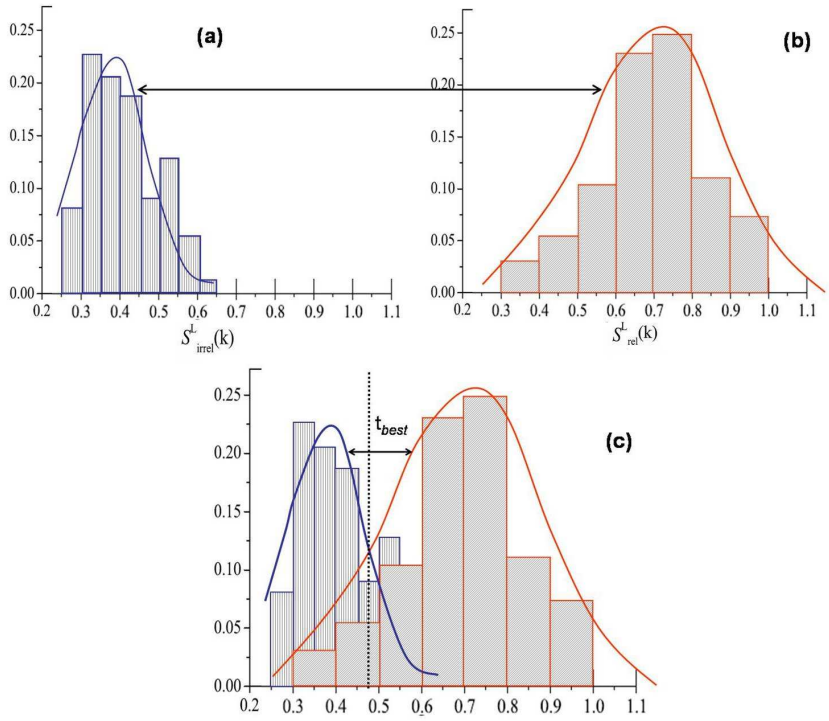

Figura 4.2: Exemplo ilustrativo para determinar t_{best} usando a distribuição bivariada.

uma amostra, como ilustrado com valores artificiais na Figura 4.2. De forma simplificada, pode-se dizer que para determinar o limiar, ocorre um ajuste entre as curvas resultantes dos valores de s_{irrel} e s_{rel} como mostra, respectivamente, os gráficos (a) e (b) da Figura 4.2. O ajuste é feito pelo cálculo dos valores médios entre as duas curvas, até que se obtenha a melhor combinação entre as duas curvas, definindo o limiar com maior probabilidade de minimizar falsos positivos e falsos negativos. O limiar é representado por t_{best} no gráfico (c) da Figura 4.2, como resultado obtido pela Equação 4.13, a qual é baseada na distribuição da probabilidade calculada pela Equação 4.10, ambas descritas mais adiante nesta seção.

Formalmente, a probabilidade é determinada pela função de densidade de probabilidade (*probability density function* - PDF) para s^L_{irrel} e s^L_{rel}, sendo denotada por $P(s^L_{irrel})$ e $P(s^L_{rel})$, respectivamente. Considerando uma amostra de tamanho n, os valores médios experimentais são calculados como:

$$\left\langle s^L_{rel} \right\rangle = (1/n) \sum_{k=1}^{n} s^L_{rel}(k), \tag{4.6}$$

$$\left\langle s^L_{irrel} \right\rangle = (1/n) \sum_{k=1}^{n} s^L_{irrel}(k) \text{ e} \tag{4.7}$$

e o respectivo desvio padrão:

$$\sigma(s^L_{rel}) = \sqrt{[1/(n-1)] \sum_{k=1}^{n} \left[s^L_{rel}(k) - \left\langle s^L_{rel} \right\rangle\right]^2}. \tag{4.8}$$

$$\sigma(s^L_{irrel}) = \sqrt{[1/(n-1)] \sum_{k=1}^{n} \left[s^L_{irrel}(k) - \left\langle s^L_{irrel} \right\rangle\right]^2}, \tag{4.9}$$

Sejam as Equações 4.6 e 4.7 as médias e as Equações 4.8 e 4.9 para o desvio padrão, respectivamente, para s_{rel} e s_{irrel}, é calculada a distribuição $P(s_{rel})$ e $P(s_{irrel})$, definida pela Equação 4.10.

$$\begin{aligned} P(s^L_{rel}) &= \frac{1}{\sqrt{2\pi\sigma^2(s^L_{irrel})}} \exp\left[-\frac{1}{2}\left(\frac{s^L_{irrel}-\langle s^L_{rel}\rangle}{\sigma(s^L_{irrel})}\right)^2\right] \\ P(s^L_{irrel}) &= \frac{1}{\sqrt{2\pi\sigma^2(s^L_{rel})}} \exp\left[-\frac{1}{2}\left(\frac{s^L_{rel}-\langle s^L_{irrel}\rangle}{\sigma(s^L_{rel})}\right)^2\right] \end{aligned} \tag{4.10}$$

Uma vez que ocorre uma correlação entre $s^L_{irrel}(k)$ e $s^L_{rel}(k)$, a distribuição normal $P(s^L_{irrel}, s^L_{rel})$ não é necessariamente o produto $P(s^L_{irrel}) \cdot P(s^L_{rel})$. Contudo, para calcular a PDF normal, é necessário considerar o coeficiente de correlação ρ. A fórmula para PDF normal é definida pela Equação 4.11.

$$\begin{aligned} P(s^L_{irrel}, s^L_{rel}) = \frac{1}{2\pi\sigma(s^L_{rel})\sigma(s^L_{irrel})\sqrt{1-\rho^2}} \exp\Bigg\{-\frac{1}{2(1-\rho^2)}\Bigg[\left(\frac{s^L_{irrel}-\langle s^L_{irrel}\rangle}{\sigma(s^L_{irrel})}\right)^2 + \left(\frac{s^L_{rel}-\langle s^L_{rel}\rangle}{\sigma(s^L_{rel})}\right)^2 \\ -2\rho\left(\frac{s^L_{irrel}-\langle s^L_{irrel}\rangle}{\sigma(s^L_{irrel})}\right)\left(\frac{s^L_{rel}-\langle s^L_{rel}\rangle}{\sigma(s^L_{rel})}\right)\Bigg]\Bigg\}, \end{aligned} \tag{4.11}$$

onde ρ é o coeficiente de correlação definido pela fórmula:

$$\rho = \frac{\langle s^L_{rel} s^L_{irrel}\rangle - \langle s^L_{rel}\rangle \langle s^L_{irrel}\rangle}{\sigma(s^L_{rel})\sigma(s^L_{irrel})}, \tag{4.12}$$

o qual assume valores no intervalo $[-1, 1]$, onde $|\rho| \sim 1$ denota alto correlacionamento e $|\rho| \sim 0$ denota falta de correlacionamento de s^L_{rel} e s^L_{irrel}; no caso da curva gaussiana s^L_{rel} e s^L_{irrel} são variáveis aleatórias independentes.

É desejável que a probabilidade do limiar (denotado por $F(t)$) tenha uma aproximação de uma distribuição normal, uma vez que a curva contínua é o ajuste de uma PDF normal.

Para determinar t_{best}, é suficiente encontrar o valor de t que é produzido por:

$$\begin{aligned} F(t) &= P(s^L_{irrel} < t, s^L_{rel} > t) \\ &= \frac{1}{2\pi\sigma(s^L_{rel})\sigma(s^L_{irrel})\sqrt{1-\rho^2}} \int_{-\infty}^{t}\int_{t}^{\infty} \exp\Bigg\{-\frac{1}{2(1-\rho^2)}\Bigg[\left(\frac{s^L_{irrel}-\langle s^L_{irrel}\rangle}{\sigma(s^L_{irrel})}\right)^2 + \left(\frac{s^L_{rel}-\langle s^L_{rel}\rangle}{\sigma(s^L_{rel})}\right)^2 \\ &\quad -2\rho\left(\frac{s^L_{irrel}-\langle s^L_{irrel}\rangle}{\sigma(s^L_{irrel})}\right)\left(\frac{s^L_{rel}-\langle s^L_{rel}\rangle}{\sigma(s^L_{rel})}\right)\Bigg]\Bigg\} ds^L_{irrel} ds^L_{rel} \end{aligned} \tag{4.13}$$

Dessa forma, é possível encontrar o valor de t que maximiza a probabilidade de t ser simultaneamente maior que s^L_{irrel} e menor que s^L_{rel}. Um algoritmo simples foi implementado para calcular $F(t)$, descobrindo o valor de t_{best} para cada função de similaridade usada nos experimentos.

4.3 Cálculo da discernibilidade

Os dois métodos descritos na seção anterior definem um limiar adequado, o qual representa o intervalo formado pelo menor escore do elemento relevante e o maior escore do elemento irrelevante. Esses dois valores são utilizados para calcular a discernibilidade.

Conforme já descrito, a discernibilidade refere-se à habilidade de uma função de similaridade em discernir elementos relevantes dos irrelevantes. A propriedade de discernibilidade considera dois aspectos: (i) o quanto a função de similaridade separa elementos relevantes dos irrelevantes; e (ii) quão distantes estão os elementos relevantes dos irrelevantes, baseando-se no escore dos elementos resultantes para uma determinada consulta. O primeiro aspecto é definido pelo número máximo de pontos (f_{max}) obtidos pelo algoritmo ***BestThresh***. O segundo aspecto é calculado considerando a diferença entre $t_{best}^{\max}$ e $t_{best}^{\min}$. A discernibilidade também define dois coeficientes c_1 e c_2 que permitem ao especialista humano expressar a importância dada para cada um dos dois aspectos considerados. Para os experimentos descritos na Seção 5.4.4 foi considerada a mesma importância para c_1 e c_2 usando-se $c_1 = c_2 = 1$. Os valores produzidos pela função de discernibilidade estão dentro do intervalo [-1,1].

***Definição* 13** *(Discernibilidade) Seja $\mathcal{R}_L$ um* ranking *gerado por uma função de similaridade L conforme a Definição 5; sejam s_{rel} e s_{irrel} valores de $\mathcal{R}_L$ definidos pela Equação 4.1; então* `Discernibilidade` *$\mathcal{D}$ é dada pela equação:*

$$discernibilidade^L(t_{best}^{\min}, t_{best}^{\max}, f_{\max}) = \frac{c_1}{c1+c2}(t_{best}^{\max} - t_{best}^{\min}) + \frac{c_2}{c1+c2} \cdot \frac{f_{\max}}{2n} \quad (4.14)$$

Para avaliar se os valores do intervalo de limiar calculado pelos métodos apresentados na Seção 4.2 são plausíveis, são definidas duas medidas para calcular o intervalo de confiança, considerando a distribuição dos valores do limiar. Neste caso o valor médio de t é dado por

$$\langle t \rangle = \frac{\sum_{i=1}^{n} t_i F(t_i)}{\sum_{i=1}^{n} F(t_i)} \quad (4.15)$$

e o respectivo grau de incerteza associado

$$\sigma_t = \sqrt{\frac{\sum_{i=1}^{n} t_i^2 F(t_i)}{\sum_{i=1}^{n} F(t_i)} - \langle t \rangle^2} \quad (4.16)$$

A discernibilidade aumenta devido a dois fatores: (i) a função de similaridade separa corretamente os elementos relevantes no *ranking* com escore maior que os irrelevantes; e (ii) existe uma diferença considerável entre menor escore do elemento relevante e o maior escore do irrelevante. Como já dito, uma função de similaridade ideal deveria atribuir 1 para os elementos relevantes (similar) e 0 para os elementos irrelevantes (totalmente desigual).

4.4 Resumo do capítulo

Neste capítulo foi apresentada uma nova medida de avaliação da qualidade denominada discernibilidade. Tradicionalmente, a avaliação da qualidade do resultado produzido por funções de similaridade é feita por medidas baseadas em R&P. A discernibilidade utiliza não somente o retorno de elementos relevantes como critério de qualidade, mas analisa a habilidade de uma função de similaridade em separar elementos relevantes e irrelevantes. Por esta razão, um resultado produzido por uma função de similaridade ideal deveria atribuir aos elementos relevantes um escore igual a um e aos elementos irrelevantes um escore igual a zero.

O cálculo da discernibilidade é definido pelo intervalo formado pelo menor escore dos elementos relevantes e pelo maior escore dos elementos irrelevantes. O cálculo da discernibilidade é definido em duas etapas: (i) definição do limiar que separa os relevantes dos irrelevantes e (ii) cálculo da discernibilidade propriamente dito. A definição do limiar mais adequado para ser aplicado sobre um *ranking* produzido por uma determinada função de similaridade tem a finalidade de identificar o intervalo formado por s_rel e s_irrel, que representam, respectivamente, o menor escore de um elemento relevante e o maior escore de um elemento irrelevante.

Utilizando uma função de pontuação, que analisa o intervalo formado por s_rel e s_irrel, a definição do limiar adequado para separar relevantes e irrelevantes pode ser feita através de duas formas: (i) algorítmica e (ii) estatística. Para o primeiro caso foi apresentado o Algoritmo 2 na Seção 4.2.1, que retorna o um intervalo onde o limiar mais apropriado se encontra. É importante lembrar que um limiar adequado é aquele que minimiza falsos positivos e falsos negativos.

Ambos os métodos de definição do limiar fornecem um intervalo de limiares, cujos valores são usados para o cálculo da discernibilidade. O agrupamento dos elementos relevantes no topo do *ranking* e a diferença entre o menor escore relevante e maior escore irrelevante, são fatores que aumentam o valor da discernibilidade.

Algoritmo 2 ***BestThresh***

1: **Entrada**: n, $t_{\min}$, $t_{\max}$,$s^L_{rel}(k)$, $s^L_{irrel}(k)$,$k = 1...n, h$
2: **Saída**: $t^{\min}_{best}, t^{\max}_{best}, f_{max}$
3: $f\max = -2n$;
4: $n_{div} = (t_{\max} - t_{\min})/h$
5: **for** (a) $i = 0, ..., n_{div}$ **do**
6: $t = t_{\min} + i.h$;
7: $f(t) = 0$;
8: **for** (b) $k = 1$,..., n **do**
9: $d = 0$;
10: **if** $(s^L_{rel}(k) > t)$ **then**
11: $d = d + 1$;
12: **else**
13: $d = d - 1$;
14: **end if**
15: **if** $(s^L_{irrel}(k) < t)$ **then**
16: $d = d + 1$;
17: **else**
18: $d = d - 1$;
19: **end if**
20: $f(t) = f(t) + d$;
21: **end for** (b)
22: **if** ($f(t) \geq f_{\max}$) **then**
23: $f_{\max} = f(t)$;
24: **end if**
25: **end for** (a)
26: $t = t_{\min}$
27: **while** $(f(t) \neq f_{\max})$ **do**
28: $t = t + h$
29: **end while**
30: $t^{\min}_{best} = t$
31: $t = t_{\max}$
32: **while** $(f(t) \neq f_{\max})$ **do**
33: $t = t - h$
34: **end while**
35: $t^{\max}_{best} = t$
36: **if** $f_{\max} < 0$ **then**
37: $aux = t^{\max}_{best}$
38: $t^{\max}_{best} = t^{\min}_{best}$
39: $t^{\min}_{best} = aux$
40: **end if**
41: Escreva "O limiar ótimo está no intervalo" $[t^{\min}_{best}, t^{\max}_{best}]$

5 AVALIAÇÃO EXPERIMENTAL

Neste capítulo estão descritos os resultados experimentais referentes aos dois métodos descritos nos Capítulos 3 e 4. Os experimentos referentes ao método de estimativa de R&P estão descritos na Seção 5.3, onde estão descritos os objetivos da avaliação experimental, a estratégia de validação e os resultados obtidos em cada etapa do processo de estimativa apresentado. Da mesma forma, na Seção 5.4, estão descritos os experimentos realizados e os resultados obtidos com a nova medida proposta neste trabalho, a discernibilidade. Foi incluída uma avaliação comparativa entre a discernibilidade e precisão média, usando várias funções de similaridade.

Inicialmente, a Seção 5.1 apresenta uma visão geral das funções de similaridade utilizadas nos experimentos referentes aos dois métodos de estimativa de qualidade de funções de similaridade apresentados nos Capítulos 3 e 4. Em seguida, na Seção 5.2 uma breve descrição sobre os dados utilizados nos experimentos, que contempla dados reais e artificiais. As particularidades de cada método estão nas respectivas seções.

5.1 Funções de similaridade usadas

As funções de similaridade usadas nos experimentos envolvem tanto aspectos genéricos de acordo com o tipo de dados como caracter, texto, número, data, etc. como aspectos específicos do domínio usando conceitos semânticos como nome de pessoas, siglas, email, etc. (LIMA, 2002; DORNELES et al., 2004). Outro conjunto de funções de similaridade utilizadas nos experimentos é apresentado por Cohen (COHEN; RAVIKUMAR; FIENBERG, 2003), referentes aos projeto SecondString[1].

As funções de similaridade escolhidas envolvem dois tipos bem conhecidos (COHEN; RAVIKUMAR; FIENBERG, 2003): (i) baseadas em edição de distância, caracter a caracter, e (ii) baseadas em token (ou freqüência de termo). O critério de escolha para as funções de similaridade usadas nestes experimentos foi utilizar tanto funções de similaridade genéricas já conhecidas e utilizadas em ferramentas e implementações como o Projeto *SecondString*, assim como funções de similaridade mais específicas, desenvolvidas pelo especialista humano com conhecimento sobre o contexto semântico dos dados armazenados. As implementações específicas foram utilizadas certas funções de similaridade desenvolvidas no próprio grupo de pesquisa ao qual este trabalho está vinculado (LIMA, 2002; DORNELES et al., 2004). Foram

[1]Maiores informações disponíveis em *Secondstring: An opensource java toolkit of approximate string-matching techniques* (http://secondstring.sourceforge.net).

usadas as seguintes funções de similaridade:

- **Edit distance** (LEVENSHTEIN, 1966; HALL; DOWLING, 1980; NAVARRO, 2001) - esta função calcula o número mínimo de mudanças (inserção, exclusão e substituição) que são necessárias para fazer com que duas palavras ou seqüência de caracteres fiquem iguais. Tradicionalmente usada para medir a distância entre duas cadeias de caracteres baseada no custo de transformação de uma cadeia em outra, contando as diferenças tipográficas. No contexto desses experimentos essa métrica foi transformada para medir a similaridade ao invés da distância.

- **Acronyms** (DORNELES et al., 2004) - esta função é útil para analisar compatibilidade entre siglas e acrônimos com a forma não abreviada, como por exemplo, igualar "JOI"com "Journal of Informetrics".

- **Guth** (GUTH, 1976; BROU; OLSEN, 1986) - esta função é designada para detectar variações em nomes próprios.

- **N-gram** (NAVARRO, 2001; GRAVANO et al., 2001)- o escore é calculado baseado no número de caracteres que estão na mesma posição em cada *gram* (unidade mínima em que a seqüência de caracteres foi divida). Nos experimentos aqui realizados foi usado $n = 3$.

- **Jaccard** (JACCARD, 1912) - esta função simples determina a similaridade entre duas cadeias de caracteres $s1$ e $s2$, pela fórmula $(s1 \cap s2) \div (s1 \cup s2)$

- **Jaro** (JARO, 1989) - esta função baseia-se no número e na ordem de caracteres comuns entre dois termos ou seqüências de caracteres.

- **JaroWinkler** (WINKLER, 1999) - esta é uma variação da função Jaro que enfatiza a coincidência dos primeiros caracteres.

- **SLIM** - refere-se ao grau de sobreposição dos caracteres de cada termo. *Same-Letter Index Mixture* (SLIM) foi criada para fins experimentais, por William Cohen[2].

- **TF-IDF** (SALTON; MCGILL, 1983) - refere-se ao método baseado em freqüência amplamente usado na área de RI. A idéia do TF-IDF (*Term-Frequency Inverse Document Frequency*) é o uso de pesos para dar maior importância às palavras menos freqüentes. Para o casamento de uma seqüência de caracteres, TF é a freqüência do termo desta seqüência e IDF a freqüência inversa do termo em relação a coleção completa.

As quatro primeiras funções de similaridade (*Edit distance*, *Acronyms*, *Guth* e *N-gram*) estão reportadas nos dois conjuntos de experimentos, i.é, para validar o método semi-automático de estimativa de limiar apresentado no Capítulo 3 e para validar a nova medida de discernibilidade apresentada no Capítulo 4. As outras funções de similaridade estão reportadas somente nos experimentos realizados com a medida de discernibilidade, embora todas estejam implementadas na ferramenta FERP (Ferramenta para Estimativa de Revocacação e Precisão) (BONATO; STASIU; HEUSER, 2005) que implementa o método de estimativa de limiar descrita no Capítulo 3.

[2] http://secondstring.sourceforge.net/javadoc/com/wcohen/secondstring/SLIM.html

5.2 Características dos dados

Uma das premissas que norteiam este trabalho é a viabilidade de aplicação dos métodos definidos em grandes coleções, provenientes de sistemas corporativos reais. Por esta razão os experimentos realizados foram desenvolvidos sobre dados reais, provenientes de um sistema de seleção de candidatos para vestibular, cujos dados foram fornecidos pelo próprio candidato. Cada atributo selecionado originou uma coleção separada, como por exemplo, a instituição de origem do candidato, a cidade, bairro ou rua onde o candidato reside. Foram definidos dois subconjuntos para reportar os experimentos para validar o processo de estimativa de limiar baseado em agrupamento por similaridade que foi descrito no Capítulo 3: cidades e ruas, constituindo duas coleções de dados. Estes dois conjuntos de dados foram escolhidos por apresentarem características diferentes, resumidas na tabela 5.1. As outras coleções foram utilizadas nos outros experimentos utilizando a nova métrica chamada *Discernibilidade* descrita no Capítulo 4. A coleção chamada de Títulos é considerada sintética, pois foi manualmente alterada. Trata-se de 18 títulos de artigo científicos originais que foram modificados simulando situações e variações na ortografia para testar as funções de similaridade utilizadas.

Tabela 5.1: Principais características dos dados usados no experimento.

Conjunto de dados	Número de instâncias na coleção	Número de objetos reais
Cidades	10180	387
Ruas	10180	6944
Bairros	10180	1199
Instituições	3500	2377
Títulos	150	18

A Tabela 5.1 mostra o número de elementos (ou instâncias) nas coleções utilizadas. A coluna "número de instâncias" foi obtida através da contagem do número de elementos de cada coleção. Por "número de objetos reais", entende-se o número de instâncias que `realmente` se referem a `objetos distintos` do mundo real. A discrepância entre esses dois números corresponde às diferentes representações e repetições, incluindo erros de digitação, ortografia, entre outros. As coleções `Cidades`, `Ruas`, `Bairros` e `Instituições` foram obtidas de um sistema de informações corporativo, cuja finalidade é o registro de candidatos para o vestibular. Estas coleções representam dados reais, sem nenhum tipo de processamento prévio. A entrada dos dados foi feita pelo próprio candidato através de uma interface do sistema corporativo.

Um contraste pode ser observado em duas coleções: `Cidades` e `Ruas`, embora estejam relacionadas ao mesmo grupo de candidatos. A coleção de `Cidades` contém relativamente poucos objetos do mundo real (387). Este fato ocorre porque a maioria dos candidatos residem nas cidades próximas da instituição onde o candidato prestará o exame do vestibular. Na coleção de `Cidades` aproximadamente 45% dos valores correspondem ao mesmo objeto do mundo real, caracterizando um significativo número de repetições. Já a coleção de `Ruas` contém muitos objetos diferentes (2377), pois o número de ruas diferentes é muito maior que o número de cidades.

Proporcionalmente, isso significa que existem menos objetos repetidos se comparado com a coleção de `Cidades`.

Já a coleção de `Títulos` corresponde a uma coleção de dados modificados, criada propositalmente com erros e variações ortográficas. Foram escolhidos 18 títulos de artigos científicos diferentes de uma base de dados de referências bibliográficas. As modificações (como por exemplo, adicionando, alterando, removendo e/ou trocando caracteres ou palavras) têm a finalidade de simular possíveis erros de digitação que poderiam ter ocorrido no momento de cadastrar tais títulos. Ao todo, 150 títulos de artigos foram gerados, incluindo os originais.

5.3 Experimentos usando o método semi-automático de estimativa de R&P

Conforme apresentado no Capítulo 3, o método de estimativa de R&P para vários limiares apresenta etapas distintas, conforme apresentado na Figura 3.1: (i) amostragem, (ii) agrupamento por similaridade, (iii) consulta por similaridade e (iv) seleção do limiar apropriado. Este processo foi implementado em uma ferramenta resultante de um trabalho de conclusão do curso de graduação. Uma descrição detalhada da ferramenta é apresentada em (BONATO; STASIU; HEUSER, 2005).

Considerando o processo de amostragem e os algoritmos de similaridade utilizados, os valores estimados têm validade para a função de similaridade e para a coleção usada. Para obter confiança nos valores estimados, o processo de estimativa de limiar descrito aqui, baseia-se nas seguintes premissas:

1. Assume que os valores de R&P que foram calculados para a amostra, possam ser generalizados para a coleção completa;

2. Assume que o processo de agrupamento dividiu corretamente a amostra, o que significa que cada grupo representa exatamente um e somente um objeto do mundo real.

3. É aceitável que limiar "ótimo" estimado seja dependente da coleção e da função de similaridade usada.

As duas primeiras premissas estão diretamente relacionadas com o fato de permitir o uso de coleções com grande volume de dados e procurar minimizar a dependência de intervenção do especialista humano. Tais afirmações implicam em mecanismos e estratégias para validar que: (i) o resultado de estimativa obtido na amostra é válido para a coleção toda, conforme apresentado em 5.3.4 e (ii) cada grupo contém somente representações de um objeto do mundo real (5.3.3). Já a terceira premissa é derivada do fato observado durante o desenvolvimento desta pesquisa. Pode ser comprovado pelos experimentos que, usando funções de similaridade diferentes sobre a mesma coleção, obtém-se resultados diferentes. Da mesma forma, usando a mesma função de similaridade sobre coleções diferentes, produz resultados diferentes. Por resultados diferentes entende-se os valores de R&P em função dos limiares escolhidos. Portanto, o limiar estimado é dependente da coleção e da função de similaridade escolhida, de forma semelhante ao que ocorre com os dados estatísticos obtidos nos bancos de dados tradicionais.

5.3.1 Objetivos

Os experimentos realizados nesta seção têm por objetivo mostrar a aplicação do método de estimativa de R & P para vários limiares apresentado no Capítulo 3. Por esta razão, os experimentos têm os seguintes objetivos:

- Verificar a representatividade das amostras em relação à coleção;
- Avaliar o grau de correção dos grupos formados no processo de agrupamento por similaridade;
- Validar se os resultados obtidos para a amostra podem ser aplicados à coleção.

5.3.2 Representatividade das amostras

Uma característica importante da qualidade da amostra é a análise do respectivo grau de representatividade da amostra em relação à coleção. Portanto, uma amostra adequada deve conter elementos de forma proporcional aos dados contidos na coleção. Neste trabalho, foram utilizadas amostras geradas tanto pelo processo de catação quanto pelo processo aleatório. Foram geradas diversas amostras sem sobreposição (conforme a Definição 4, com 50 elementos em cada amostra. Foi permitida a repetição de valores nas amostras porque o mesmo acontece na coleção. Foi determinado que cada amostra deve conter 50 elementos para que o especialista humano possa contar quantos objetos distintos existem na referida amostra. Uma amostra muito pequena não teria representatividade alta da coleção, assim como uma amostra com um grande número de elementos dificultaria a contagem por parte do especialista humano.

Primeiramente, é necessário verificar se as amostras são representativas o suficiente para então utiliza-las com confiança como representantes da coleção. A representatividade da amostra pode ser medida de forma estatística considerando dois aspectos: (i) a proporção quantitativa em relação à coleção e (ii) o conteúdo distribuído de maneira uniforme. Para analisar a representatividade da amostra foi feita uma análise estatística usando a coleção de `Cidades`. Considerando o primeiro aspecto, analisando o dados quantitativos da amostra e da coleção, é possível observar que:

- Amostras com 50 elementos representam 0,49% da coleção que tem um total de 10180 elementos;
- O tamanho da amostra é 203 vezes menor que a coleção;
- Considerando um conjunto de 40 amostras com 50 elementos cada, o número médio ($\overline{x}$) é de 19,4 objetos distintos comparados a 387 objetos distintos na coleção. A amostra tem 39% de objetos distintos enquanto a coleção tem 4%, o que é explicado pelo fato da coleção conter proporcionalmente mais elementos repetidos que a amostra.

Foram consideradas 40 amostras da coleção de `Cidades`, sobre as quais foi identificado o número de objetos distintos e comparado proporcionalmente com o número de objetos distintos da amostra existentes na coleção. Portanto, cada amostra gerou

um número de objetos distintos x que foi usado para o cálculo do desvio padrão $\sigma(n)$ foi obtido por:

$$\sigma(n) = \sqrt{\frac{1}{m^2(n-1)} \sum_{i=1}^{n} (x_i - \overline{x})^2}$$

variando de $n_{\min} = 3$ até $n_{\max} = 40$, onde cada elemento da amostra é denotado por x_i, onde $n \in [n_{\min}, n_{\max}]$ e o valor médio é denotado por $\overline{x} = (1/n) \sum_{i=1}^{n} x_i$.

Uma representação gráfica de $\sigma(n)$ em função de n, deve refletir em $n \to \infty$, $\sigma(n) \to \sigma_{est}$, onde σ_{est} é uma constante. Conforme representado de forma gráfica na Figure 5.1, $\sigma(n)$ estabiliza o desvio padrão em torno de 2.5, o que representa em torno de 13% de variação em relação ao valor médio ($x_i = 19.4$).

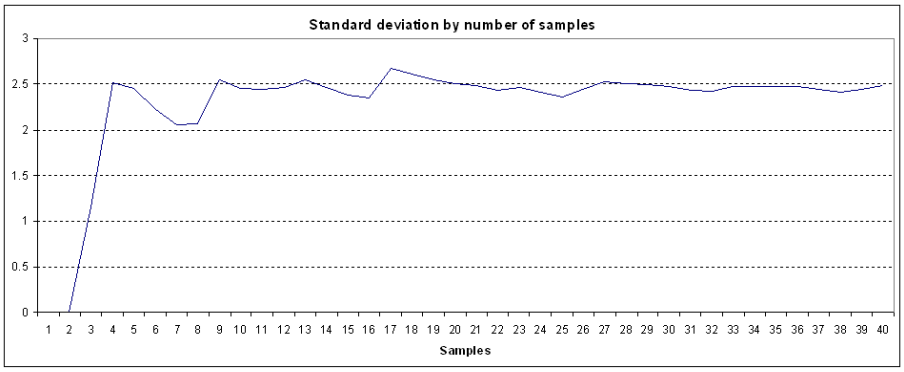

Figura 5.1: Desvio padrão sobre 40 amostras da coleção `Cidades`.

O outro aspecto analisado, refere-se à análise do conteúdo das amostras e seu respectivo grau de representatividade da coleção. Através da comparação do histograma de um conjunto de amostras com o histograma da coleção, é possível verificar graficamente se ocorre uma distribuição dos dados semelhante. Uma amostra representativa da coleção deve apresentar a curva resultante do histograma da amostra semelhante à curva resultante do histograma da coleção. Para gerar o histograma foram agrupados manualmente os elementos que representam o mesmo objeto do mundo real. Dessa forma, é possível obter uma distribuição do conteúdo tanto na amostra como na coleção. A Figura 5.2 mostra o histograma dos grupos de 4 amostras da coleção de `Cidades`. Na Figura 5.3 é possível observar o histograma dos grupos que representam os objetos distintos da coleção de `Cidades`.

É possível observar que existem semelhanças entre os histogramas representantes das amostras e da coleção, como mostrado nas Figuras 5.2 e 5.3. Existem muitos grupos formados por um único elemento e um grupo significativamente maior que os demais. Em ambas as figuras esse grupo que representa um único objeto real corresponde à cerca de 50% dos elementos. É possível identificar que as curvas que representam a forma dos dois histogramas, da amostra e da coleção, apresentam uma significativa semelhança.

5.3.3 Resultados agrupamento por similaridade

Para gerar os grupos de elementos que representam o mesmo objeto do mundo real, foi utilizado o método de agrupamento hierárquico baseado em aglomeração

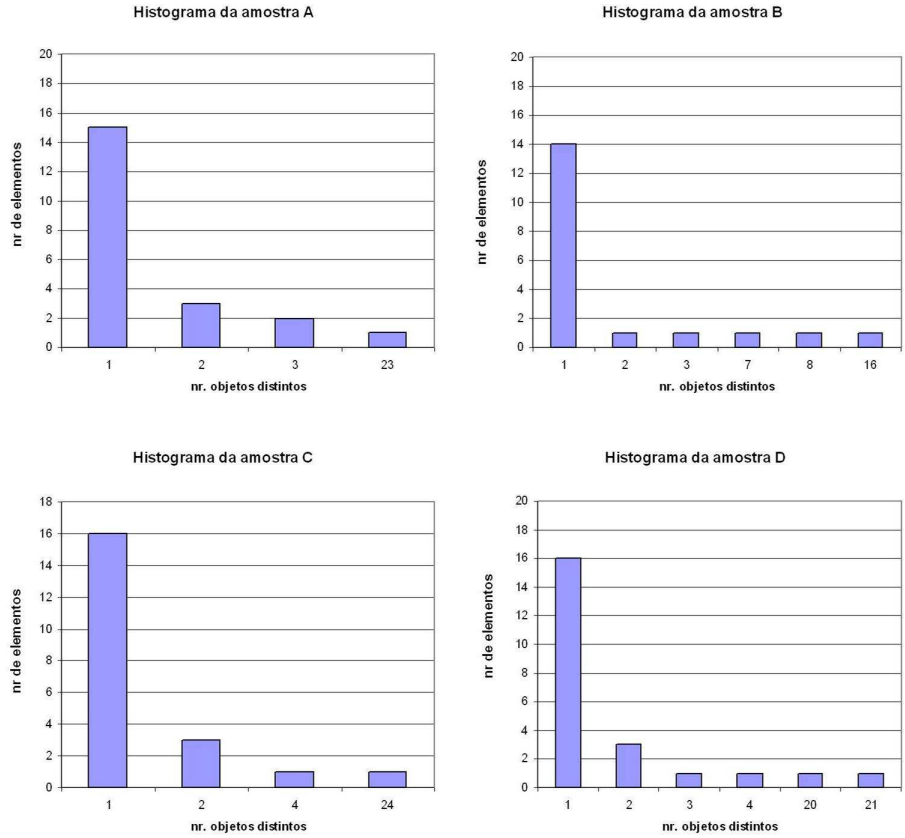

Figura 5.2: Histograma de cada amostra usando a coleção `cidades`.

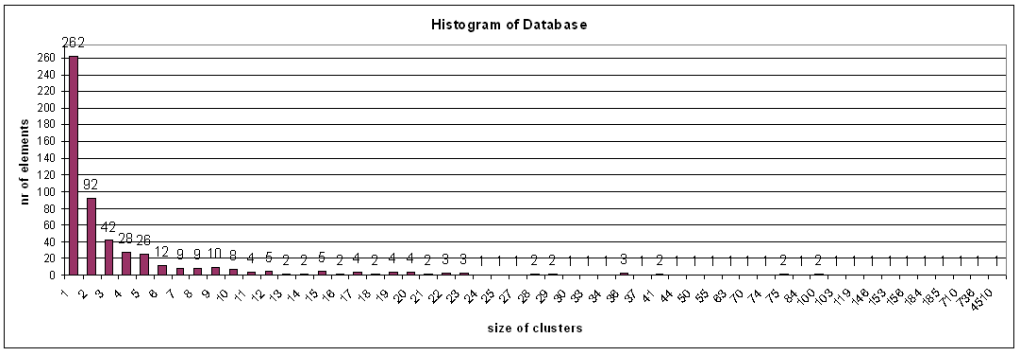

Figura 5.3: Histograma da coleção `cidades`.

(*Hierarchical Agglomerative Clustering Method*) (HARTIGAN, 1975). O algoritmo de ligação dos grupos usado foi o *SLINK(single-link) algorithm* (SIBSON, 1973), implementado com funções de similaridade descritas na Seção 5.1.

Para a coleção de `Cidades` foram extraídas 40 amostras com 50 elementos cada. Em seguida, foram utilizados algoritmos de agrupamento por similaridade para gerar o número de grupos igual ao número de objetos distintos indicado pelo especialista humano. Analisando o resultado do processo de agrupamento verificou-se que algumas funções de similaridade agruparam corretamente, mas outras não. Com base nessa análise foram encontrados três casos:

1. Os grupos corretamente formados, indicando nenhum erro.

2. O processo de agrupamento não encontrou o número de grupos igual ao informado pelo especialista humano.

3. O número de grupos encontrados foi igual ao informado pelo especialista humano, porém houveram agrupamentos incorretos.

Os casos 2 e 3 representam erros no processo de agrupamento, pelo fato da função de similaridade não ser adequada ao conjunto de dados usado. Foi verificado que somente em alguns casos o número de grupos igual ao número de objetos distintos não é suficiente para garantir que a função de similaridade agrupou de forma adequada. É possível que existam elementos que precisam trocar de grupo. A tabela 5.2 resume os erros encontrados depois da análise das 40 amostras extraídas da coleção `Cidades`.

Tabela 5.2: Resultado das amostras agrupadas e validadas pelo especialista humano.

Função de similaridade	Número de grupos é incorreto	Conteúdo do grupo é incorreto
Edit	nenhum	nenhum
Guth	1.6%	2%
N-Grams	0.4%	0.4%
Acronyms	nenhum	nenhum

Considerando os resultados obtidos, pode-se afirmar que é possível utilizar algoritmos de agrupamento por similaridade para definir o número de variações do objeto real. Pode-se dizer que é aceitável o número de erros encontrados, pois ainda existem alternativas que podem melhorar ainda mais esse resultado. Uma alternativa seria uma correção dos grupos incorretos através da comparação das funções de similaridade. Outra forma poderia ser com intervenção do especialista humano ou através de técnicas de aprendizado.

De maneira geral, o agrupamento é dependente da qualidade da função de similaridade usada para o conjunto de dados.

Pelo fato do processo de agrupamento por similaridade ser um dos elementos essenciais do método proposto neste capítulo, os critérios de avaliação da qualidade dos grupos não podem ser descartados. Uma distribuição dos elementos da amostra em grupos incorretos, estima valores também incorretos. Como forma de validação, foi estabelecido o uso de critérios internos e externos, através dos testes estatísticos, justamente por se tratar de um processo de aprendizado, através da iteração com diversas amostras, até obter um refinamento adequado dos valores estimados.

Conforme mencionado na Seção 2.5.4 do Capítulo 2, Halkidi et. al (HALKIDI; BATISTAKIS; VAZIRGIANNIS, 2001) apresenta métodos para avaliar a qualidade dos grupos formados por algoritmos de agrupamento por similaridade. Entre os critérios mencionados, neste trabalho foram utilizados critérios internos e externos, como forma de analisar estatisticamente o resultado obtido.

As avaliações por critérios externos e internos se assemelham em muitos pontos. Ambas são baseadas em testes estatísticos, e sua maior desvantagem é o alto custo computacional envolvido. Como o agrupamento por similaridade é feito sobre amostras não sobrepostas conforme a Definição 4, e o método utilizado neste

trabalho assume que as amostras terão um número limitado de elementos, a análise estatística o custo computacional não é um fator limitante. Os índices obtidos pelos testes estatísticos medem o grau de proximidade dos grupos formados com algum esquema de validação pré-definido (o que implica a participação de especialistas na validação). Os critérios relativos de avaliação do agrupamento resultante visam obter o melhor agrupamento possível, de acordo com certas restrições nos parâmetros de entrada do algoritmo de agrupamento.

Considerando os **critérios externos** de validação, a Tabela 5.3 apresenta o resultado da validação dos grupos formados através dos referidos métodos para cada função de similaridade usando a coleção de `Cidades`. Conforme já mencionado, os valores dos índices geralmente estão entre 0 e 1. Quanto mais altos os valores dos índices, maior é a qualidade do agrupamento, pois aumenta a coincidência entre o agrupamento feito pela função de similaridade e por um especialista humano com conhecimento do domínio.

Tabela 5.3: Resultado da validação do agrupamento por similaridade usando diferentes índices.

Função similaridade	*Estatística de Rand*	*Coeficiente de Jaccard*	*Índice de Folkes & Mallows*
Edit	1	1	1
Guth	0.9956	0.9780	0.9782
N-Grams	0.9960	0.9797	0.9896
Acronyms	1	1	1

Considerando a validação baseada por **critérios internos**, optou-se por verificar a confiabilidade do processo de agrupamento analisando uma instância específica, comparando os resultados obtidos na amostra com os resultados obtidos na base completa.

Foi escolhida uma instância que representa uma cidade, `Curitiba` por exemplo, e foram utilizados os resultados obtidos pelo agrupamento das amostras e ao invés de gerar números aleatórios, os resultados obtidos pelo agrupamento da coleção completa foram utilizados. Então foi aplicado o teste de hipótese Monte Carlo, onde a proporção esperada em cada conjunto de $m = 50$ elementos (cidades). Para tanto, é considerada uma amostragem de tamanho $n = 40$, i.e., são considerados $n = 40$ conjuntos de $m = 50$ elementos. O objetivo é medir o número de ocorrências da cidade de `Curitiba` no conjunto de m elementos escolhidos de forma aleatória.

Através de uma contagem heurística sobre a base de dados é obtida a proporção de $x_0 = 0.4489$ ocorrências de `Curitiba`. Então, considerando o exemplo, é testada a hipótese

$$\begin{aligned} H_0 &: \mu = x_0 \\ H_1 &: \mu \neq x_0. \end{aligned}$$

é obtido com a amostra um resultado igual a $\overline{x} = 0.4570$. A Estatística de Teste mais apropriada para o caso em que a variância é desconhecida é o teste *t-Student*. Neste caso, calcula-se a variância da amostra

$$s^2 = \frac{1}{n-1} \sum_{i=1}^{n} \left(\frac{x_i}{M} - \overline{x} \right)^2,$$

onde a estatística é dada por

$$t = \frac{\overline{x} - x_0}{s/\sqrt{n}} = 0.6.$$

Com este resultado, pode-se concluir que a hipótese seria rejeitada somente com um nível de confiança $\alpha = 60\%$, ou seja, somente com uma probabilidade de erro consideravelmente grande. É possível rejeitar a hipótese, quando a probabilidade de que a mesma se encontra na região crítica, i.é., quando H_0 é 60% verdadeira. Para tanto, $\alpha = 60\%$ define a região crítica $RC = [-\infty, -0.529] \cup [0.529, \infty]$, para $\alpha = 50\%$ contudo, não é rejeitada a hipótese. O intervalo de confiança pode ser construído para validar os resultados

$$IC(\overline{x} : \gamma) = \overline{x} \pm t_\gamma \frac{s}{\sqrt{n}}$$

onde γ é a probabilidade de uma medida pertencer ao IC Intervalo de Confiança.

$$\begin{aligned} IC(0.4570 &: 99\%) = 0.4570 \pm 2.704 \cdot 0.0136 \equiv [0.420\,23, 0.493\,77] \\ IC(0.4570 &: 95\%) = 0.4570 \pm 2.021 \cdot 0.0136 \equiv [0.429\,51, 0.484\,49] \\ IC(0.4570 &: 90\%) = 0.4570 \pm 1.684 \cdot 0.0136 \equiv [0.434\,10, 0.479\,90] \end{aligned}$$

5.3.4 Resultados obtidos na estimativa de R&P

Os experimentos foram realizados com coleções `Cidades` reais, conforme já descrito. Nesta seção, estão apresentados os resultados referentes a coleção `Cidades`. Considerando cada função de similaridade avaliada, os passos executados para a realização do experimento podem ser resumidos em:

1. A partir de várias as amostras geradas, foram selecionadas 4 amostras representativas;

2. Para cada amostra, um especialista humano informou o número de objetos distintos (no caso cidades diferentes) de acordo com os elementos da amostra. Essa tarefa corresponde a intervenção do especialista humano, ou administrador do banco de dados, por exemplo, em uma fase de pré-processamento.

3. As amostras foram agrupadas usando algoritmos por similaridade, conforme já mencionado.

4. Cada elemento da amostra foi usado como um objeto consulta sobre a amostra. Em cada consulta, os valores de R&P foram calculados de acordo com limiares definidos por 0.9, 0.8, 0.7, 0.5 e 0.3, utilizando o resultado do processo de agrupamento por similaridade de cada amostra.

5. Em cada limiar de cada amostra, foi calculado a média dos valores de R&P para o mesmo limiar para todas as consultas executadas.

6. Como foram utilizadas várias amostras, para a mesma função de similaridade, foi calculada a média aritmética dos valores médios de R&P em para cada limiar, obtendo as curvas de R&P.

7. De acordo com o critério da aplicação, i.é, revocação alta ou precisão alta, o limiar, ou intervalo de limiar é escolhido para ser usado em um mecanismo de busca.

8. De forma semelhante, o processo foi aplicado sobre a coleção para obter as curvas de R&P para a coleção.

9. Comparando a estimativa de R&P para a amostra com a coleção, os resultados apresentaram-se visualmente próximos, como pode ser observado na Figura 5.4.

10. Medir, de forma estatística, a distância entre os valores de R&P calculados para a amostra e os valores de R&P da coleção.

A estratégia de usar amostras extraídas do banco de dados, i.é. criar um subconjunto da coleção, apresenta algumas vantagens como redução do processamento, simplificação do processo pelo escopo de dados menor, facilidade e pouca intervenção do especialista humano, entre outras. Porém, para se tornar confiável, o processo de amostragem precisa ter o respaldo de refletir o mesmo comportamento da coleção completa. Caso contrário, os valores estimados estariam incorretos e poderiam causar muitas inconsistências nas consultas executadas por um mecanismo de consulta por similaridade.

A estratégia para verificar se os resultados obtidos para a amostra também se aplicam para coleção completa, envolve os seguintes passos:

1. Aplicação do método proposto para uma ou mais amostras representativas para obter valores de R&P para cada limiar pré-definido;

2. Obter a curva de R&P da amostra;

3. Considerar a coleção toda como uma amostra e aplicar o método proposto, obtendo os valores de R&P para os mesmos limiares usados na amostra;

4. Obter a curva de R&P da coleção;

5. Calcular a distância entre as curvas, considerando que quanto menor a distância, mais confiáveis são os resultados obtidos para a amostra.

A partir de experimentos usando dados reais, a validação é feita usando um método estatístico. A distância entre a curva de R&P da amostra e a respectiva curva da coleção é medida através do Desvio Quadrático Médio–`MSD`(*Mean Square Deviation*), definido pela equação (5.1). O `MSD` tem por objetivo medir a distância entre os pontos das duas curvas. Quanto menor for a distância, significa que as curvas estão mais próximas e são confiáveis, pois os mesmos valores podem ser utilizados nas duas curvas. Como o objetivo do método é escolher o limiar ou os limiares mais apropriados para a amostra, mas que também sejam adequados para a coleção, é desejável uma certa consistência do padrão das duas curvas.

$$f(x_M^b, x_M^s) = \frac{1}{n}\sum_{i=1}^{n}(x_{M,i}^b - x_{M,i}^s)^2, \tag{5.1}$$

onde $x^b_M = (x^b_{M,1}, x^b_{M,2}, ..., x^b_{M,n})$ é o valor de escore da coleção e $x^s_M = (x^s_{M,1}, x^s_{M,2}, ..., x^s_{M,n})$ é o valor da escore da amostra.

A aplicação do método do MSD pode ser considerado um problema de minimização, no sentido de que quanto mais se aproximar de zero, indica que os valores estimados para a amostra podem ser usados com maior confiança para consultas sobre a coleção. O valor de MSD é uma medida de distância entre as curvas dos valores de R&P estimados para amostra e para a coleção é menor. Uma sobreposição das duas curvas caracteriza uma situação ideal.

Com o intuito de avaliar os resultados estimados nas amostras, o processo de estimativa foi aplicado sobre a coleção completa, porém ao invés do agrupamento por similaridade, foi feita uma verificação dos dados através de um processo manual. Por processo manual entende-se a contagem dos elementos distintos na coleção por um especialista humano com conhecimento do domínio. Cada um dos valores das amostras (4 amostras $*$ 50 valores por amostra = 200 elementos) foi utilizado como objeto consulta sobre a base e foram calculados os valores de R&P conforme os passos executados para a amostra. O procedimento foi repetido para cada uma das quatro funções de similaridade usadas.

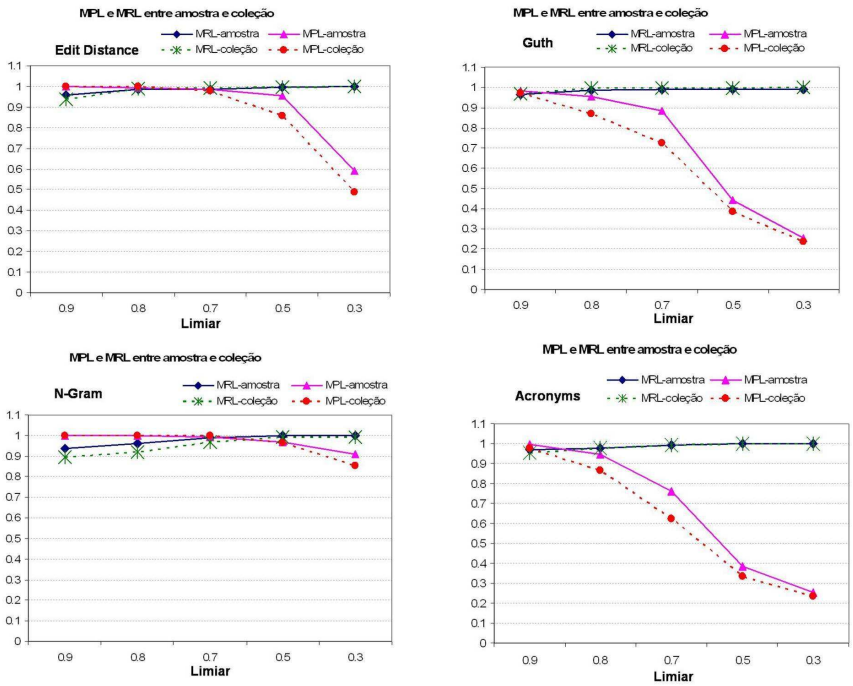

Figura 5.4: Cidades – Comparação da estimativa de R&P entre amostra e coleção.

Os resultados obtidos com a coleção `cidades` são apresentados na Figura 5.4, através da curvas de R&P da amostra e da coleção. Cada gráfico representa uma função de similaridade, que é indicada próximo ao eixo y. O eixo x corresponde aos valores de limiar estabelecidos e o eixo y corresponde aos valores de escore produzidos pela respectiva função de similaridade.

A curva de de R&P representa a qualidade da função de similaridade. Quanto mais altos os valores tanto para precisão e quanto para revocação, melhor a qualidade da função de similaridade. Por esta razão, os resultados apresentados na Figura 5.4 mostram que algumas funções de similaridade são mais adequadas que outras. Neste

caso, *Edit Distance* e *N-Grams* são funções de similaridade melhores, pois os valores de R&P tendem a ser mais altos e são menos dependentes dos valores dos limiares, definindo uma curva mais acentuada no canto superior direito. *Guth* e *Acronyms* são menos adequados pelo fato da precisão cair rápido com valores de limiar menores.

Com a finalidade de interpretar a aplicação da curva de R&P sobre dados obtidos da amostra e da coleção apresentado na Figura 5.4, é importante ressaltar dois aspectos importantes:

- O primeiro refere-se à amostra, a qual deve representar de forma fiel a coleção, o que resulta na proximidade das curvas de R&P. Em uma situação onde as duas curvas estejam sobrepostas, significa que os valores obtidos para a amostra e para a coleção são iguais.

- O segundo refere-se à distância entre as duas curvas. A curva de R&P correspondente a amostra foi obtida com o uso de algoritmos de agrupamento por similaridade, portanto, representam os valores estimados. Os valores de R&P correspondentes à coleção foram calculados manualmente. Por esta razão, é desejável a curva da amostra seja próxima à curva da coleção.

Os dois aspectos ressaltados acima implicam em medir a proximidade dos valores de R & P obtidos da amostra e da coleção. Conforme apresentado no Capítulo 3, foi calculado o Desvio Quadrático Médio-MSD(*Mean Square Deviation*) definido pela Equação (5.1). É importante lembrar que o MSD tem por objetivo medir a distância entre os pontos das duas curvas Portanto, quanto menores os valores obtidos, significa que as curvas estão mais próximas.

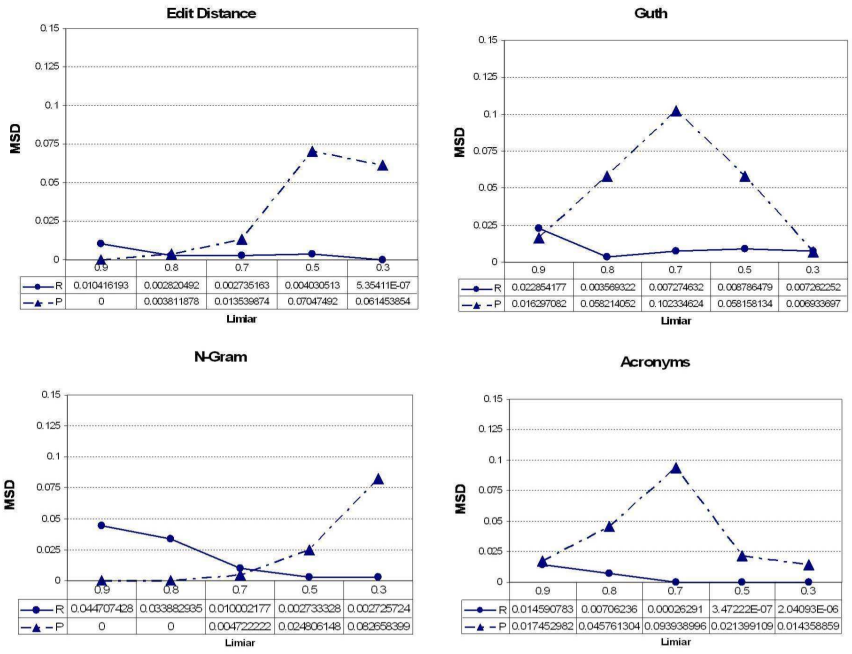

Figura 5.5: Cidades – Desvio Quadrático Médio (MSD) entre amostras e a coleção.

A Figura 5.5 mostra graficamente os valores calculados pela Equação (5.1), referente ao desvio quadrático médio (MSD). Foram utilizados os mesmos limiares

adotados nas curvas de R&P, para cada função de similaridade. A Figura 5.5 representa numericamente a distância entre os valores obtidos de R&P para as amostras comparado com os valores de R&P obtidos para a coleção. O eixo das coordenadas representa os valores obtidos pela Equação 5.1, que representa a distância entre amostra e coleção, para diferentes limiares representados no eixo da abscissa. Quanto mais altos os valores de MSD, maior é a diferença entre amostra e coleção. Como é possível observar, os valores são baixos indicando, que os valores estimados são próximos dos valores reais obtidos da coleção. Fato este que indica a viabilidade do processo de amostragem.

O mesmo processo, inclusive com a medição da distância entre as curvas de R&P usando MSD, foi aplicado utilizando outras coleções. Os resultados foram semelhantes aos obtidos para a coleção de `Cidades`. Os detalhes e respectivos resultados encontram-se no relatório técnico (STASIU; HEUSER; SILVA, 2004).

5.3.5 Considerações sobre o processo de estimativa de R&P

Os experimentos realizados mostram que os valores de R&P estimados para vários limiares nas amostras é semelhante aos valores estimados para a coleção. Por esta razão, o agrupamento por similaridade mostrou ser uma alternativa viável para tornar o processo de avaliação da qualidade baseada na estimativa de R&P semi-automático, com pouca intervenção do especialista humano. O processo de amostragem pode ser repetido várias vezes atuando como um mecanismo de aprendizado, refinando os valores estimados e tornando-os mais significativos. Dessa forma, o método proposto é aplicável em grandes coleções, permitindo maior flexibilidade nas consultas por similaridade.

Durante o processo de estimativa foi observado nos grupos formados que havia diferença entre fazer a estimativa mantendo os elementos duplicados, conforme ocorre na coleção, e removendo as duplicações sintaticamente idênticas. Os valores duplicados direcionavam para valores mais altos de similaridade. Porém, esta é uma característica dos dados, o que pode gerar estimativas tendenciosas. O uso de amostras não sobrepostas permite maior confiabilidade nos valores estimados.

Um ponto importante é a definição de algoritmos de agrupamento eficientes, que possuam uma função de similaridade coerente com o conjunto de dados, pois a formação correta dos grupos é essencial para obter valores estimados mais próximos do ideal. Outra característica a ser observada é o tamanho da amostra. Uma amostra muito grande dificulta a contagem e identificação do número de objetos diferentes. Uma amostra muito pequena não é significativa e pode não representar claramente as propriedades dos dados armazenados na coleção.

Um problema ainda em aberto é a validação empírica de grandes bases de dados. Definir o conjunto de valores reais para calcular a R&P é um processo manual e desgastante. Por isso, o uso de agrupamento por similaridade facilita a identificação das variações de cada objeto real. Durante a implementação da ferramenta, verificou-se que os algoritmos de agrupamento por similaridade consomem recursos computacionais, o que muitas vezes não permite o uso de amostras com um grande número de elementos. Da mesma forma, para calcular os valores de R&P dos elementos da coleção, através do mesmo processo usado para as amostras, é preciso definir soluções parciais. Uma alternativa encontrada foi a definição de amostras contendo um grande número de elementos (com cerca de 300 ou 500 elementos) para simular a coleção completa. Dessa forma, foi possível avaliar de forma mais extensiva e au-

tomática o uso de diferentes funções de similaridade, com diferentes coleções, com o auxílio de uma ferramenta automatizada (BONATO; STASIU; HEUSER, 2005). É importante salientar que os resultados descritos sobre o método de estimativa de limiar foram realizados de forma manual. A ferramenta foi desenvolvida posteriormente para expandir a validação do método para outras coleções e funções de similaridade.

5.4 Experimentos usando discernibilidade

Os experimentos descritos nesta sessão foram realizados para avaliar os dois métodos desenvolvidos para a definição do limiar mais adequado: baseado na função de recompensa e baseado em probabilidade. O primeiro, apresentado na Seção 4.2.1, baseia-se na idéia de recompensa para os escores encontrados dentro do intervalo esperado para uma métrica de boa qualidade, i.e., valores maiores que s^L_{rel} para os elementos relevantes e menores que s^L_{irrel} para os elementos irrelevantes. O segundo método, descrito na Seção 4.2.2, utiliza a abordagem de uma distribuição normal bivariada.

Os objetivos dos experimentos envolvendo a discernibilidade poder ser resumidos em:

- Apresentar a discernibilidade como medida de qualidade comparando-a com a precisão média;
- Mensurar a qualidade de uma função de similaridade;
- Apresentar os resultados obtidos por métodos diferentes para atestar a validade da discernibilidade como medida de qualidade.

5.4.1 Exemplo de uma amostra com s_{rel} e s_{irrel}

Uma amostra de 120 elementos foi criada a partir da coleção de títulos de artigos científicos, conforme descrito na Seção 5.2. Cada um dos elementos da amostra foi utilizado como uma consulta sobre a coleção. A similaridade entre a consulta e os elementos da coleção foi calculada usando a função de similaridade *Edit distance* (LEVENSHTEIN, 1966; NAVARRO, 2001; HALL; DOWLING, 1980).

Como foi dito na Seção 4.2, um especialista humano humano identificou os elementos retornados como relevantes ou irrelevantes. Baseado nesta identificação, os valores para $s^L_{rel}(k)$ e $s^L_{irrel}(k)$ foram obtidos. A Figura 5.6 mostra graficamente os valores de $s^L_{rel}(k)$ e $s^L_{irrel}(k)$ em função da $k-$ésima consulta, para uma amostra de 120 elementos (n). É importante notar que em alguns pontos $s^L_{irrel}(k)$ é maior que $s^L_{rel}(k)$, o que indica que neste ponto a função de similaridade não separou corretamente os elementos relevantes dos irrelevantes.

5.4.2 Resultados usando o Algoritmo *BestThresh*

Considerando um número de $k = 1$ até n consultas e uma precisão de $h = (t_{\max} - t_{\min})/n_{div} = 0.001$, onde $t_{\max} = 1$, $t_{\min} = 0$ e $n_{div} = 1000$, o algoritmo BestThresh foi executado avaliando cada limiar calculado pela fórmula $t_i = t_{\min} + i \cdot h$, onde $i = 1, ..., n_{div}$. O resultado produzido pelo algoritmo indica que t_{best} está no intervalo $I = [0.524, 0.529]$. Todos os valores de limiar dentro desse intervalo obtiveram $f_{\max} = f^{edit}(n, t) = 154$. Portanto, qualquer valor pertencente a I seria um limiar

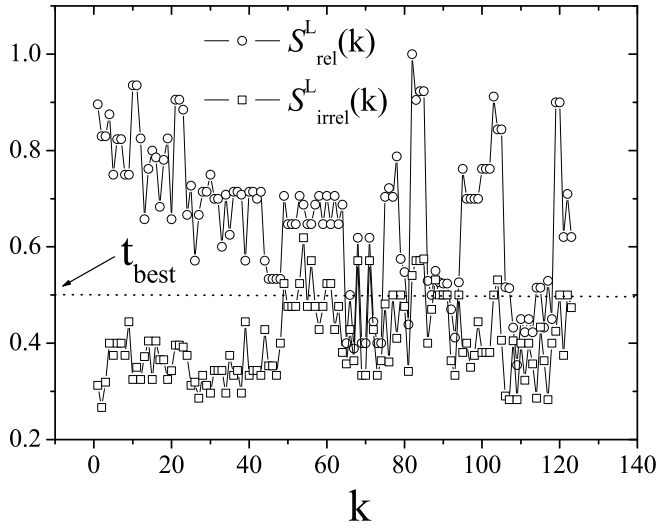

Figura 5.6: Menor escore relevante e maior escore irrelevante em função da $k-$ésima consulta para a função de similaridade L (*edit distance*).

adequado para realizar uma busca ao acaso usando esta função de similaridade em particular. A Figura 5.7 o gráfico de $f^{edit}(n,t)$ em função de t.

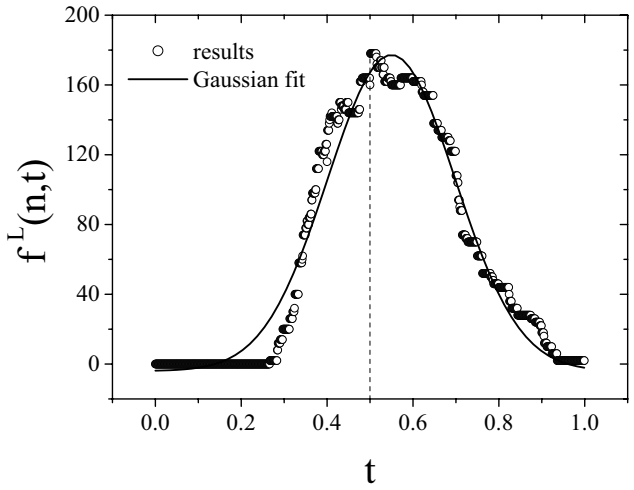

Figura 5.7: Gráfico de $f^{edit}(n,t)$ em função de t para a função *Edit distance.*

É importante notar que os valores de $f^{edit}(n,t)$ são distribuídos simetricamente em função de t, como pode ser observado na Figura 5.7. A curva contínua na Figura 5.7 segue o comportamento de uma curva Gaussiana, o que mostra como os valores estão distribuídos.

É possível aplicar um teste de robustez para avaliar como t_{best} comporta-se a medida que (n) cresce. Intuitivamente, t_{best} deveria convergir para um valor constante t^{∞}_{best} quando extrapolando $k \to n$. A Figura 5.8 mostra que t_{best} estabiliza quando o número de consultas se aproxima de 120.

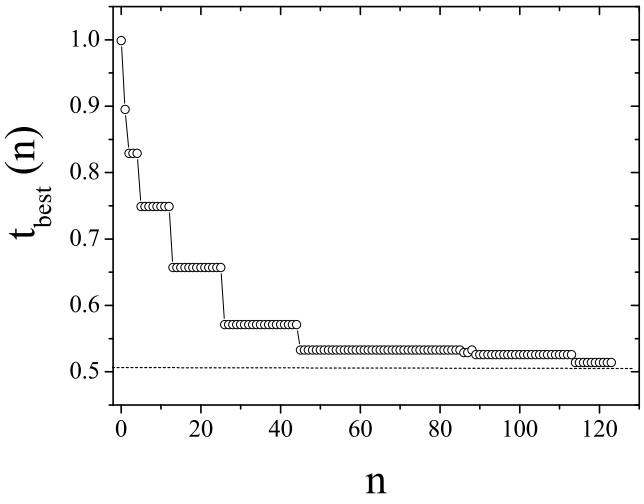

Figura 5.8: Evolução de t_{best} em função do tamanho da amostra. O gráfico claramente mostra que t_{best} converge para valores no intervalo $I = [0.524, 0.529]$ quando $n \to \infty$.

5.4.3 Resultados usando a distribuição normal bivariada

Os parâmetros $\left(\left\langle s^L_{irrel}\right\rangle, \sigma^2(s^L_{irrel})\right)$ e $\left(\left\langle s^L_{rel}\right\rangle, \sigma^2(s^L_{rel})\right)$ foram calculados para as consultas da amostra usando a mesma função de similaridade (*edit distance*). Os Histogramas para os valores de s^L_{irrel} e s^L_{rel} foram calculados. As Figuras 5.9 mostram como s^L_{irrel} e s^L_{rel} são distribuídos em torno do valor médio de $\left\langle s^L_{irrel}\right\rangle$ e $\left\langle s^L_{rel}\right\rangle$. As curvas contínuas nos gráficos denotam ajustamento normal. Calculando a correlação $\rho = 0.022$, obtém-se a normal bivariada, mostrada na Figura 5.10.

Utilizando o aplicativo *Maple* foi calculado $F(t)$ especificado pela Equação 4.13, definida na Seção 4.2.2 distribuindo t no intervalo $[0, 1]$ para encontrar o valor de t que produz $F(t)$. Os resultados são mostrados na Figura 5.10 onde $F(t)$ é representado graficamente em função de t. Note-se que a probabilidade de $F(t)$ é aproximadamente uma PDF normal uma vez que a curva contínua é o ajuste de uma PDF normal. A figura mostra que o valor de t_{best} é 0.515. Este valor foi obtido pelo método estatístico definido na Seção 4.2.2 considerando a precisão de $h = 0.001$. É importante lembrar que o valor para t_{best} calculado pelo algoritmo BestThresh foi no intervalo $I = [0.524, 0.529]$. Tal fato mostra que ambos os métodos estão em concordância.

5.4.4 Resultado do uso da função de discernibilidade

O objetivo desta seção é usar os métodos de definição de limiar para avaliar a qualidade de diferentes funções de similaridade. Lembrando que uma função de similaridade pode ser considerada melhor que outra se a mesma produz uma melhor separação dos elementos relevantes e irrelevantes retornados como resposta a uma consulta. De acordo com este trabalho, a função de similaridade que tem maior $f_{\max}$ é considerada melhor que outra com um $f_{\max}$ menor. Da mesma forma, o tamanho de variação do intervalo para t_{best} é outro indicador de qualidade da função. Posto que uma boa função de similaridade deveria separar elementos relevantes dos

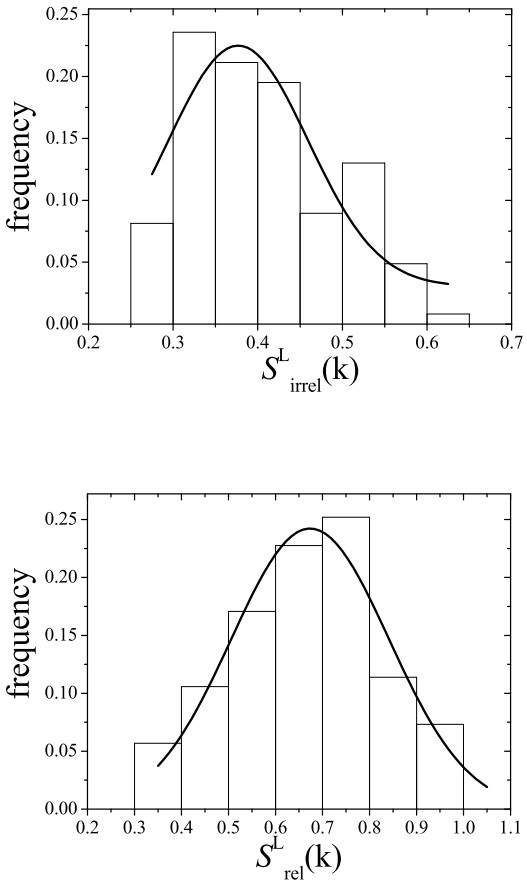

Figura 5.9: Histogramas para os valores de s^L_{irrel} e s^L_{rel}. As curvas contínuas mostram o ajuste gaussiano para estes histogramas.

irrelevantes, quanto maior o intervalo, melhor.

Além das funções de similaridade descritas no início deste Capítulo, na Seção 5.2 foram definidas três situações representadas específicas, por métricas ou funções de similaridade fictícias denominadas por:

- `Ideal` (ou *optimal*) - representa a similaridade perfeita; esta função separa corretamente elementos relevantes e irrelevantes, mantendo-os o mais longe possível no resultado classificado, i.e., para todas as consultas $s_{rel} = 1$ e $s_{irrel} = 0$.

- `Vazia` - função que atribui o escore zero para todos os elementos, independente se são relevantes ou não. É o caso de uma função aplicada para um domínio inadequado. Neste caso, todas as consultas possuem $s_{rel} = s_{irrel} = 0$.

- `Pior Possível` - o pior caso é resultado da função de similaridade que coloca todos os irrelevantes com um escore maior que o escore dos elementos relevantes, i.e. para todas as consultas $s_{rel} = 0$ and $s_{irrel} = 1$.

A precisão de $h = 0.001$ foi usada para calcular o intervalo $\left[t^{\min}_{best}, t^{\max}_{best}\right]$ pelo algoritmo BestTresh. Os limites do intervalo foram usados para calcular a

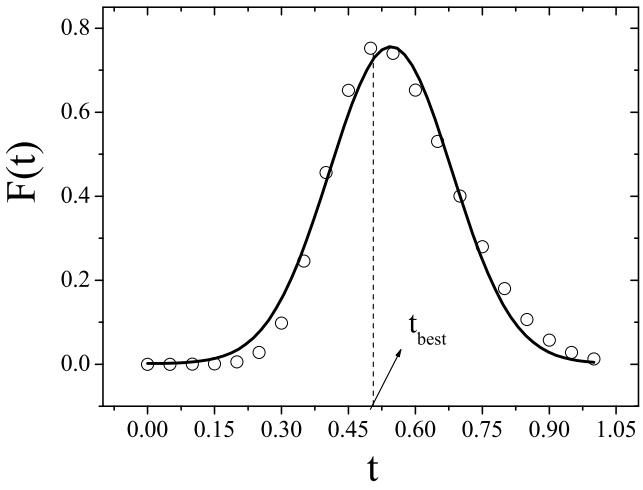

Figura 5.10: Gráfico de $F(t) \times t$, mostrando que o valor de t_{best} é o valor de t correspondente ao maior valor de $F(t)$.

`discernibilidade` para a função de similaridade. t_{best} foi calculado usando distribuição normal bivariada.

Tabela 5.4: Comparação entre diferentes funções de similaridade.

Função$_{sim}$	f_{max}	**Discernibilidade**	$\left[\ t_{best}^{\min},\ t_{best}^{\max}\right]$	t_{best}	**Interv. de confiança**
Jaro-Winkler	184	0.4048	$[0.768, 0.811]$	0.791	$[0.716, 0.887]$
Jaro	178	0.3713	$[0.755, 0.756]$	0.753	$[0.703, 0.888]$
Acronyms	158	0.3616	$[0.601, 0.666]$	0.592	$[0.455, 0.781]$
Edit distance	154	0.3233	$[0.524, 0.529]$	0.515	$[0.404, 0.697]$
N-gram	134	0.2851	$[0.576, 0.588]$	0.553	$[0.440, 0.723]$
Guth	38	0.0821	$[0.905, 0.911]$	0.801	$[0.686, 0.896]$
Jaccard	16	0.0468	$[0.401, 0.428]$	0.301	$[0.169, 0.428]$
TFIDF	16	0.0438	$[0.578, 0.599]$	0.442	$[0.273, 0.614]$
`Ideal`*	240	0.9999	$[0.001, 0.999]$	$---$	$---$
`Vazia`*	0	0.0000	$[0.000, 0.000]$	$---$	$---$
`Pior Possível`*	-240	-0.9999	$[0.999, 0.001]$	$---$	$---$

Nota: As funções de similaridade representadas por (*) são fictícias.

Os resultados estão apresentados na Tabela 5.4. A segunda coluna da tabela mostra os valores para f_{max} que representam o número de pontos obtidos por t_{best} para uma determinada função. A terceira coluna mostra o resultado para *discernibilidade*. A quarta coluna mostra o intervalo para t_{best} calculado pelo algoritmo *BestThresh*. A quinta coluna contem o valor mais apropriado para t_{best} calculado pelo método estatístico para definição do limiar. E a última coluna da tabela apresenta o intervalo de confiança construído a partir da distribuição $F(t)$ pelas Equações 4.15 e 4.16, apresentadas na Seção 4.3.

A Tabela 5.4 mostra que a diferença absoluta dos dois métodos propostos (algorítmico e estatístico) em concordância. A interpretação é feita analisando o valor produzido para t_{best} se encaixam tanto no intervalo definido pelo algoritmo *Best-Thresh* como dentro do intervalo de confiança produzido pelo método estatístico. É importante salientar que quanto melhor a qualidade da função de similaridade, melhor é a concordância entre os dois métodos. Além do mais, em todos os casos os valores calculados por ambos os métodos estão dentro de um intervalo de confiança aceitável. A Tabela 5.4 também mostra o resultado da `discernibilidade`. De acordo com tais valores, a melhor função real para o conjunto de dados analisado foi *Jaro-Winkler* e a pior foi *TFIDF*.

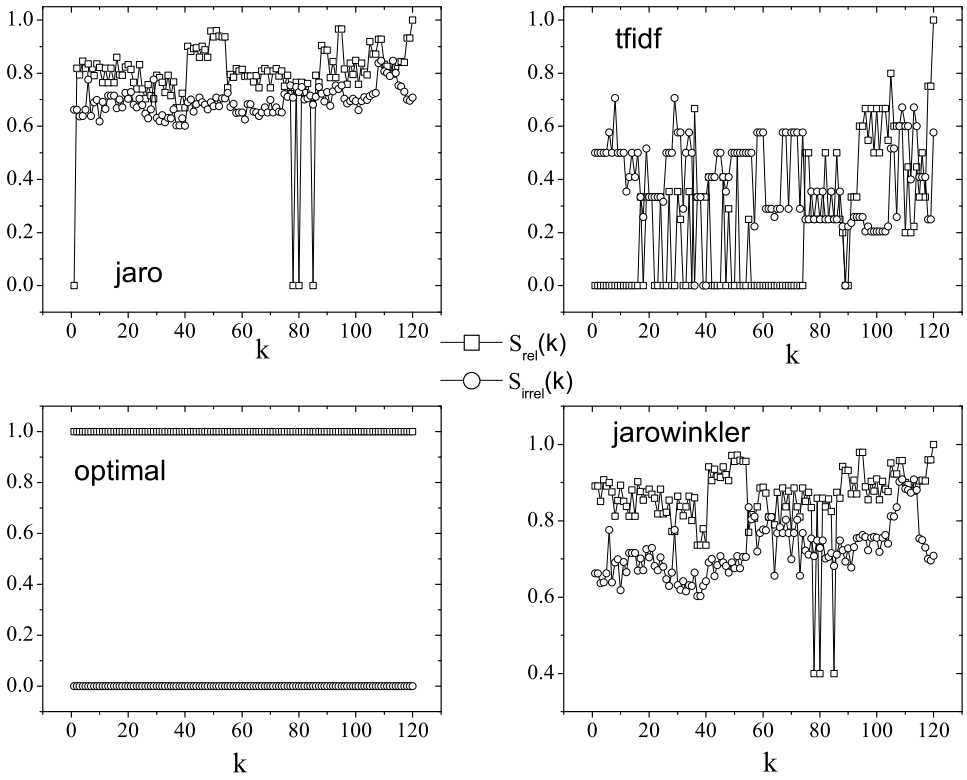

Figura 5.11: Distribuição de S_{rel} e S_{irrel} para diferentes funções de similaridade.

Como forma ilustrativa, a Figura 5.11 mostra a representação gráfica da distribuição de s_{rel} e s_{irrel} de 4 funções de similaridade diferentes: *jaro*, *jaro-winkler*, *optimal* e *tfidf*. De fato, a melhor separação entre relevantes e irrelevantes foi obtida por *jaro-winkler*, enquanto que usando TFIDF estes elementos freqüentemente embaralhavam e/ou ficaram muito próximos no *ranking*. O gráfico também mostra o comportamento da função artificial `Optimal`, que obteve a pontuação máxima de acordo com a função de `discernibilidade`. O comportamento da função Jaro, que obteve o segundo melhor resultado é apresentado de forma gráfica na Figura 5.11, mostra que embora tenha separado adequadamente, a distância dos elementos relevantes e dos irrelevantes, é menor se comparado com o resultado da função *jaro-winkler*. É importante salientar que o resultado das funções de similaridade é dependente dos

dados, o que significa que um conjunto de dados diferente apresentaria um outro *ranking* de funções de similaridade.

As funções de similaridade da Tabela 5.4 com o símbolo $*$ são funções que a abordagem estatística não se aplica devido a natureza dos dados, i.e., não há variação nos valores usando estas funções, o que significa que os valores para s_{rel} e/ou s_{irrel} são constantes para a maioria das consultas. Em outras palavras, o desvio padrão para s_{irrel} ($\sigma(s^L_{rel})$ e $\sigma(s^L_{irrel})$) são próximos de zero. Contudo, tais dados podem ser usados para encontrar um limiar ótimo usando o algoritmo *BestThresh*.

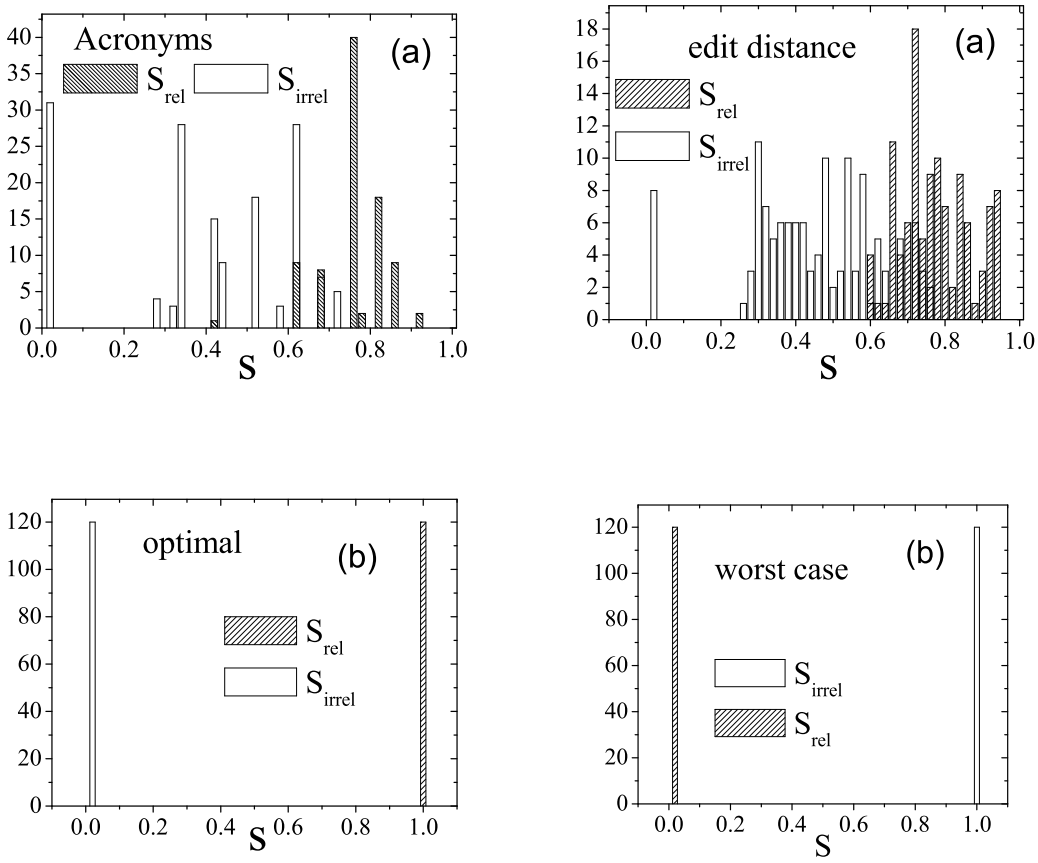

Figura 5.12: O eixo x representa os valores de limiar e eixo y representa o número de ocorrências, i.e., quantas consultas obtiveram o mesmo limiar para s_{rel} e s_{irrel}. Os gráficos identificados por (a) mostram exemplo de histograma para funções que são estatisticamente tratáveis e os gráficos identificados por (b) ilustram histogramas para funções não tratáveis estatisticamente.

Na Figura 5.12 é apresentado o gráfico de dispersão pra valores de similaridade usando dois tipos de funções. O tipo (a) representa funções estatisticamente tratáveis (ou que apresentam variabilidade nos dados); e o tipo (b) representa funções não tratáveis estatisticamente.

5.5 Comparação de discernibilidade com precisão média

Nesta seção será apresentada a diferença entre a dicernibilidade e a $MAvP$. Foram utilizados dados artificiais para os respectivos cálculos com o intuito é mostrar a sensibilidade da discernibilidade em capturar variações não perceptíveis pela $MAvP$. Portanto, estes dados não fazem parte dos resultados experimentais provenientes de dados reais.

De forma simplificada, os dados utilizados nas tabelas abaixo representam o resultado de um *ranking* obtido por um função de similaridade. Como exposto anteriormente, nos experimentos foi utilizado um conjunto de *rankings*. As Tabelas 5.5, 5.6 e 5.7 mostram o exemplo de consulta sobre o mesmo conjunto de dados, com três funções de similaridade diferentes: A, B e C. O exemplo, ilustrado em um único *ranking*, utiliza os valores do *ranking* para o cálculo da discernibilidade e da $MAvP$, conforme mostra a Tabela 5.8.

Tabela 5.5: Resultado obtido de acordo com a função de similaridade A.

Escore	**Elemento**	**Relevância**
1.0000	Ranking in Databases	Relevante
0.9581	Ranking on Databases	Relevante
0.8753	Ranking on DBs	Relevante
0.8391	Rankin on DBs	Relevante
0.7720	Ranking and DBs	Relevante
0.3312	Relational Databases	Irrelevante
0.3040	Ranking on IR	Irrelevante
0.2871	Ranking Correlation	Irrelevante

Tabela 5.6: Resultado obtido de acordo com a função de similaridade B.

Escore	**Elemento**	**Relevância**
1.0000	Ranking in Databases	Relevante
0.9873	Ranking on Databases	Relevante
0.9040	Ranking on DBs	Relevante
0.8755	Rankin on DBs	Relevante
0.7341	Ranking and DBs	Relevante
0.7221	Relational Databases	Irrelevante
0.7044	Ranking on IR	Irrelevante
0.7025	Ranking Correlation	Irrelevante

Interpretando os dados da Tabela 5.8, pode-se observar que enquanto o valor de $MAvP$ é 1.0 para as funções de similaridade A e B, pelo fato dos relevantes estar corretamente posicionado no topo do *ranking*, o valor de discernibilidade varia de forma significativa. Tal variação é influenciada pelo valor dos escores dos elementos. A função B, por exemplo, atribuiu o escore 0.7341 para o elemento relevante e 0.7221 para o elemento irrelevante. Como é possível observar, mesmo sendo valores próximos, a $MAvP$ não foi alterada se comparado aos valores obtidos com a função A. Este aspecto mostra a sensibilidade da discernibilidade quanto à habilidade da função de similaridade em diferenciar entre elementos relevantes e irrelevantes. Analisando os valores obtidos sobre o resultado da função C, nota-se que a função não

Tabela 5.7: Resultado obtido de acordo com a função de similaridade C.

Escore	**Elemento**	**Relevância**
1.0000	Ranking in Databases	Relevante
0.9581	Ranking on Databases	Relevante
0.7023	Relational Databases	Irrelevante
0.6789	Ranking Correlation	Irrelevante
0.6767	Ranking on DBs	Relevante
0.6089	Rankin on DBs	Relevante
0.5543	Ranking and DBs	Relevante
0.4412	Ranking on IR	Irrelevante

Tabela 5.8: Comparativo entre valores de $MAvP$ e *Discernibilidade*.

Função	$MAvP$	*Discernibilidade*
A	1.0000	0.7204
B	1.0000	0.4937
C	0.7960	- 0.4260

separou adequadamente os elementos, deixando elementos relevantes e irrelevantes misturados. Enquanto a discernibilidade reflete uma mudança considerável, a $MAvP$ apresenta uma ligeira variação. As diferenças entre a discernibilidade e a $MAvP$ mostram que ambas têm aplicações distintas.

Conforme descrito no Capítulo 2, na Seção 2.4, a precisão média (*Mean Average Precision-*$MAvP$) considera a ordem dos elementos do resultado até a posição corrente do *ranking*. Por esta razão, funções de similaridade que recuperam os valores mais relevantes nas primeiras posições do *ranking* são mais adequadas para consultas delimitadas por quantidade de elementos no resultado (*top-k queries*), pois permite que sejam avaliados e priorizados resultados relevantes nas primeiras posições.

Já o cálculo da discernibilidade, por levar em consideração o intervalo formado pelos escores do menor elemento relevante e do maior elemento irrelevante, mostra a capacidade da função de similaridade de separar os elementos relevantes dos irrelevantes. É intuitivo que quanto mais separados os relevantes dos irrelevantes, bem como, a separação mutuamente exclusiva dos dois conjuntos de elementos relevantes e irrelevantes, permita maior flexibilidade na escolha do limiar, mantendo somente os relevantes no conjunto resposta. Por esta razão, a discernibilidade captura variações nas consultas por abrangência que não são perceptíveis pelas medidas baseadas em precisão.

5.5.1 Resultados experimentais comparando discernibilidade e $MAvP$

Os experimentos apresentados nesta seção mostram a comparação entre *discernibilidade* e $MAvP$. Foram utilizadas as coleções reais e sintéticas descritas na Seção 5.2, com o número de consultas conforme apresentado na Tabela 5.9. É importante lembrar que cada consulta produz um *ranking* usando uma determinada função de similaridade. A avaliação de um conjunto de *rankings* através do critério de relevância é calculado em termos de *discernibilidade* e $MAvP$ é apresentada na Tabela 5.4 apresenta valores médios obtidos para cada função de similaridade.

Usando um julgamento por relevância é possível calcular os valores de

Tabela 5.9: Detalhes das coleções de teste usadas para comparar discernibilidade e $MAvP$.

Coleção	Número de consultas
`Títulos`	120
`Cidades`	300
`Bairros`	300
`Instituições`	100

Discernibilidade e $MAvP$ para um conjunto de consultas. A Tabela 5.4 mostra o resultado desta avaliação. Da mesma forma, é possível estabelecer uma visualização gráfica através de curvas de R & P. A Figura 5.13 mostra as curvas dos valores de R & P para a coleção de `Bairros`. Como foram utilizadas diversas consultas, o critério adotado foi comparar a discernibilidade com a $MAvP$, obtida conforme exposto no Capítulo 2, na Seção 2.4.

Comparando experimentalmente a discernibilidade com a $MAvP$, é possível observar diferenças. A discernibilidade é uma medida capaz de identificar particularidades da função de similaridade que não são expressas através da $MAvP$. Em certos casos, onde alterações no escore não influenciam no resultado da $MAvP$, a discernibilidade é mais sensível ao grau de separação do conjuntos dos relevantes e irrelevantes. A vantagem de ter uma informação sobre a distância entre o escore do menor elemento relevante e do maior irrelevante no *ranking* permite que se houver funções similares alternativas, um processador poderá optar por aquela que permite maior variação do limiar.

Os resultados experimentais, utilizando dados provenientes de sistemas reais, mostram que os valores para $MAvP$ são sempre maiores que os valores de *Discernibilidade*. O escore médio definido pela $MAvP$ é 0.9450, enquanto que *Discernibilidade* apresenta uma média de 0.3488, considerando todas as coleções utilizadas nos testes. A maior discrepância pode ser encontrada na coleção de `Títulos` entre as funções de similaridade *Jaccard* e *TFIDF* (linhas 7 e 8 na Tabela 5.4).

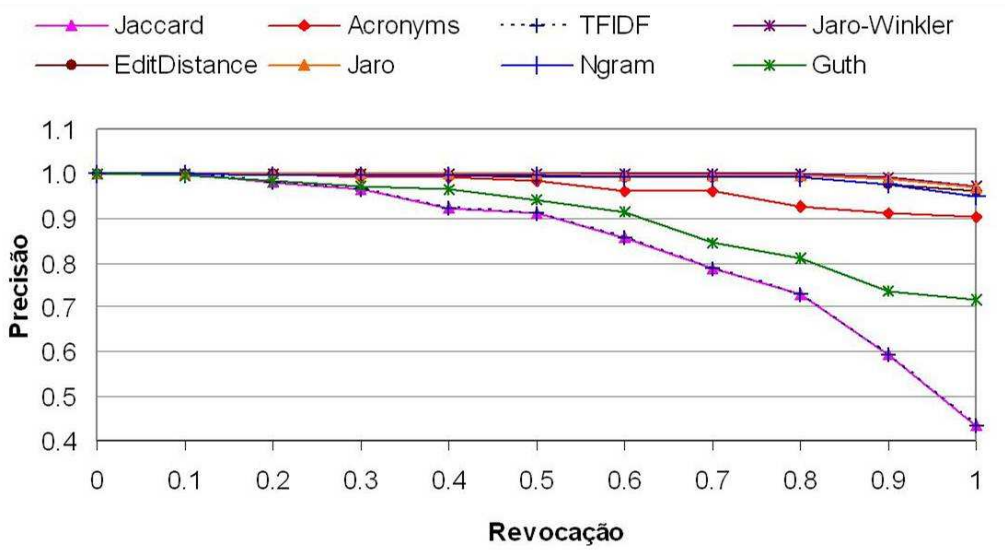

Figura 5.13: Curvas de R & P em ordem decrescente de $MAvP$.

Tabela 5.10: *Discernibilidade* e *MAvP*(*Mean Average Precision Scores*) para diferentes funções de similaridade.

	Database	Sim Function	*MAvP*	*Discernability*
1	`Títulos`	Jaro-Win	0.9908	0.4048
2	`Títulos`	Jaro	0.9898	0.3755
3	`Títulos`	EditDist	0.9859	0.3233
4	`Títulos`	Ngram	0.9841	0.2893
5	`Títulos`	Acronyms	0.9643	0.3617
6	`Títulos`	Guth	0.8920	0.0822
7	`Títulos`	Jaccard	0.8332	0.0468
8	`Títulos`	TFIDF	0.8332	0.0438
9	`Títulos`	slim	0.5636	0.1237
10	`Cidades`	Acronyms	0.9898	0.4415
11	`Cidades`	Jaro	0.9886	0.4193
12	`Cidades`	Jaro-Win	0.9884	0.4195
13	`Cidades`	EditDist	0.9896	0.4488
14	`Cidades`	Ngram	0.9863	0.4152
15	`Cidades`	Slim	0.9708	0.3458
16	`Cidades`	Guth	0.9575	0.3460
17	`Cidades`	Jaccard	0.8961	0.3408
18	`Cidades`	TFIDF	0.8958	0.3408
19	`Bairros`	Ngram	0.9743	0.4177
20	`Bairros`	Acronyms	0.9710	0.4253
21	`Bairros`	EditDist	0.9708	0.4317
22	`Bairros`	Slim	0.9667	0.3687
23	`Bairros`	Jaro	0.9646	0.4022
24	`Bairros`	Jaro-Win	0.9612	0.3982
25	`Bairros`	Guth	0.9414	0.3965
26	`Bairros`	Jaccard	0.9171	0.5210
27	`Bairros`	TFIDF	0.9171	0.4460
28	`Instituições`	Acronyms	0.9987	0.3540
29	`Instituições`	EditDist	0.9970	0.3616
30	`Instituições`	Ngram	0.9925	0.3615
31	`Instituições`	Jaccard	0.9678	0.3603
32	`Instituições`	TFIDF	0.9678	0.3471
33	`Instituições`	Slim	0.9633	0.3206
34	`Instituições`	Jaro	0.9623	0.3817
35	`Instituições`	Jaro-Win	0.9514	0.3526
36	`Instituições`	Guth	0.9354	0.3420

Comparando os resultados obtidos pelas duas métricas, *MAvP* e *Discernibilidade*, os valores obtidos diferem, mostrando diferentes aspectos. Um aspecto interessante é que em três das quatro coleções, os resultados obtidos para *MAvP* através das funções de similaridade *Jaccard* e *TFIDF* foram idênticas. Entretanto, o valor obtido para *Discernibilidade*, demonstra que estas duas funções de similaridade são diferentes. A razão dessa diferença é porque a *Discernibilidade* leva em consideração o valor do escore. Considerando que o *ranking* produzido

Tabela 5.11: Coeficientes de correlação entre *Discernibilidade* e $MAvP$.

Coleção	***Pearson's*** r	***Kendall's*** τ	***Spearman's*** ρ
`Títulos`	0.662	0.704	0.845
`Cidades`	0.791	0.817	0.946
`Bairros`	-0.650	-0.141	-0.226
`Instituições`	0.289	0.310	0.393

por estas duas funções de similaridade é o mesmo, *Jaccard* apresenta um resultado separando com uma distância maior no *ranking* os elementos relevantes dos irrelevantes.

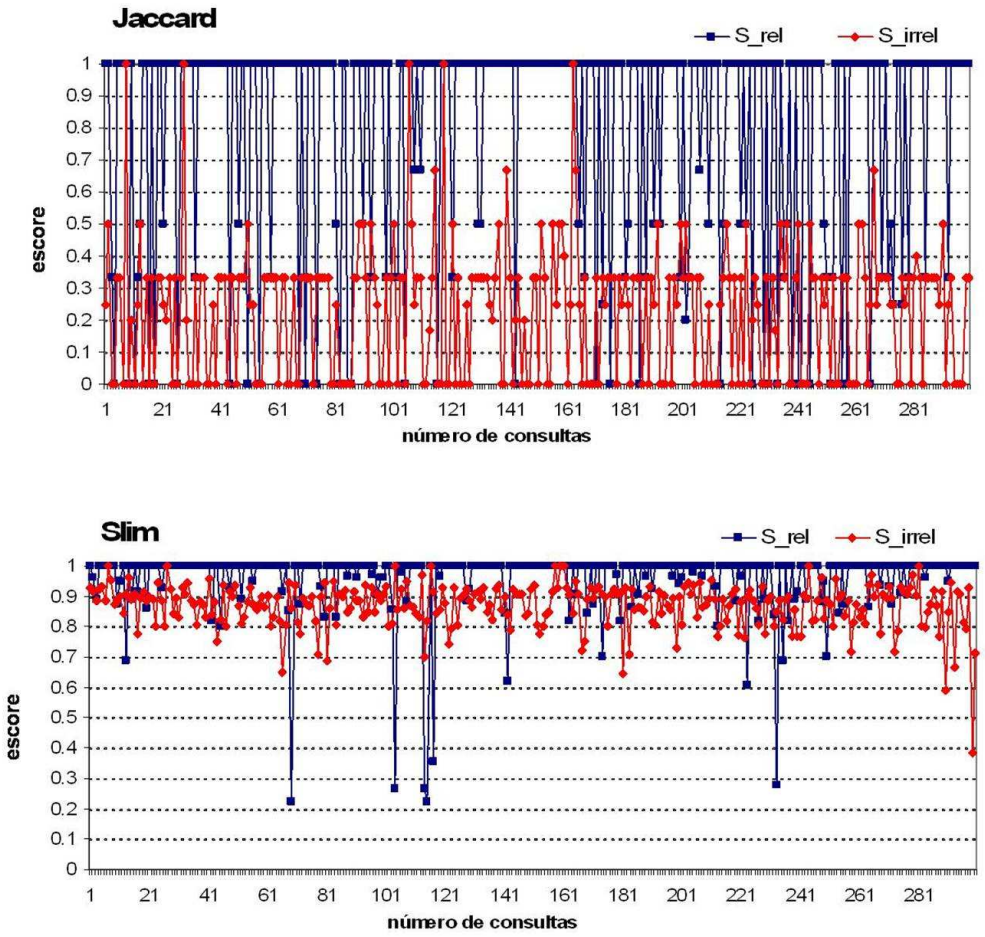

Figura 5.14: Representação gráfica de S_{rel} e S_{irrel}.

A Figura 5.14 mostra a distribuição de S_{rel} (dots) e S_{irrel} (squares) para duas funções de similaridade, Slim e Jaccard, usando a coleção de `Bairros`. Analisando o gráfico apresentado na Figura 5.14, que a separação entre relevantes e irrelevantes obtida pela função de similaridade Jaccard é melhor que a separação obtida pela função Slim, i.é., o intervalo entre relevantes e irrelevantes é maior usando a função Jaccard. Este fato é corroborado com o valor obtido pelo cálculo da *Discernibilidade*, onde Jaccard obteve um valor maior comparado a função Slim (ver linhas 22 e 26 da Tabela 5.4). O valor da precisão média ($MAvP$), contudo, é contraditório com o valor da *Discernibilidade*, pois a função SLIM obteve um valor maior de $MAvP$.

5.5.2 Correlação entre *Discernibilidade* e *MAvP*

Um ponto importante a ser considerado é calcular o grau de concordância entre *Discernibilidade* e *MAvP* quanto à efetividade do resultado produzido pelas funções de similaridade. Para definir esse grau de efetividade, foram produzidos dois *rankings* das funções de similaridade usadas nos experimentos: um ordenado pelo grau de *Discernibilidade* e o outro ordenado por *MAvP*. Para medir a correlação desses dois *rankings*, foram utilizados três métodos diferentes (LEWICKI; HILL, 2006): *Pearson's r*, *Kendall's* τ e *Spearman's* ρ.

Pearson's r é a correlação comum entre pares de valores em um determinado par de *rankings*. Essa medida de correlação considera os valores do *ranking* como variáveis contínuas. *Pearson's r* é a medida comum de correlação linear entre dois conjuntos de variáveis contínuas, medindo o grau de associação entre tais variáveis. *Kendall's* τ e *Spearman's* ρ diferem da primeira por serem testes não-paramétricos, i.é, derivados de uma distribuição não-gaussiana. *Spearman's* ρ é a correlação mais comum de variáveis numéricas (ou um número, ou um intervalo). Diferentemente de *Pearson's r*, *Spearman's* ρ não requer obrigatoriamente que o relacionamento entre as variáveis seja linear, assim como não requer que as variáveis estejam dentro de um mesmo intervalo escalar. Por esta razão, *Spearman's* ρ pode ser usado com o valor de cada instância do *ranking* e não somente atribuindo um valor representando a ordem dos elementos no *ranking*. É importante lembrar que os valores reais representam diretamente a medida de *Discernibilidade* e a *MAvP* em cada *ranking*, e conseqüentemente, representam de forma mais adequada o relacionamento entre as duas medidas, não considerando a distribuição de freqüência das variáveis. Seguindo o mesmo raciocínio, *Kendall's* τ é usado para medir o grau de correspondência entre dois *rankings* e avaliar o significado dessa correspondência.

O cálculo da correlação utilizando os três métodos descritos mostram a avaliação de dois *rankings*. O primeiro *ranking* corresponde à lista das funções de similaridade ordenada por *Discernibilidade* e o segundo, composto pelas mesmas funções, porém a ordem é definida pela *MAvP*. A Tabela 5.11 mostra correlação dos valores obtidos entre os dois *rankings* que representam a ordem das funções de similaridade usando *Discernibilidade* e *MAvP*. Todos os valores resultantes dos métodos de correlação estão dentro do intervalo [-1,1].

Como pode ser observado na Tabela 5.11, os coeficientes de correlação variam, estendendo-se da correlação positiva de 0.791 (`Títulos`) para a correlação negativa de 0.650 (`Bairros`). A Figura 5.15 mostra graficamente os valores de *MAvP* e *Discernibilidade* para cada função de similaridade em cada coleção. Interpretando as curvas apresentadas na Figura 5.15, é possível observar que embora exista uma semelhança, em alguns pontos a diferença entre *MAvP* e *Discernibilidade* é notável. Observando a representação gráfica na Figura 5.15, é possível verificar que as duas medidas são positivamente ou negativamente correlacionadas. Para a coleção `Títulos`, *Slim* e TFIDF têm um aumento significativo enquanto que a *Discernibilidade* decresce para o mesmo intervalo. O mesmo pode ser dito para a coleção de `Instituições` para as funções de similaridade *Jaccard* e *Jaro*.

A *Discernibilidade* é uma medida que avalia aspectos não capturados pela *MAvP*, como a habilidade de uma função de similaridade em separar elementos relevantes e irrelevantes que compõe o resultado de uma consulta. Comparando a *Discernibilidade* com a *MAvP*, a principal diferença é que enquanto a *MAvP* considera somente a ordem de relevância dos elementos no *ranking* produzido por uma

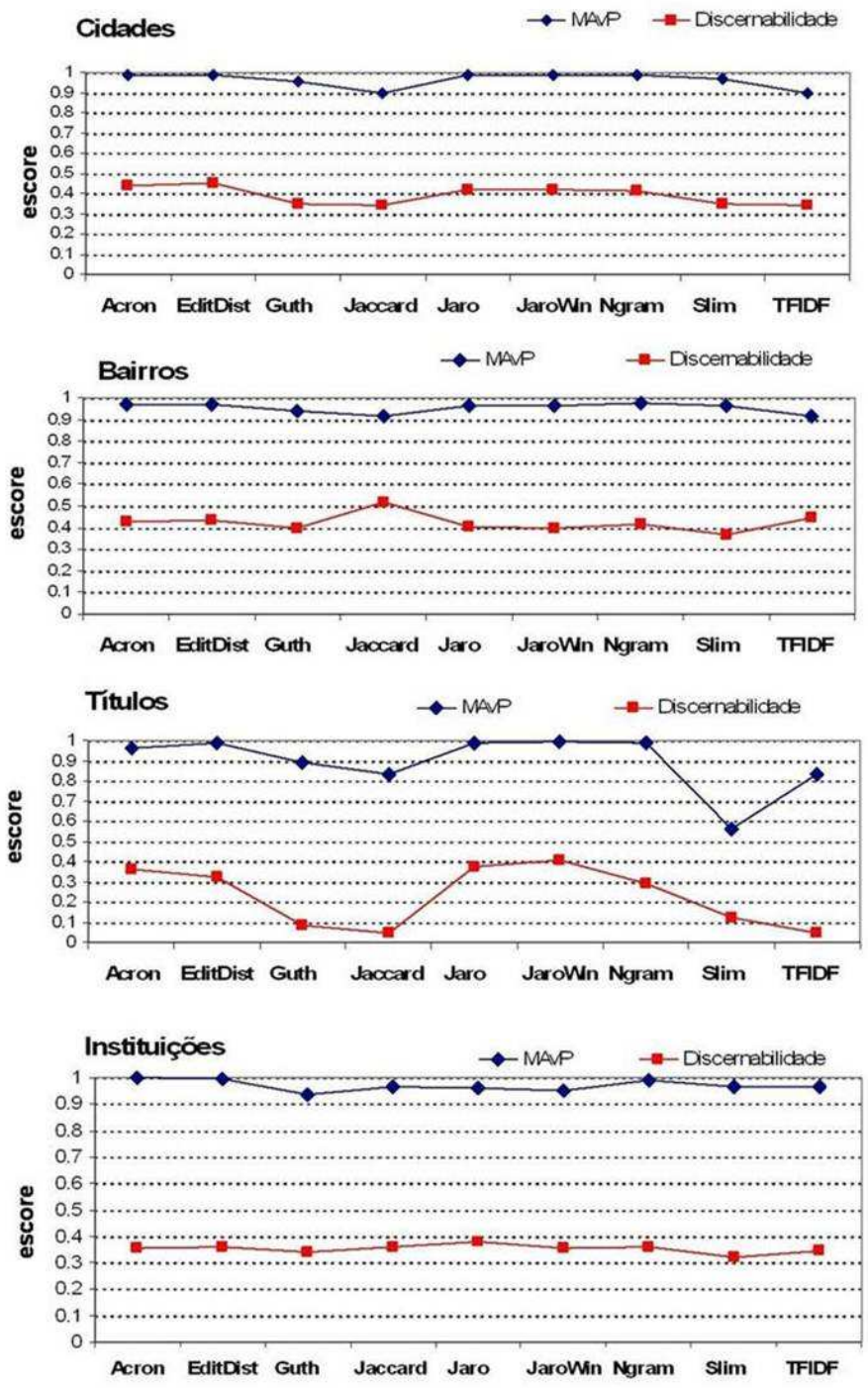

Figura 5.15: Valores para *Discernibilidade* e $MAvP$ para diferentes coleções.

função de similaridade, a *Discernibilidade* considera o valor do escore associado aos elementos do ranking. É importante que observar que o uso do escore no cálculo de avaliação do ranking permite identificar diferenças de qualidade entre as funções de similaridade que não são perceptíveis pelos métodos baseados em R&P.

5.5.3 Considerações sobre a *Discernibilidade*

A *Discernibilidade* mostra a habilidade de uma função de similaridade em separar elementos relevantes dos irrelevantes. Uma função com maior capacidade de discernibilidade permite maior flexibilidade na escolha de um limiar, pois mantém o grupo dos elementos relevantes mais distante dos irrelevantes. Essa propriedade da função de similaridade, embora não seja perceptível pelos métodos baseados em R&P, é uma propriedade importante na avaliação da qualidade de funções de similaridade. Em uma consulta por por abrangência, um `ranking` deve ser composto somente pelos elementos que representam o mesmo objeto consultado. Como as funções de similaridade atribuem escores diferentes de acordo com o algoritmo interno, o fato desta função de similaridade separar com uma distância significativa os elementos relevantes dos irrelevantes é um indício de que a mesma está mais próxima de uma função ideal, que atribuiria 1 para os relevantes e 0 para os irrelevantes.

Comparando a discernibilidade com precisão média, é possível observar que a discernibilidade é mais sensível a certas variações no *ranking*. Pelo fato da precisão considerar somente a ordem em que os elementos relevantes aparecem nas posições mais altas do *ranking*, quando a função de similaridade atribui valores de escore muito próximos aos elementos relevantes e irrelevantes, somente a discernibilidade é afetada. Para consultas vagas por abrangência, a distância entre os valores de s_rel e s_irrel indica um aspecto importante na avaliação da qualidade do *ranking* que não é percebido usando a precisão média.

6 CONCLUSÃO

Esta tese trata do problema de avaliar funções de similaridade no contexto de consultas por abrangência, as quais referem-se ao caso especial das consultas por similaridade que buscam encontrar diferentes representações do mesmo objeto do mundo real. As consultas por abrangência utilizam as funções de similaridade com um limiar, que restringe o resultado, com o objetivo de limitar somente as variações de representação do objeto consultado.

Especificamente com relação às funções de similaridade, os experimentos realizados demonstram que: (i) permitem maior flexibilidade no processamento de consultas por abrangência, pois as coleções provenientes de sistemas reais contém representações diferentes para o mesmo objeto; (ii) precisam de uma avaliação da qualidade do resultado apresentado, pois certamente existem funções de similaridade que são mais apropriadas que outras; e (iii) são dependentes do domínio dos dados, portanto, é possível dizer são imperfeitas na classificação de relevantes e irrelevantes, o que resulta na necessidade de intervenção de um especialista humano.

A aplicação dos resultados obtidos nesta tese é dependente do domínio de aplicação das funções de similaridade utilizadas. A mesma estratégia pode ser aplicada à qualquer domínio, uma vez que o próprio método de estimativa pode facilitar o processo de decisão de qual função de similaridade é mais adequada para a coleção usada. Por exemplo, o uso de funções de similaridade variadas e a conseqüente avaliação da qualidade do resultado obtido, permite uma validação cruzada como forma de selecionar aquelas que obtiverem maior índice de qualidade, ou pelo menos, descartar aquelas que apresentarem resultados inválidos. Um resultado inválido refere-se a sucessivas consultas obtendo a ausência ou totalidade de elementos relevantes, pois uma consulta por abrangência busca por um subconjunto dos objetos armazenados.

Esta tese mostra a aplicação do método semi-automático para estimar R&P em vários limiares, apresentado no Capítulo 3. Comparado ao processo tradicional de estimativa de R&P, pelo fato deste ser altamente dependente do especialista humano, a avaliação da qualidade baseada em R&P seria impraticável em grandes coleções. Já o método apresentado nesta tese permite o uso de grandes coleções com pouca dependência do usuário especialista.

Como forma de permitir o uso de funções de similaridade em grandes coleções e obter a estimativa da qualidade do resultado produzido por tais funções, a primeira contribuição desta tese, apresenta um método para estimar valores de revocação e precisão (R&P) em vários limiares com baixa intervenção do especialista humano. O uso de R&P é amplamente conhecido como medida de avaliação da qualidade na área de Recuperação de Informações. O fato de estimar valores de R&P para vários

limiares oferece em contrapartida um ou mais limiares que são mais adequados para determinada função de similaridade sobre uma coleção. Por esta razão, como uma contribuição adicional, o método de estimativa apresentado permite definir o limiar mais apropriado, uma vez que o resultado esperado em termos de alta revocação ou alta precisão seja previamente conhecido.

Os resultados obtidos comprovam que, se (i) as amostras são representativas da coleção e (ii) os algoritmos de agrupamento estão corretos, as distâncias estatísticas entre os limiares da amostra e da coleção, que foram calculadas pelo método desvio quadrático médio (MSD) apresentado no Capítulo 5, representam valores baixos, próximos de zero. Por esses resultados, entre outros realizados durante o desenvolvimento desta tese conclui-se que é viável aplicar o método de estimativa de R&P sobre a amostra e aplicar tais valores para a coleção. As amostras selecionadas em um processo iterativo da coleção permitem uma análise dos dados armazenados, e a conseqüente avaliação permitem a avaliação de uma coleção, registrando o histórico dos valores estimados, através de um processo de refinamento, onde amostras inadequadas podem ser descartadas.

Embora o método apresentado no Capítulo 3 considere que os grupos formados pelo algoritmo de agrupamento por similaridade estejam corretos, certas coleções podem apresentar particularidades que não são resolvidas pela função de similaridade. Conforme exposto, as funções de similaridade podem atribuir um escore maior para elementos distintos que para elementos que representam o mesmo objeto. Uma estratégia simples que pode ser adotada é introduzir na etapa de agrupamento por similaridade (passo 2b da Figura 3.1) uma confirmação do usuário e a possibilidade de correção dos grupos resultantes. Dessa forma garante-se que os elementos estão agrupados corretamente para que possam ser usados no cálculo de R&P (passo 3 da Figura 3.1). Embora implica em adicionar mais uma intervenção de um especialista humano, pode ser uma alternativa viável dependendo da coleção e da qualidade pretendida, em certos casos somente um especialista humano com conhecimento do domínio poderá decidir se os elementos representam o mesmo objeto ou não.

Uma propriedade importante para determinar a qualidade de uma função de similaridade para o contexto de consultas por abrangência, caracteriza-se pela capacidade da função de similaridade em separar relevantes e irrelevantes. Sem considerar um limiar, o resultado obtido da função de similaridade é um *ranking* de todos os elementos da coleção. Portanto, quanto maior a diferença absoluta entre o menor escore de um elemento relevante e o maior escore de um elemento irrelevante, maior é a flexibilidade para determinar o limiar mais apropriado. É equivalente afirmar que tal flexibilidade aumenta a capacidade da função de similaridade de discernir entre elementos relevantes e irrelevantes. Por esta razão, a medida de qualidade apresentada nesta tese foi denominada discernibilidade.

Os resultados obtidos com a discernibilidade mostram que a habilidade de separar os relevantes e irrelevantes é uma propriedade importante e significativa para avaliar a qualidade da função de similaridade aplicadas ao contexto de consultas por abrangência. Comparando a discernibilidade com medidas tradicionais baseadas em R&P, é possível observar que esta última favorece os elementos no topo do *ranking*, o que nem sempre representa maior qualidade da função de similaridade quando se trata de consultas por abrangência. É intuitivo que quanto mais separados os elementos relevantes dos irrelevantes, bem como, a separação mutuamente exclusiva dos dois conjuntos, permita maior flexibilidade na escolha do limiar, mantendo so-

mente os relevantes no conjunto resposta. A ausência desta flexibilidade no limiar pode apresentar um conjunto de resposta vazio, mesmo tendo dados que poderiam atender à consulta em certas aplicações. Por esta razão, a discernibilidade captura variações nas consultas por abrangência que não são perceptíveis pelas medidas baseadas em precisão, como os experimentos comparando discernibilidade e precisão média apresentam na Seção 5.5.

Concentrando-se em estratégias de avaliação da qualidade de funções de similaridade, esta tese permitiu uma investigação aprofundada do uso de funções de similaridade aplicada no contexto de consultas por abrangência. Embora o tema consulta por similaridade tenha uma vasta bibliografia, os métodos de avaliação da qualidade estão voltados para consultas por quantidade. Portanto, estudo voltado ao problema de avalia a qualidade no contexto das consultas por abrangência é significativo para o estado da arte atual. Dessa forma, é possível resumir as três principais contribuições resultantes desta tese na:

1. definição de um método semi-automático para estimar R&P em vários limiares, com baixa intervenção do especialista humano;

2. introdução da discernibilidade, como uma medida mais apropriada que a precisão média para avaliar a qualidade de funções de similaridade para consultas por abrangência;

3. estimativa do limiar "ótimo", que pode ser obtida pelo processo de estimativa baseado em R&P ou pelo intervalo de entrada do método de cálculo da discernibilidade.

6.1 Publicações

O desenvolvimento desta tese resultou em publicações que estão agrupadas de agrupadas de acordo com o tipo de veículo de publicação. Para cada publicação, segue-se um breve comentário relacionando à qual parte ou fase da tese esta publicação se refere.

Artigo publicado em Periódico Internacional – O artigo abaixo apresenta a medida de discernibilidade. Apresenta os métodos para a estimativa do limiar através das duas abordagens: a análise bivariada e o algoritmo. Em seguida, apresenta o cálculo da similaridade corroborado com o resultado dos experimentos. Os fundamentos descritos neste artigo encontram-se descritos no Capítulo 4 desta tese.
Measuring quality of similarity functions in approximate data matching
JOI'07 - Journal of Informetrics (Elsevier)
Roberto da Silva, Raquel Stasiu, Viviane Moreira Orengo, Carlos A. Heuser

Artigo submetido para Conferência Internacional – Através da avaliação da qualidade de funções funções de similaridade através da discernibilidade sobre diferentes coleções, este artigo compara os resultados obtidos com a precisão média. Os resultados demonstram o quanto a discernibilidade é mais sensível que a precisão média para capturar variações na qualidade apresentada pela função de similaridade. Alguns exemplos foram deste comparativo foram

apresentados no Capítulo 5, na Seção 5.5.1 desta tese, entretanto, o artigo contempla resultados estendidos para outras funções de similaridade.
Comparing methods for the evaluation of similarity functions
CIKM'07 - Lisboa, Portugal
Viviane Moreira Orengo, Carlos A. Heuser, Roberto da Silva, Raquel Stasiu

Artigo publicado em Conferência Internacional – Este artigo descreve o método semi-automático de estimativa de R&P para vários limiares com pouca intervenção do especialista humano. Refere-se ao Capítulo 3, descrito nesta tese.
Estimating recall and precision for imprecise queries in databases
CAISE'05 - Porto, Portugal
Raquel Stasiu, Carlos A. Heuser, Roberto da Silva

Artigo publicado em Conferência Nacional – Estes dois artigos descrevem diferente estágios de evolução da ferramenta desenvolvida como trabalho de conclusão de curso, que implementa o método descrito no Capítulo 3.
FERP: Ferramenta para Estimativa de Revocação e Precisão
SBBD'05 - Sessão Demo - Brasilia, DF
Juliana Bonato, Raquel Stasiu , Carlos Heuser

Ferramenta para estimativa deRecall/Precision usando amostras do Banco de Dados
ERBD'05 - Escola Regional de BD - Porto Alegre, RS
Juliana Bonato dos Santos, Raquel Stasiu, Carlos A. Heuser

Outros – Este artigo não está relacionado ao contexto desta tese. Foi desenvolvido junto ao grupo de banco de dados e recuperação de informações da Universidade de Toronto, sob a supervisão do Professor Mariano Consens, durante o programa de estágio no exterior (doutorado sanduíche). Embora não esteja relacionado ao tema desta tese, tem importância no sentido de que permitiu compreender o funcionamento de consultas por quantidade (*top-k queries*) e delinear o escopo desta tese no contexto de consultas por abrangência.
Efficient, Effective and Flexible XML Retrieval Using Summaries
INEX'06 - Dagstuhl, Alemanha
M. S. Ali, Mariano Consens, Xin Gu, Yaron Kanza, Flavio Rizzolo, Raquel Stasiu

6.2 Trabalhos futuros

Como trabalhos futuros, podem ser elencadas as seguintes propostas:

6.2.1 Aplicar o método de estimativa com técnicas de aprendizado

Em certos casos, dependendo da aplicação, somente o conhecimento do domínio e do contexto é que permite identificar elementos relevantes e irrelevantes. Isto conduz à conclusão de que eliminar totalmente a intervenção do especialista humano com a garantia da qualidade desejada nem sempre será possível. O que pode ser feito é utilizar o método descrito no Capítulo 3 sobre várias amostras, obtendo um perfil da coleção através dos resultados obtidos. Tais resultados constituem um

histórico que pode ser usado como um treinamento para um sistema especialista. Um trabalho futuro pode ser o desenvolvimento do sistema especialista de forma que, utilizando técnicas de aprendizado e refinamentos sucessivos, minimize cada vez mais a intervenção humana, e o sistema especialista possa agregar o conhecimento do especialista humano sobre um determinado domínio.

6.2.2 Explorar aplicação de outros algoritmos de agrupamento por similaridade

Um outro aspecto que pode ser explorado em continuidade ao trabalho desenvolvido nesta tese é o estudo mais aprofundado a respeito de outros algoritmos de agrupamento. Optou-se por utilizar a abordagem aglomerativa, porém outros métodos como os divisivos, ou mesmo inovações poderiam trazer benefícios maiores. Pelos experimentos realizados, os resultados obtidos com os aglomerativos foram satisfatórios e por esta razão este método foi escolhido. Todas esses dados podem ser usados para testar os limites do método, de forma que possa ser determinado um perfil de qual função é mais adequada para qual coleção. Este perfil pode ser determinado pela identificação das propriedades que a coleção precisa ter para aplicar determinada função de similaridade, ou vice-versa.

6.2.3 Análise das características dos grupos formados

Outro aspecto, ainda relacionado ao processo de agrupamento refere-se à investigação da qualidade dos grupos formados visando a remoção total de intervenção do especialista humano. Características como entropia, a distribuição dos valores, informações sobre o domínio, inclusive aspectos semânticos obtidos na comparação do resultado várias funções de similaridade, aprendizado sobre dados históricos, entre outros, podem permitir que seja minimizada ainda mais a interação humana no método apresentado no Capítulo 3.

6.2.4 Calcular os valores de relevantes e irrelevantes de forma semi-automática para discernibilidade

Durante os experimentos realizados para demonstrar a aplicação da *Discernibilidade*, a identificação dos elementos relevantes foi feita manualmente. Um especialista humano com conhecimento do domínio identificou os valores para o menor escore de um elemento relevante e o maior escore para o irrelevante. Entretanto, uma forma de automatizar o processo poderia ser a adaptação do método de estimativa de R&P descrito no Capítulo 3, para tornar o processo de identificação de relevantes e irrelevantes semi-automático, reduzindo a interação com o especialista humano. A combinação do processo de amostragem e do agrupamento por similaridade mostrou ser viável para estimar R&P e pode ser definido um trabalho futuro a definição de um processo semelhante para o cálculo da discernibilidade, como mostrado na Figura 3.1, do Capítulo 3.

6.2.5 Implementar uma ferramenta de avaliação da qualidade

De certa forma, pode-se dizer que uma ferramenta que combine uma avaliação da qualidade de consulta por similaridade em geral poderia ser extremamente útil. Portanto, constitui uma possibilidade de extensão a combinação dos métodos de avaliação para consultas por quantidade e por abrangência, integrando as medias

de R&P e discernibilidade. Com vantagens, poderiam ser citadas a comparação dos resultados obtidos entre as duas medidas de avaliação da qualidade, de forma semi-automática, pode-se ter uma análise mais completa, traçando um perfil das funções de similaridade e da coleção.

6.2.6 Aplicar a avaliação da qualidade sobre o resultado de operadores por similaridade

As consultas por abrangência permitem o processamento de consultas no estilo de consultas sobre banco de dados, o que implica na execução parcial da consulta através de operadores algébricos. Com o uso de funções de similaridade nos operadores como seleção ou junção, por exemplo, uma atividade que pode ser explorada como continuidade desta tese é a otimização do plano consulta considerado aspectos qualitativos e não somente o custo. Novos algoritmos de otimização podem considerar a qualidade da função de similaridade para decidir sobre qual implementação de operador físico usar, uma vez que o critério de menor número de tuplas pode inserir um fator de erro sobre os demais operadores do plano de consulta.

6.2.7 Explorar a análise estatística combinado as funções de similaridade e a coleção

Intuitivamente, é possível obter propriedades onde cada combinação de funções de similaridade específicas para o domínio são apropriadas para uma determinada coleção. Através da análise estatística, tais propriedades podem representar importantes informações sobre os dados. O resultado do processamento estatístico prévio da coleção permite registrar metadados para serem utilizados durante o processamento da consulta. Adicionalmente, tais propriedades permitem testar os valores limites que podem ser usados, mantendo o mesmo nível de qualidade no resultado da consulta.

REFERÊNCIAS

ALDENDERFER, M. S.; BLASHFIELD, R. K. **Cluster Analysis**. Beverly Hills: Sage, 1984. n.07-044. (Sage University Paper Series on Quantitative Applications in the Social Science, n.07-044).

ALI, M.; CONSENS, M. P.; GU, X.; KANZA, Y.; RIZZOLO, F.; STASIU, R. Efficient, Effective and Flexible XML Retrieval Using Summaries. In: INTERNATIONAL WORKSHOP OF THE INITIATIVE FOR THE EVALUATION OF XML RETRIEVAL,INEX, 5., 2006, Schloss Dagstuhl, Germany. **Proceedings...** [S.l.: s.n.], 2006. p.93–108.

ANANTHAKRISHNA, R.; CHAUDHURI, S.; GANTI, V. Eliminating Fuzzy Duplicates in Data Warehouses. In: VLDB, 2002, San Fransisco, CA. **Proceedings...** San Fransisco(CA): Morgan Kaufmann, 2002. p.586–597.

ARANTES, A. S.; VIEIRA, M. R.; JR., C. T.; TRAINA, A. J. M. Operadores de Seleção por Similaridade para Sistemas de Gerenciamento de Bases de Dados Relacionais. In: SIMPÓSIO BRASILEIRO DE BANCOS DE DADOS, SBBD, 18., 2003. **Anais...** Belo Horizonte: Departamento de Ciência da Computação/UFMG, 2003. p.341–355.

ARANTES, A. S.; VIEIRA, M. R.; JR., C. T.; TRAINA, A. J. M. Efficient Algorithms to Execute Complex Similarity Queries in RDBMS. **Journal of The Brazilian Computer Society - JCBS**, Porto Alegre, RS, v.9, n.3, p.5–24, 2004.

BAEZA-YATES, R.; RIBEIRO-NETO, B. **Modern Information Retrieval**. New York: Addison Wesley, 1999.

BARTA, A.; CONSENS, M. P.; MENDELZON, A. O. Benefits of path summaries in an XML query optimizer supporting multiple access methods. In: INTERNATIONAL CONFERENCE ON VERY LARGE DATA BASES, 31., 2005, New York, NY, USA. **Proceedings...** New York: ACM Press, 2005. p.133–144.

BENJELLOUN, O.; GARCIA-MOLINA, H.; KAWAI, H.; LARSON, T. E.; MENESTRINA, D.; SU, Q.; THAVISOMBOON, S.; WIDOM, J. Generic Entity Resolution in the SERF Project. **IEEE Data Eng. Bull.**, [S.l.], v.29, n.2, p.13–20, 2006.

BILENKO, M.; MOONEY, R.; COHEN, W.; RAVIKUMAR, P.; FIENBERG, S. Adaptive Name Matching in Information Integration. **IEEE Intelligent Systems**, [S.l.], v.18, n.5, p.16–23, Sept. 2003.

BILENKO, M.; MOONEY, R. J. Adaptive duplicate detection using learnable string similarity measures. In: INTERNATIONAL CONFERENCE ON KNOWLEDGE DISCOVERY AND DATA MINING, ACM SIGKDD, 9., 2003, New York, NY, USA. **Proceedings...** New York: ACM Press, 2003. p.39–48.

BIMBO, A. D. **Visual Information Retrieval**. San Francisco: Morgan Kaufmann, 1999.

BONATO, J.; STASIU, R.; HEUSER, C. FERP: ferramenta para estimativa de revocaćão e precisão. In: SIMPÓSIO BRASILEIRO DE BANCOS DE DADOS, SBBD, 20., 2005, Uberlândia, MG, Brasil. **Sessão de Ferramentas.** Uberlândia: UFU, 2005. CD-ROM.

BONCZ, P. A. **Monet**: a next-generation DBMS kernel for query-intensive applications. 2002. Ph.D. Thesis — Universiteit van Amsterdam, Amsterdam, The Netherlands.

BROU, D. D.; OLSEN, M. The Guth algorithm and the nominal record linkage of multi-ethnic populations. **Historical Methods Newsletter**, [S.l.], v.19, n.1, p.20–24, 1986.

CHAUDHURI, S.; GANJAM, K.; GANTI, V.; MOTWANI, R. Robust and efficient fuzzy match for online data cleaning. In: ACM SIGMOD INTERNATIONAL CONFERENCE ON MANAGEMENT OF DATA,ACM SIGMOD, 22., 2003, New York, NY, USA. **Proceedings...** New York: ACM Press, 2003. p.313–324.

CHAUDHURI, S.; GANTI, V.; KAUSHIK, R. A Primitive Operator for Similarity Joins in Data Cleaning. In: INTERNATIONAL CONFERENCE ON DATA ENGINEERING,ICDE, 22., 2006, Washington, DC, USA. **Proceedings...** Los Alamitos: IEEE Computer Society, 2006. p.5.

CHINENYANGA, T. T.; KUSHMERICK, N. An expressive and efficient language for XML information retrieval. **J. Am. Soc. Inf. Sci. Technol.**, New York, NY, USA, v.53, n.6, p.438–453, 2002.

COHEN, W. W. Integration of heterogeneous databases without common domains using queries based on textual similarity. In: ACM SIGMOD INTERNATIONAL CONFERENCE ON MANAGEMENT OF DATA, SIGMOD, 1998, New York, NY, USA. **Proceedings...** New York: ACM Press, 1998. p.201–212.

COHEN, W. W. Data integration using similarity joins and a word-based information representation language. **ACM Trans. Inf. Syst.**, New York, NY, USA, v.18, n.3, p.288–321, 2000.

COHEN, W. W.; KAUTZ, H. A.; MCALLESTER, D. A. Hardening soft information sources. In: INTERNATIONAL CONFERENCE ON KNOWLEDGE DISCOVERY AND DATA MINING, ACM SIGKDD, 6., 2000, Boston, MA, USA. **Proceedings...** New York: ACM Press, 2000. p.255–259.

COHEN, W. W.; RAVIKUMAR, P.; FIENBERG, S. A comparison of string distance metrics for name-matching tasks. In: WORKSHOP ON INFORMATION INTEGRATION ON THE WEB - IIWEB, IJCAI, 2003, 2003. **Proceedings...** San Francisco(CA): Morgan Kaufmann, 2003. p.73–78.

COHEN, W. W.; RICHMAN, J. Learning to match and cluster large high-dimensional data sets for data integration. In: INTERNATIONAL CONFERENCE ON KNOWLEDGE DISCOVERY AND DATA MINING, ACM SIGKDD, 8., 2002, Edmonton, Alberta, Canada. **Proceedings...** New York: ACM Press, 2002. p.475–480.

CONSENS, M. P.; MILO, T. Algebras for querying text regions: expressive power and optimization. **J. Comput. Syst. Sci.**, [S.l.], v.57, n.3, p.272–288, 1998.

DOAN, A.; LU, Y.; LEE, Y.; HAN, J. Profile-Based Object Matching for Information Integration. **IEEE Intelligent Systems**, [S.l.], v.18, n.5, p.54–59, 2003.

DORNELES, C. **Uma Estratégia Genérica para Casamento Aproximado de Instâncias**. 2006. Tese (Doutorado em Ciência da Computação) — Instituto de Informática, (UFRGS - Universidade Federal do Rio Grande do Sul), Porto Alegre, RS.

DORNELES, C. F.; LIMA, A. E. N.; HEUSER, C. A.; SILVA, A. da; MOURA, E. Measuring Similarity between Collection of Values. In: ACM INTERNATIONAL WORKSHOP ON WEB INFORMATION AND DATA MANAGEMENT, WIDM, 6., 2004, New York, NY, USA. **Proceedings...** Washington: ACM Press, 2004. p.56 – 63.

ELMAGARMID, A. K.; IPEIROTIS, P. G.; VERYKIOS, V. S. Duplicate Record Detection: a survey. **IEEE Transactions on Knowledge and Data Engineering**, Los Alamitos, CA, USA, v.19, n.1, p.1–16, 2007.

FAGIN, R.; LOTEM, A.; NAOR, M. Optimal aggregation algorithms for middleware. In: SYMPOSIUM ON PRINCIPLES OF DATABASE SYSTEMS,ACM SIGACT-SIGMOD-SIGART, 12., 2001. **Proceedings...** New York: ACM Press, 2001. p.102–113.

FALOUTSOS, C. **Searching Multimedia Databases by Content**. Norwell, MA, USA: Kluwer Academic Publishers, 1996.

FELLEGI, I. P.; SUNTER, A. B. A theory for record linkage. **Journal of the American Statistical Society**, [S.l.], v.64, p.1183–1210, 1969.

FLICKNER, M.; SAWHNEY, H.; NIBLACK, W.; ASHLEY, J.; HUANG, Q.; DOM, B.; GORKANI, M.; HAFNER, J.; LEE, D.; PETKOVIC, D.; STEELE, D.; YANKER, P. Query by Image and Video Content: the QBIC system. **Computer**, Los Alamitos, CA, USA, v.28, n.9, p.23–32, 1995.

GALLAIRE, H.; MINKER, J.; NICOLAS, J.-M. Logic and Databases: a deductive approach. **ACM Comput. Surv.**, New York, NY, USA, v.16, n.2, p.153–185, 1984.

GAO, L.; WANG, M.; WANG, X. S.; PADMANABHAN, S. Expressing and Optimizing Similarity-Based Queries in SQL. In: INTERNATIONAL CONFERENCE ON CONCEPTUAL MODELING, ER, 23., 2004. **Proceedings...** Berlin: Springer-Verlag, 2004. p.464–478. (Lecture Notes in Computer Science, v.3288).

GARCIA-MOLINA, H. Pair-Wise entity resolution: overview and challenges. In: ACM INTERNATIONAL CONFERENCE ON INFORMATION AND KNOWLEDGE MANAGEMENT,CIKM, 15., 2006. **Proceedings...** New York: ACM Press, 2006. p.1–1.

GRAVANO, L.; IPEIROTIS, P. G.; JAGADISH, H. V.; KOUDAS, N.; MUTHUKRISHNAN, S.; PIETARINEN, L.; SRIVASTAVA, D. Using q-grams in a DBMS for Approximate String Processing. **IEEE Data Engineering Bulletin**, [S.l.], v.24, n.4, p.28–34, 2001.

GRAVANO, L.; IPEIROTIS, P. G.; JAGADISH, H. V.; KOUDAS, N.; MUTHUKRISHNAN, S.; SRIVASTAVA, D. Approximate String Joins in a Database (Almost) for Free. In: INTERNATIONAL CONFERENCE ON VERY LARGE DATA BASES,VLDB, 27., 2001, San Francisco, CA, USA. **Proceedings...** San Fransisco(CA): Morgan Kaufmann, 2001. p.491–500.

GRAVANO, L.; IPEIROTIS, P. G.; KOUDAS, N.; SRIVASTAVA, D. Text joins in an RDBMS for web data integration. In: INTERNATIONAL CONFERENCE ON WORLD WIDE WEB, 12., 2003. **Proceedings...** New York: ACM Press, 2003. p.90–101.

GUERRA, M. J.; DONAIRE, D. **Estatística Indutiva**: teoria e exercícios. São Paulo: LTCE, 1944.

GUHA, S.; KOUDAS, N.; MARATHE, A.; SRIVASTAVA, D. Merging the Results of Approximate Match Operations. In: INTERNATIONAL CONFERENCE ON VERY LARGE DATA BASES,VLDB, 30., 2004. **Proceedings...** San Francisco(CA): Morgan Kaufmann, 2004. p.636–647.

GUTH, G. J. Surname spellings and computerized record linkage. **Historical Methods Newsletter**, [S.l.], v.10, n.1, p.10–19, 1976.

HALKIDI, M.; BATISTAKIS, Y.; VAZIRGIANNIS, M. Clustering Algorithms and Validity Measures. In: INTERNATIONAL CONFERENCE ON SCIENTIFIC AND STATISTICAL DATABASE MANAGEMENT, 13., 2001, Washington, DC, USA. **Proceedings...** Washington: IEEE Computer Society, 2001. p.3–22.

HALL, P. A. V.; DOWLING, G. R. Approximate String Matching. **ACM Comput. Surv.**, New York, NY, USA, v.12, n.4, p.381–402, 1980.

HARTIGAN, J. A. **Clustering Algorithms**. New York, NY, USA: John Wiley and Sons, 1975.

HAVELIWALA, T. H.; GIONIS, A.; KLEIN, D.; INDYK, P. Evaluating Strategies for Similarity Search on the Web. In: INTERNATIONAL WORLD WIDE WEB CONFERENCE, HONOLULU, HAWAII, USA, 11., 2002. **Proceedings...** New York: ACM Press, 2002.

HERNANDEZ, M. A.; STOLFO, S. J. The merge/purge problem for large databases. In: INTERNATIONAL CONFERENCE ON MANAGEMENT OF DATA, ACM SIGMOD, 14., 1995. **Proceedings...** New York: ACM Press, 1995. p.127–138.

HIRATA, K.; KATO, T. Query by Visual Example - Content based Image Retrieval. In: INTERNATIONAL CONFERENCE ON EXTENDING DATABASE TECHNOLOGY, EDBT, 3., 1992, London, UK. **Proceedings...** Berlin: Springer-Verlag, 1992. p.56–71. (Lecture Notes in Computer Science, v.580).

ILYAS, I. F.; AREF, W. G.; ELMAGARMID, A. K. Supporting Top-k Join Queries in Relational Databases. In: INTERNATIONAL CONFERENCE ON VERY LARGE DATA BASES, VLDB, 29., 2003. **Proceedings...** St. Louis:MO: Morgan Kaufmann, 2003. p.754–765.

JACCARD, P. The Distribution of the Flora in the Alpine Zone. **New Phytologist**, [S.l.], v.11, n.2, p.37–50, Feb. 1912.

JAIN, A. K.; MURTY, M. N.; FLYNN, P. J. Data clustering: a review. **ACM Comput. Surv.**, [S.l.], v.31, n.3, p.264–323, 1999.

JARO, M. A. Advances in Record-Linkage Methodology as Applied to Matching the 1985 Census of Tampa-Florida. **Journal of the American Statistical Association**, [S.l.], v.84, n.406, p.414–420, June 1989.

KOUDAS, N.; SARAWAGI, S.; SRIVASTAVA, D. Record linkage: similarity measures and algorithms. In: MANAGEMENT OF DATA, ACM SIGMOD, 25., 2006, New York, NY, USA. **Proceedings...** New York: ACM Press, 2006. p.802–803.

LEVENSHTEIN, V. I. Binary Codes Capable of Correcting Deletions, Insertions and Reversals. **Soviet Physics Doklady**, [S.l.], v.10, p.707–727, Feb. 1966.

LEWICKI, P.; HILL, T. **STATISTICS Methods and Applications**. Tulsa, OK: StatSoft, 2006. Disponível em: <http://www.statsoft.com/textbook/stathome.html>. Acesso em: 20 jul. 2007.

LI, C.; CHANG, K. C.-C.; ILYAS, I. F.; SONG, S. **Query Algebra and Optimization for Relational Top-k Queries**. [S.l.]: University of Illinois at Urbana-Champaign - Computer Science Department, 2004. 17 p.

LI, C.; JIN, L.; MEHROTRA, S. Supporting Efficient Record Linkage for Large Data Sets Using Mapping Techniques. **World Wide Web**, [S.l.], v.9, n.4, p.557–584, 2006.

LIMA, A. E. N. **Pequisa de Similaridade em XML**. 2002. 81 p., Projeto de Diplomação (Curso de Ciência da Computação) – Instituto de Informática, UFRGS, Porto Alegre.

LIST, J.; MIHAJLOVIC, V.; VRIES, A. P. de; RAMIREZ, G.; HIEMSTRA, D. The TIJAH XML-IR system at INEX. In: INITIATIVE ON THE EVALUATION OF XML RETRIEVAL (INEX), 2., 2003. **Proceedings...** ERCIM Workshop Proceedings, 2003.

LIST, J.; MIHAJLOVIC, V.; VRIES, A. P. de; RAMIREZ, G.; HIEMSTRA, D.; BLOK, H. E. TIJAH: embracing ir methods in xml databases. **Information Retrieval Journal**, [S.l.], v.8, n.4, p.547 – 570, Dec. 2005.

MCCALLUM, A.; NIGAM, K.; UNGAR, L. H. Efficient clustering of high-dimensional data sets with application to reference matching. In: KNOWLEDGE DISCOVERY AND DATA MINING, ACM SIGKDD, 6., 2000, Boston, MA, USA. **Proceedings...** New York:ACM Press, 2000. p.169–178.

MENESTRINA, D.; BENJELLOUN, O.; GARCIA-MOLINA, H. Generic Entity Resolution with Data Confidences. In: CLEANDB, 2006, New York, NY, USA. **Proceedings...** New York: ACM Press, 2006.

MONGE, A. E.; ELKAN, C. An Efficient Domain-Independent Algorithm for Detecting Approximately Duplicate Database Records. In: WORKSHOP ON RESEARCH ISSUES ON DATA MINING AND KNOWLEDGE DISCOVERY, DMKD, 1997. **Proceedings...** New York: ACM Press, 1997.

MOTRO, A. VAGUE: a user interface to relational databases that permits vague queries. **ACM Trans. Inf. Syst.**, New York, NY, USA, v.6, n.3, p.187–214, 1988.

NAMBIAR, U.; KAMBHAMPATI, S. Answering imprecise database queries: a novel approach. In: ACM INTERNATIONAL WORKSHOP ON WEB INFORMATION AND DATA MANAGEMENT, 2003. **Proceedings...** New York: ACM Press, 2003. p.126–133.

NAVARRO, G. A Guided Tour to Approximate String Matching. **ACM Computing Surveys**, [S.l.], v.33, n.1, p.31–88, Mar. 2001.

ORTEGA-BINDERBERGER, M. **Integrating Similarity Based Retrieval and Query Refinement in Databases**. 2002. PhD Thesis — UIUC - University of Illinois at Urbana-Champaign, Urbana, Illinois.

RAHM, E.; DO, H.-H. Data Cleaning: problems and current approaches. **IEEE Bulletin of the Technical Committee on Data Engineering**, [S.l.], v.23, n.4, Dec. 2000.

ROBERTSON, S. E.; WALKER, S. Some simple effective approximations to the 2-Poisson model for probabilistic weighted retrieval. In: ACM SIGIR CONFERENCE ON RESEARCH AND DEVELOPMENT IN INFORMATION RETRIEVAL, SIGIR, 17., 1994. **Proceedings...** London: UK: Springer-Verlag, 1994. p.232–241.

SALTON, G. **Automatic text processing**: the transformation, analysis, and retrieval of information by computer. Boston, MA, USA: Addison-Wesley Longman Publishing, 1989.

SALTON, G.; LESK, M. E. Computer Evaluation of Indexing and Text Processing. **J. ACM**, New York, NY, USA, v.15, n.1, p.8–36, 1968.

SALTON, G.; MCGILL, M. **Introduction to Modern Information Retrieval**. New York, NY, USA: McGraw-Hill, 1983.

SARAWAGI, S.; BHAMIDIPATY, A. Interactive deduplication using active learning. In: INTERNATIONAL CONFERENCE ON KNOWLEDGE DISCOVERY AND DATA MINING, ACM SIGKDD, 8., 2002, New York, NY, USA. **Proceedings...** New York:ACM Press, 2002. p.269–278.

SARAWAGI, S.; BHAMIDIPATY, A. Interactive deduplication using active learning. In: KNOWLEDGE DISCOVERY AND DATA MINING ACM SIGKDD, 8., 2002, New York, NY, USA. **Proceedings...** New York: ACM Press, 2002. p.269–278.

SCHALLEHN, E.; SATTLER, K.-U. Using Similarity-Based Operations for Resolving Data-Level Conflicts. In: BRITISH NATIONAL CONFERENCE ON DATABASES - NEW HORIZONS IN INFORMATION MANAGEMENT, BNCOD, 20., 2003. **Proceedings...** Berlin: Springer-Verlag, 2003. p.172–189. (Lecture Notes in Computer Science, v.2712).

SCHALLEHN, E.; SATTLER, K.-U.; SAAKE, G. Efficient similarity-based operations for data integration. **Data Knowl. Eng.**, [S.l.], v.48, n.3, p.361–387, 2004.

SIBSON, R. SLINK: an optimally efficient algorithm for the single-link cluster method. **The Computer Journal**, [S.l.], v.16, n.1, p.30–34, 1973.

SPIEGEL, M. **Theory and Problems of Probability and Statistics**. New York, NY, USA: McGraw-Hill, 1992.

STASIU, R. K.; HEUSER, C. A.; SILVA, R. **Estimating recall and precision for imprecise queries in databases**. Porto Alegre,RS: Instituto de Informática/UFRGS, 2004. Disponível em: <http://metropole.inf.ufrgs.br/ raquel/phd/TR348.pdf>. Acesso em: 20 dez. 2005. (TR348).

TEJADA, S.; KNOBLOCK, C. A.; MINTON, S. Learning object identification rules for information integration. **Information System**, Oxford, UK, v.26, n.8, p.607–633, 2001.

THEOBALD, M.; SCHENKEL, R.; WEIKUM, G. An efficient and versatile query engine for TopX search. In: VERY LARGE DATA BASES, VLDB, 31., 2005. **Proceedings...** San Francisco(CA): Morgan Kaufmann, 2005. p.625–636.

ULLMAN, J. D.; GARCIA-MOLINA, H.; WIDOM, J. **Database Systems**: the complete book. Upper Saddle River, New Jersey, USA: Prentice Hall, 2002.

WINKLER, W. E. **The state of record linkage and current research problems.** [S.l.]: Statistical Research Division - U.S. Bureau of the Census, 1999. (R99/04).

WIVES, L. K. **Utilizando conceitos como descritores de textos para o processo de identificação de conglomerados (clustering) de documentos**. 2004. Tese (Doutorado em Ciência da Computação) — Instituto de Informática, UFRGS, Porto Alegre, RS.

ZHONG, Q.; LAZARIDIS, I.; DESHPANDE, M.; LI, C.; MEHROTRA, S.; STERN, H. Supporting Approximate Similarity Queries with Quality Guarantees in P2P Systems. In: INTERNATIONAL CONFERENCE ON MANAGEMENT OF DATA, COMAD, 13., 2006, Delhi, India. **Proceedings...** [S.l.: s.n.], 2006. Disponível em: <http://www.cse.iitb.ac.in/comad/2006/proceedings/60.pdf>. Acesso em: 20 jun. 2007.