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

MARCOS FREITAS NUNES

Avaliação Experimental de uma Técnica de Padronização de Escores de Similaridade

Dissertação apresentada como requisito parcial
para a obtenção do grau de
Mestre em Ciência da Computação

Prof. Dr. Carlos Alberto Heuser
Orientador

Porto Alegre, Dezembro de 2009

CIP – CATALOGAÇÃO NA PUBLICAÇÃO

Nunes, Marcos Freitas

Avaliação Experimental de uma Técnica de Padronização de Escores de Similaridade / Marcos Freitas Nunes. – Porto Alegre: PPGC da UFRGS, 2009.

72 f.: il.

Dissertação (mestrado) – Universidade Federal do Rio Grande do Sul. Programa de Pós-Graduação em Computação, Porto Alegre, BR–RS, 2009. Orientador: Carlos Alberto Heuser.

1. Consulta por Similaridade. 2. Integração de Dados. 3. Data Cleaning. 4. Casamento Aproximado de Registros. 5. Escore Ajustado. 6. Qualidade de Dados. I. Heuser, Carlos Alberto. II. Título.

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
Reitor: Prof. Carlos Alexandre Netto
Pró-Reitor de Coordenação Acadêmica: Prof. Rui Vicente Oppermann
Pró-Reitor de Pós-Graduação: Prof. Aldo Bolten Lucion
Diretor do Instituto de Informática: Prof. Flávio Rech Wagner
Coordenador do PPGC: Prof. Álvaro Freitas Moreira
Bibliotecária-chefe do Instituto de Informática: Beatriz Regina Bastos Haro

“Você não pode arriscar o que você não tem e você não pode ter o que não veio do risco.”

— DAVE WEINBAUM

AGRADECIMENTOS

Inicialmente agradeço aos meus pais, Otávio Pureza Nunes e Vera Nívea Freitas Nunes que são a minha maior estrutura. Agradeço todo o apoio, amor, carinho e compreensão, pois sem eles não estaria onde estou hoje.

A minha namorada Helena Bento Bosenbecker pelo amor, carinho, incentivo e paciência no decorrer destes sete anos. Por compreender a distância que foi necessária para que eu pudesse cursar o mestrado e por me apoiar emocionalmente nos momentos difíceis, durante a nossa caminhada.

A minha irmã pelo apoio, carinho e torcida que mesmo a distância foi prestado.

Aos meus colegas Adrovane Kade, Alexander Vinson, Daniel Litchnow, Edimar Manica, Eduardo Borges, Euler de Oliveira, Gabriel Simões, Giseli Lopes, Guilherme Bianco, Gustavo Piltcher, Mariusa Warpechowski, Maurício Dias, Otávio Acosta, Sérgio Mergen e Thyago Borges, que foram ótimos companheiros de estudo, churrasco e cerveja. Em especial, ao Eduardo e a Giseli pelas discutições sobre o trabalho e por me auxiliarem na correção do texto e ao Euler, Gustavo, Mauricio e Otávio pela companhia nas disciplinas.

Ao professor Carlos Alberto Heuser pela orientação, confiança e paciência disponibilizadas até mesmo antes do inicio deste trabalho.

Aos professores da UFRGS pelo conhecimento passado. Em especial a professora Viviane, pelas discutições sobre o trabalho, idéias para aperfeiçoá-lo, pelo auxílio na definição dos experimentos e métodos para avaliá-los.

Aos membros do GPSI - Grupo de Pesquisa em Sistemas de Informação, grupo a qual participei na graduação e que foi o início da minha vida acadêmica.

Por fim, agradeço a UFRGS, e ao instituto de informática pela infraestrutura disponibilizada. Ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) e a Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (Capes), pelo apoio financeiro, sem o qual o desenvolvimento deste trabalho seria muito mais difícil.

SUMÁRIO

LISTA DE ABREVIATURAS E SIGLAS

ACM	Association for Computing Machinery
CCSB	Collection of Computer Science Bibliographies
CEP	Código de Endereçamento Postal
DBLP	Digital Bibliography & Library Project
Febrl	Freely Extensible Biomedical Record Linkage
IDF	Inverse Document Frequency
IEEE	Institute of Electrical and Electronics Engineers
IR	Information Retrieval
RI	Recuperação de Informação
TaME	Tabela de Mapeamento de Escores
TF-IDF	Term Frequency - Inverse Document Frequency
XML	Extensible Markup Language

LISTA DE FIGURAS

LISTA DE TABELAS

RESUMO

Com o crescimento e a facilidade de acesso a Internet, o volume de dados cresceu muito nos últimos anos e, consequentemente, ficou muito fácil o acesso a bases de dados remotas, permitindo integrar dados fisicamente distantes. Geralmente, instâncias de um mesmo objeto no mundo real, originadas de bases distintas, apresentam diferenças na representação de seus valores, ou seja, os mesmos dados no mundo real podem ser representados de formas diferentes. Neste contexto, surgiram os estudos sobre casamento aproximado utilizando funções de similaridade. Por consequência, surgiu a dificuldade de entender os resultados das funções e selecionar limiares ideais. Quando se trata de casamento de agregados (registros), existe o problema de combinar os escores de similaridade, pois funções distintas possuem distribuições diferentes.

Com objetivo de contornar este problema, foi desenvolvida em um trabalho anterior uma técnica de padronização de escores, que propõe substituir o escore calculado pela função de similaridade por um escore ajustado (calculado através de um treinamento), o qual é intuitivo para o usuário e pode ser combinado no processo de casamento de registros. Tal técnica foi desenvolvida por uma aluna de doutorado do grupo de Banco de Dados da UFRGS e será chamada aqui de *MeaningScore* (DORNELES et al., 2007).

O presente trabalho visa estudar e realizar uma avaliação experimental detalhada da técnica *MeaningScore*. Com o final do processo de avaliação aqui executado, é possível afirmar que a utilização da abordagem *MeaningScore* é válida e retorna melhores resultados. No processo de casamento de registros, onde escores de similaridades distintos devem ser combinados, a utilização deste escore padronizado ao invés do escore original, retornado pela função de similaridade, produz resultados com maior qualidade.

Palavras-chave: Consulta por Similaridade, Integração de Dados, Data Cleaning, Casamento Aproximado de Registros, Escore Ajustado, Qualidade de Dados.

Experimental Evaluation of a Similarity Score Standardization Technique

ABSTRACT

With the growth of the Web, the volume of information grew considerably over the past years, and consequently, the access to remote databases became easier, which allows the integration of distributed information. Usually, instances of the same object in the real world, originated from distinct databases, present differences in the representation of their values, which means that the same information can be represented in different ways. In this context, research on approximate matching using similarity functions arises. As a consequence, there is a need to understand the result of the functions and to select ideal thresholds. Also, when matching records, there is the problem of combining the similarity scores, since distinct functions have different distributions.

With the purpose of overcoming this problem, a previous work developed a technique that standardizes the scores, by replacing the computed score by an adjusted score (computed through a training), which is more intuitive for the user and can be combined in the process of record matching. This work was developed by a Phd student from the UFRGS database research group, and is referred to as *MeaningScore* (DORNELES et al., 2007).

The present work intends to study and perform an experimental evaluation of this technique. As the validation shows, it is possible to say that the usage of the *MeaningScore* approach is valid and return better results. In the process of record matching, where distinct similarity must be combined, the usage of the adjusted score produces results with higher quality.

Keywords: Similarity Querying, Data Integration, Data Cleaning, Record Matching, Adjusted Score, Data Quality.

1 INTRODUÇÃO

O problema do casamento aproximado ocorre em inúmeros contextos, tais como: consulta por similaridade (BUENO; TRAINA; TRAINA, 2005), integração de dados (BILENKO et al., 2003) e data cleaning (CHAUDHURI et al., 2003). Em consulta por similaridade, o problema é identificar registros do banco de dados que representam o mesmo objeto no mundo real. Em integração de dados, o problema é determinar os registros de diferentes bases de dados que representam o mesmo objeto no mundo real. O problema em *data cleaning* é a detecção e correção de erros nos dados recebidos de fontes externas em um *data warehouse*. Tais dados podem conter erros, que devem ser resolvidos antes do armazenamento do dado no *data warehouse*, como: erros de ortografia, convenções inconsistentes através de fontes de dados e campos ausentes. Tais situações são totalmente comuns em bases de dados modernas, especialmente nos casos em que se recebem dados e consultas de fontes diferentes.

Exemplos reais de dados diferentes que representam o mesmo objeto no mundo real são encontrados em bibliotecas digitais como a BDBComp (LAENDER; GONÇALVES; ROBERTO, 2004), DBLP [1] e a ACM [2]. Nestas bases, campos como nome de autor, por exemplo, podem aparecer com duplicações, pois normalmente são representados nos artigos de maneiras diferentes. Por exemplo, o nome "Manoel Silva da Silva" pode aparecer como "Silva, Manoel da Silva", ou ainda "Silva, M.S.". Este tipo de situação, onde registros distintos se referem ao mesmo objeto, gera um grande problema. Como não é possível garantir que os dois campos se referem ao mesmo objeto, a solução é utilizar um mecanismo que forneça um valor de proximidade entre os dois campos. Este mecanismo é chamado de casamento aproximado.

Em casamento aproximado, mais especificamente em similaridade de consultas, já existem várias técnicas que podem ser utilizadas para comparar similaridade em campos atômicos. Campos atômicos são valores únicos, como pequenas sequências de caracteres, por exemplo, duas *strings* contendo valores textuais, como nome de pessoas, endereço, instituições, datas, etc. Para esses tipos de campos existem muitas funções como *Levenshtein* (também conhecida como Edit Distance) (LEVENSHTEIN, 1966), Jaccard, Jaro (JARO, 1989), JaroWinkler (WINKLER, 1990, 1999), etc. Tais funções estão implementadas e disponíveis em algumas bibliotecas como, por exemplo, a *SimMetrics* (CHAPMAN, 2009) desenvolvida para Java e C# .NET.

As funções de similaridade ou algoritmos de similaridade $f(a_1, a_2) \mapsto s$ calculam quanto uma *string* a_1 é similar a outra a_2. Tais algoritmos determinam um escore s para cada par de valores de dados (*strings*). Escores altos significam similaridade alta. Na

[1] `http://www.informatik.uni-trier.de/~ley/db/`
[2] `http://portal.acm.org/`

maioria das abordagens, dois objetos são considerados similares, ou seja, são considerados os mesmo objetos no mundo real, se o escore de similaridade ultrapassar um limiar predefinido.

A qualidade do processo de casamento não depende apenas da capacidade da função de similaridade em separar valores relevantes de irrelevantes, mas também do limiar que é escolhido.

Porém, a escolha do valor do limiar é uma tarefa muito difícil. Os escores retornados pela função dependem de detalhes internos do algoritmo que implementa a função e geralmente não tem significado para o usuário, com exceção do fato que um alto valor significa que dois objetos são mais similares. Por isso, não se pode esperar que o limiar fornecido pelo usuário seja exato. Provavelmente, o limiar será de alguma maneira, predefinido por uma estimação ou por um processo de aprendizagem (STASIU; HEUSER; SILVA, 2005). Além disso, os valores de escore retornados por funções distintas possuem distribuições diferentes, então a qualidade do resultado pode variar de uma função para outra quando um limiar específico é considerado. A distribuição também pode variar quando a mesma função é aplicada a dois conjuntos de dados de domínios distintos. Isto significa que, um valor de limiar que tenha sido predefinido para uma função específica, pode não ser adequado para outra, ou mesmo um valor de limiar que tenha sido predefinido para uma função x utilizando o domínio y pode não ser adequado para a mesma função no domínio z.

Outro problema relacionado ao uso de diferentes funções de similaridade ocorre quando os objetos a serem casados são compostos de objetos, ou seja, são estruturas aninhadas, como tuplas de bases de dados ou árvores XML. Este problema é encontrado em casamento de registros, onde o desafio é identificar e casar registros que representam o mesmo objeto no mundo real. Neste cenário, é importante combinar corretamente os escores de similaridade de cada componente (atributo), para estimar a similaridade entre o objeto composto (como por exemplo, uma tupla). Existem vários trabalhos que reconhecem este problema e propõem técnicas para resolvê-los (CHEN; FARAHAT; BRANTS, 2004; FERGUSON; BRIDGE, 1999; MOTRO, 1988; TEJADA; KNOBLOCK; MINTON, 2001). Em alguns casos, estas soluções são acompanhadas do uso de alguma técnica de transformação antes da aplicação da função de similaridade (TEJADA; KNOBLOCK; MINTON, 2001). No entanto, como funções diferentes geram valores de escore que não são comparáveis, não existe uma maneira simples para combinar funções distintas em uma única medida.

Na figura 1.1 são apresentados dois gráficos obtidos através da execução de vários casamentos utilizando as funções JaroWinkler e Q-GramSimilarity sobre um conjunto de strings contendo, respectivamente, títulos de periódicos (*journals*) e títulos de filmes. Cada gráfico plota a média da precisão obtida em um escore de similaridade específico.

Analisando estes gráficos, é possível observar que a precisão em um escore específico (por exemplo, o escore 0,5) pode variar entre as duas funções de similaridade. Para o conjunto de dados de títulos de periódicos, utilizando a função JaroWinkler, a precisão resultante para o limiar 0,5 é muito próxima de zero, enquanto que, para o conjunto de filmes, utilizando a função Q-GramSimilarity, a precisão para o mesmo limiar fica em torno de 0,6. Este exemplo ilustra dois problemas em aberto em casamento aproximado: (i) os valores retornados por uma função de similaridade não tem um significado claro para o usuário, e (ii) estes valores não são comparáveis, quando os resultados de diferentes funções de similaridade são combinados, como por exemplo, em casamento de registros (campos complexos).

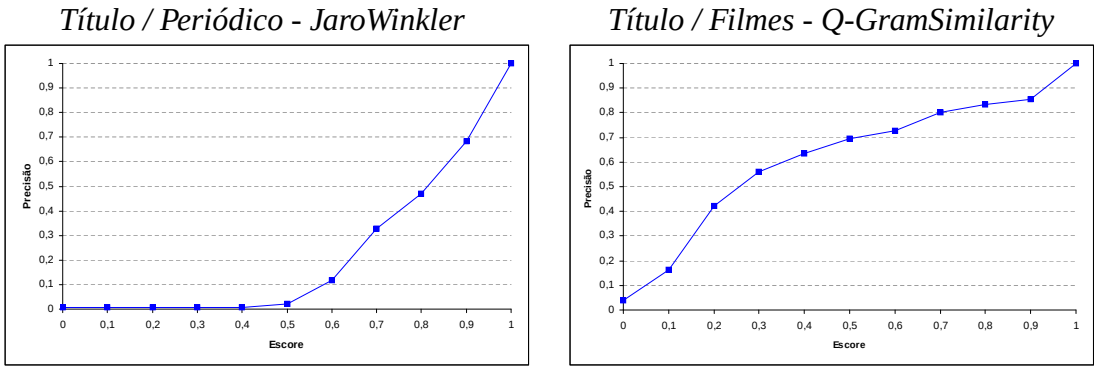

Figura 1.1: Exemplo da diferença de distribuição entre funções de similaridade. Precisão em vários pontos de escore de similaridade para um conjunto de consultas utilizando as funções de similaridade JaroWinkler e Q-GramSimilarity

Técnicas para similaridade de campos complexos também são encontradas na literatura, porém com menor quantidade em comparação com as de campos atômicos. Geralmente, algoritmos para campos complexos utilizam um conjunto de técnicas de valores atômicos, combinando seus escores resultantes para que se possa ter apenas um valor de similaridade para o campo complexo. Por exemplo, precisa-se saber a similaridade entre duas tuplas, T_1 e T_2, cada uma dessas tuplas possui dados de uma pessoa, como nome, endereço e telefone. A maior parte das técnicas de similaridade de registros conhecidas (campos complexos) utilizam uma função de similaridade para cada atributo, como por exemplo, uma para nome, outra para endereço e outra para telefone, e então fazem a combinação desses valores para obter um valor de similaridade entre as tuplas, enquanto outras concatenam todos os campos em apenas um (formando apenas uma *string*) e verificam a similaridade da *string* inteira. Porém, Tejada, Knoblock e Minton (2001) dizem que concatenar campos e verificar a similaridade como uma *string* apenas gera imprecisão. Estudos sobre esse assunto são encontrados em Tejada, Knoblock e Minton (2001), Elmagarmid, Ipeirotis e Verykios (2007) e Guha et al. (2004).

A técnica de escore ajustado (*MeaningScore*), estudada e apresentada neste trabalho (Capítulo 3), desenvolvida em um trabalho anterior por uma aluna de doutorado do grupo de Banco de Dados da UFRGS (DORNELES et al., 2007), aborda os problemas descritos anteriormente (escores retornados pelas funções de similaridade não são significativos para o usuário e não são combináveis em casamento de registros). Tal técnica propõe o uso de um novo escore de similaridade, o escore ajustado, ao invés dos escores originalmente gerados pelas funções. A ideia é uma abordagem baseada em um processo de treinamento, durante o qual a precisão é estimada para cada função de similaridade e para diferentes conjuntos de dados. O resultado desta fase é uma tabela que mapeia escores de similaridade para valores de precisão, com determinada função sobre um conjunto de dados, permitindo assim que, durante o processo de casamento, o usuário ao invés de especificar um limiar, especifique a precisão esperada dos resultados. A ideia desta abordagem é que a precisão estimada expressa a convicção de um valor, de um dado atributo, ser uma representação diferente do mesmo atributo. Portanto, o escore ajustado pode ser utilizado como uma função que não é apenas única para qualquer tipo de atributo comparado, mas que também fornece um resultado que é significativo ao usuário, permitindo que seja expresso em consultas como um parâmetro que define o grau de proximidade permitida em um processo de casamento aproximado.

Além de o escore ajustado tratar o problema da escolha de um limiar ideal, ele também pode ser visto como um escore universal (escore padronizado), o qual permite combinar os resultados de funções de similaridade distintas, adequadas para cada atributo, com intuito de avaliar a similaridade entre registros (estruturas aninhadas em geral).

No entanto, após a elaboração desta abordagem (escore ajustado) em um trabalho anterior, alguns pontos ficaram em aberto, como a avaliação experimental da abordagem. Foi feita uma avaliação inicial, que, entretanto, necessitava ser aprofundada.

Com isso, o presente trabalho visa estudar e validar experimentalmente a abordagem de escore ajustado, tratando os pontos que ficaram em aberto no trabalho anterior. Estes pontos são: (i) avaliar experimentalmente a utilização do escore ajustado, (ii) analisar o comportamento do escore ajustado utilizando diferentes funções de similaridade com a mesma base de dados, (iii) utilizar o escore ajustado no processo de casamento de registros, onde se faz necessária a combinação dos resultados de diferentes funções de similaridade, e (iv) durante todo o processo do cálculo do escore ajustado, a precisão é calculada em 11 pontos de escore ($[0, 0.1, \ldots, 0.9, 1]$), então outro ponto em aberto é testar se a utilização dos 11 pontos de precisão é o suficiente, ou se com a utilização de mais pontos os resultados melhoram.

O restante do texto está organizado da seguinte forma. No capítulo 2 são discutidos trabalhos relacionados ao problema de casamento aproximado. É interessante observar que os trabalhos que foram encontrados na literatura não possuem o mesmo objetivo do escore ajustado, mas podem fazer uso do mesmo para tentar melhorar seus resultados. No capítulo 3 é apresentada a abordagem de escore ajustado, aqui estudada, descrevendo detalhadamente o mecanismo proposto para ajustar (padronizar) o escore retornado por uma função de similaridade. O capítulo 4 é dividido em duas grandes áreas. A primeira validando a abordagem *MeaningScore* (apresentada no capítulo 3), onde são apresentadas avaliações experimentais com bases de dados reais e representativas, as quais mostram os resultados de *MeaningScore* com o treinamento realizado na mesma base de dados do processo de casamento (base de dados real), e o resultado de *MeaningScore* com o treinamento realizado em outra base de dados do mesmo domínio (base representativa). A segunda grande área é a de casamento de registros, onde os experimentos demonstram que a qualidade das abordagens de casamento de registros, baseadas na combinação dos escores de similaridade dos atributos, é melhorada quando o *MeaningScore* é utilizado. Por fim, no capítulo 5, são descritas as considerações finais, conclusões e são apresentadas possibilidades de trabalhos futuros.

2 TRABALHOS RELACIONADOS

Este capítulo apresenta o contexto no qual a dissertação está inserida: casamento aproximado. Como este trabalho é experimental e a abordagem utilizada foi desenvolvida em trabalhos anteriores, o estudo aprofundado sobre trabalhos relacionados foi realizado anteriormente. Neste capítulo, será feita uma breve contextualização da área de casamento aproximado, onde os experimentos aqui realizados estão inseridos.

No contexto de Bancos de Dados, o casamento aproximado constitui uma operação essencial para solucionar problemas relacionados a processamento de consultas aproximadas, resolução de entidades e *data cleaning* (limpeza de dados).

Em processamento de consultas aproximadas, o objetivo é fornecer respostas adequadas para consultas mal especificadas, nas quais o usuário possui um conhecimento limitado sobre os dados ou sobre o esquema, ou quando preferências pessoais específicas devem ser cumpridas. Neste caso, ao invés de exigir que os valores nas tuplas casem exatamente com o argumento consulta, o casamento aproximado é utilizado (BRUNO; CHAUDHURI; GRAVANO, 2002; DALVI; SUCIU, 2004; KOUDAS; MARATHE; SRIVASTAVA, 2004; MOTRO, 1988). Para essa finalidade, funções de similaridade baseadas em caracteres (como por exemplo, *edit distance* e *Q-GramSimilarity*), tokens (como exemplo *Jaccard distance* e TF-IDF), fonética (por exemplo, a função *soundex distance*) ou semânticas (*ontological distance*) são utilizadas para comparar os valores e calcular um escore de similaridade, o qual permite avaliar o casamento. Em particular, o casamento aproximado tem sido utilizado no contexto de operadores de junção (GRAVANO et al., 2003; GUHA et al., 2006). O grau de similaridade entre os argumentos das consultas e os valores no banco de dados é usualmente restringido por um limiar especificado pelo usuário. Especificar um limiar adequado é uma tarefa difícil, pois diversas funções de similaridade podem ser utilizadas em uma mesma consulta. Para especificar este limiar, o usuário deve ter conhecimento do domínio dos dados utilizados, a sua distribuição e também conhecer bem as funções de similaridade.

Técnicas de casamento aproximado têm sido amplamente utilizadas há vários anos para integração de dados, provenientes de fontes distintas, em uma operação conhecida como resolução de entidade (*entity resolution*), deduplicação de dados (*data deduplication*) ou casamento de registros (*record linkage*).

Um dos trabalhos pioneiros na área de casamento de registros foi proposto por Fellegi e Sunter (1969). Neste, a similaridade entre os registros de dois arquivos distintos é dada pela média dos escores de similaridade entre pares de atributos comuns destes arquivos. Os autores formalizaram uma solução para o problema de reconhecer registros duplicados (pessoas, objetos, ou eventos idênticos) em dois arquivos de dados diferentes através de um modelo matemático. O modelo proposto exige a definição de dois valores de limiar. Uma função de similaridade, denominada *linkage rule*, é aplicada a um par de registros

e o valor retornado é comparado com os dois limiares definidos anteriormente. Se o valor de similaridade for maior do que ambos os limiares, os registros são considerados duplicatas, já se o valor de similaridade for menor do que ambos os limiares, os registros são confirmados como diferentes, ou seja, não são réplicas. Os registros ainda podem ser classificados como possíveis casamentos quando o valor de similaridade retornado pela função estiver entre os dois limiares. Neste caso, é necessária a intervenção humana para julgar a similaridade.

Ultimamente, como diferentes funções de similaridade são utilizadas para tipos de atributos e domínios distintos, esta simples estratégia teve que ser revisada. O trabalho de Fellegi e Sunter serviu como base para diversas alternativas propostas pela comunidade científica. Então, trabalhos mais recentes propõem diferentes estratégias, as quais, de forma geral, podem ser classificadas sob dois aspectos principais: aquelas que apresentam propostas para combinar os valores de escores dos atributos através de alguma métrica de similaridade (CHAUDHURI et al., 2003; GUHA et al., 2004; DORNELES et al., 2004; CARVALHO; SILVA, 2003; CULOTTA; MCCALLUM, 2005); e aquelas que, além de combinar os escores, propõem métodos baseados em alguma técnica de aprendizado de máquina (BILENKO et al., 2003; DOAN et al., 2003; TEJADA; KNOBLOCK; MINTON, 2001; COHEN; RICHMAN, 2002; CARVALHO et al., 2006; BILENKO; MOONEY, 2003).

A deduplicação de dados é definida por Cohen e Richman (2002) como a seguinte tarefa: a partir de duas listas de nomes de entidades, provenientes de duas fontes distintas, determinar pares de nomes que se referem à mesma entidade do mundo real. Os autores exploram a similaridade textual dos objetos em diferentes bases de dados, propondo uma técnica escalável e adaptativa para agrupar esses objetos. A técnica utiliza um conjunto de treinamento e um algoritmo de aprendizado. O algoritmo realiza o casamento entre pares ou conjuntos de *strings*. A técnica proposta é comparada a outras duas abordagens que utilizam a distância de edição (LEVENSHTEIN, 1966) e *Term Frequency - Inverse Document Frequency* (TFIDF) (BAEZA-YATES; RIBEIRO-NETO, 1999).

Em (BILENKO et al., 2003), os autores experimentam três estratégias diferentes para avaliar a similaridade entre registros: (i) comparar os dois registros como um todo, como se fossem um único atributo; (ii) comparar cada atributo e calcular a média aritmética entre seus escores de similaridade, que é essencialmente o proposto em (FELLEGI; SUNTER, 1969); e (iii) usar um vetor de características (*feature vector*) para representar os atributos e treiná-lo utilizando um classificador SVM (*Support Vector Machine*) (BOSER; GUYON; VAPNIK, 1992). Cada experimento com as três estratégias envolve uma única função de similaridade pré-determinada. Os autores reconhecem que utilizar uma única função para todos os atributos é muito restritivo, então, também propõem uma função de distância de edição adaptável, que pode ser adaptada a determinados tipos de dados textuais. Esta função é semelhante a uma proposta por Ristad e Yianilos (1998). Na abordagem apresentada nesta dissertação, funções de similaridade de qualquer tipo podem ser utilizadas, e não apenas funções de distância de edição sobre atributos textuais.

O Sistema Active Atlas (TEJADA; KNOBLOCK; MINTON, 2001) também utiliza aprendizado de máquina, e tem como objetivo efetuar o mapeamento entre objetos a fim de integrar fontes de dados. Inicialmente, o sistema efetua o cálculo do escore de similaridade para cada atributo de um objeto através de transformações nas cadeias de caracteres e de funções de similaridade específicas para o domínio dos atributos. Após, regras de mapeamento entre os atributos são especificadas a partir de um processo de treinamento, estas regras são geradas através da aplicação de aprendizado com árvores de

decisão (QUINLAN, 1986). A ideia básica é utilizar técnicas da recuperação de informação (similaridade textual) para fornecer um mapeamento inicial e aplicar técnicas de aprendizagem de máquina para melhorar o mapeamento. O Active Atlas necessita obrigatoriamente da participação de um usuário especialista, o qual é requisitado para verificar o mapeamento inicial de alguns pares de objetos. A partir das verificações do especialista, o sistema tenta aprender regras novas e seleciona pares de objetos adicionais para que o especialista verifique novamente. A ideia de utilizar regras ou árvores de decisão também é encontrada em outras abordagens de deduplicação (CHAUDHURI et al., 2007; SARAWAGI; BHAMIDIPATY, 2002).

Um trabalho um pouco mais recente propõe uma abordagem baseada em programação genética para deduplicar objetos (CARVALHO et al., 2006). A programação genética é uma técnica de aprendizado de máquina que da apoio na solução de problemas onde o espaço de busca é muito grande e quando há mais de um objetivo a ser cumprido (BORGES; GALANTE, 2008). Tal abordagem proposta é capaz de gerar funções de similaridade automaticamente para identificar registros duplicados em um dado repositório.

Recentemente, o problema de efetuar casamento entre objetos XML, que é mais complexo que o de tuplas, tem recebido bastante atenção (LEITÃO; CALADO; WEIS, 2007). Leitão, Calado e Weis (2007) utilizam um modelo de rede bayesiana para calcular a probabilidade de dois objetos XML, representados pelo elemento XML, serem duplicatas. É importante salientar que o trabalho apresentado nesta dissertação não tem intenção alguma de propor um novo método para combinar os escores de similaridade dos atributos, como os métodos descritos acima. No entanto, pode ser utilizado juntamente com esses métodos, a fim de melhorar a qualidade dos seus resultados.

Dados recebidos por um *data warehouse* de fontes externas geralmente contém erros. Para tratar isso, trabalhos anteriores propõem o uso de técnicas de limpeza de dados (*data cleaning*) para detectar e eliminar tais erros, assim aumentando a qualidade do *data warehouse* (CHAUDHURI et al., 2003; GUHA et al., 2004). Técnicas existentes podem também incluir detecção de dados duplicados. Em geral, uma técnica comum para efetuar limpeza de dados é validar o registro de entrada com as relações de referência. Essas relações podem ser internas ao *data warehouse*, como, por exemplo, uma das relações que já existe, ou externa, por exemplo, uma tabela de CEP dos correios. Chaudhuri et al. (2003) propõem um algoritmo de casamento *fuzzy* para limpeza de dados. Um registro de entrada é comparado a um conjunto de registros de referência através da função de similaridade proposta. Também é proposto um índice tolerante a erros e um algoritmo probabilístico para recuperar de maneira eficiente os k registros de referência mais similares ao registro de entrada, de acordo com a função de similaridade. Se mais de um candidato ao casamento for retornado, o usuário deve escolher o candidato mais próximo ao registro em questão. Em (GUHA et al., 2004), a abordagem é baseada na combinação de *rankings* de similaridade. *Rankings* individuais são gerados para cada atributo de uma relação, de acordo com uma consulta. Cada *ranking* individual é composto por um conjunto de tuplas formadas por um único atributo e são ordenados em função da similaridade com o atributo correspondente da consulta. É proposta uma função de fusão, denominada *footrule distance*, a qual combina os escores de cada atributo de uma tupla, da relação, considerando também suas posições em cada um dos *rankings*. São selecionados os $top - k$ (CHAUDHURI; GRAVANO, 1999) registros mais relevantes. O objetivo da proposta é derivar um *ranking* final "ótimo" de tamanho k, definido pelo usuário.

Funções de similaridade possuem distribuições de escore diferentes, este fato acarreta em problemas durante o processo de casamento de registros, pois os resultados de dife-

rentes funções, utilizando diferentes atributos, devem ser combinados. Este problema tem sido pesquisado em trabalhos anteriores (GUHA et al., 2004; TEJADA; KNOBLOCK; MINTON, 2001). A maior parte das técnicas de casamento não normalizam os escores calculados por diferentes funções de similaridade, a fim de torná-los comparáveis. No próximo capítulo será apresentada a abordagem de escore ajustado, a qual trata, dentre outros, deste problema.

3 ESCORE AJUSTADO

Nesta seção, é apresentada a proposta de Dorneles et al. (2007), que tem como objetivo mapear diferentes escores de similaridade, gerados por funções de similaridade distintas, para um único escore significativo. A ideia principal consiste em obter, para os escores resultantes de cada função de similaridade, novos valores que sejam significativos para os usuários e que sejam comparáveis com valores providos por outra função. Esta correspondência é dada pela estimativa da qualidade dos escores fornecidos por cada função de similaridade. A estimativa de qualidade é feita medindo a precisão dos resultados fornecidos por uma função. Esta medida é muito utilizada na comunidade de Recuperação de Informação (RI) para avaliar a qualidade de funções de similaridade (BAEZA-YATES; RIBEIRO-NETO, 1999).

Com esta proposta, é possível que o escore seja especificado em uma consulta como um parâmetro que define o grau de proximidade permitido em um processo de casamento aproximado. Além disso, usando o novo escore, tal parâmetro terá sempre o mesmo significado, independentemente de qual função de similaridade está sendo utilizada.

Considerando uma função f que calcula a similaridade entre dois valores a_1 e a_2 de um dado atributo A, tal que $f(a_1, a_2) \mapsto e$ e $0 \leq e \leq 1$. O valor retornado pela função f é chamado aqui de *escore original*. Este conceito é formalizado na Definição 1.

Definição 1 *(Escore Original) Seja D_A um domínio, $f : D_A \times D_A \longrightarrow \mathcal{R}_S \subseteq [0, 1]$ uma função que avalia a similaridade entre um par de valores D_A e retorna um valor entre $[0, 1]$. A função f é chamada de* função de similaridade original *ou simplesmente de* função de similaridade *e qualquer valor $g \in \mathcal{R}_S$ é chamado de* escore original.

Agora, considerando que um valor q de um determinado atributo, em um conjunto de dados, é usado como consulta. Utilizando uma função de similaridade original, é possível ranquear um conjunto de valores de acordo com o escore de similaridade entre cada valor do conjunto e q.

Definição 2 *(Ranking Original) Seja $A \subseteq D_A$ um atributo cujo domínio é D_A e f uma função de similaridade (Definição 1).*

Uma permutação $R(q) = \{a_1, a_2, \ldots, a_n\}$ de A é chamada de ranking original de acordo com $q \in A$ sse $f(q, a_k) \geq f(q, a_{k+1})$ para $k = 1, \ldots, n - 1$.

Intuitivamente, pode-se dizer que a função f estima a probabilidade de dois valores comparados serem sinteticamente equivalentes. A equivalência semântica significa que os dois valores são representações distintas do mesmo objeto do mundo real. Assumindo que f seja consistente, ou seja, os escores retornados estão no intervalo de $[0, \ldots, 1]$, pode-se

dizer que, quanto mais próximo de 1 for o escore original, maior a probabilidade dos valores comparados representarem o mesmo objeto. Por outro lado, quando mais próximo de 0, menor é a probabilidade. Idealmente, escores originais maiores do que 0 deveriam ser associados apenas a valores que representem o mesmo objeto do mundo real. No entanto, funções de similaridade normalmente são imprecisas, não havendo forma de afirmar o quão perto de 1 o escore original deve ser para que dois valores sejam considerados semanticamente iguais. Esta questão está relacionada com a qualidade que uma função de similaridade identifica estas diferentes representações em um domínio de valores.

A precisão de cada função pode ser estimada pela observação dos seus escores calculados em um pequeno conjunto de valores de treinamento. Seguindo um procedimento padrão, comumente adotado em RI, é possível utilizar esse treinamento para estimar a precisão alcançada por cada função, para cada atributo, em cada nível de escore. Este processo consiste em, para cada valor q do conjunto de treinamento, marcar cada a_i no *ranking* $R(q) = \{a_1, a_2, \ldots, a_n\}$ como "relevante", quando a_i representar o mesmo objeto no mundo real que o representado por q, ou como "irrelevante" quando a_i não representar o mesmo objeto no mundo real que q representa. Após este processo, a avaliação da qualidade dos resultados da função de similaridade pode ser especificada. Esta avaliação é feita através do cálculo da precisão (BAEZA-YATES; RIBEIRO-NETO, 1999; CHOWDHURY, 2003) e pode ser calculada como a seguir.

Seja $R(q)_i$ o número de itens no *ranking* $R(q)$ que foram identificados como "relevantes" até a posição i, a precisão é calculada para cada posição i em $R(q)$ pela equação:

$$p_i = R(q)_i / i \tag{3.1}$$

Precisão é utilizada como uma medida de qualidade para funções de similaridade. Os valores de precisão, computados na fase de treinamento, são adotados como estimativas de precisão obtidas pela função de similaridade para qualquer nova comparação de pares de valores do mesmo atributo.

Considerando como exemplo um atributo que contém nomes de bairros, a tabela 3.1 apresenta uma lista de bairros ordenados pelo escore de similaridade de acordo com a comparação com o valor $q =$" bundaperg", utilizando a função de similaridade *JaroWinkler* (qualquer outra função de similaridade poderia ser utilizada para esse exemplo). Esta ordenação é feita de acordo com a Definição 2. A primeira coluna contém a posição (i) do valor a_i no *ranking*, a segunda contém o valor do atributo a_i. A terceira coluna apresenta o escore original g_i gerado pela função *JaroWinkler*. A quarta apresenta o valor de precisão p_i calculado em cada posição i (Equação 3.1) e a última coluna apresenta a avaliação, a qual classifica cada um dos valores a_i como "relevante" ou "irrelevante" de acordo com q. Os valores das posições $i = 1$, 2, 3 e 4 foram considerados como relevantes (ou seja, estes valores representam o mesmo objeto no mundo real que q), então é obtida a precisão $p_i = 1$ para essas posições, já na posição $i = 5$ a precisão cai, indicando uma baixa na qualidade dos resultados obtidos com a função de similaridade.

Os valores de precisão da tabela 3.1 podem ser interpretados como uma forma de quantificar a convicção nos escores gerados pela função de similaridade *JaroWinkler* em avaliar a similaridade entre os elementos neste conjunto de dados, dado um valor q. Mais precisamente, se um valor p_i aparece associado a um escore original g_i, em uma posição i nesta tabela, p_i pode ser interpretado como sendo a probabilidade de um valor a_i ser similar a q se o escore original for considerado maior ou igual a g_i. Por exemplo, na posição $i = 5$ da tabela, o escore original é $g_i = 0,860530317$ e a precisão é $p_i = 4/5$, uma interpretação para isto é que, com o escore original $g \geq 0,860530317$, 80% dos

Pos. (i)	**Nome** (a_i)	**Escore Original** (g_i)	**Precisão** (p_i)	**Avaliação**
1	bundaperg	1	1	relevante
2	bundaerg	0,986666679	1	relevante
3	bundahegrg	0,947609425	1	relevante
4	bundaberg	0,895000041	1	relevante
5	burpehgary	0,860530317	0,8	irrelevante
6	burpenray	0,848333299	0,6666667	irrelevante
7	burpescgary	0,846388876	0,5714286	irrelevante
8	burpeary	0,844814837	0,5	irrelevante
9	burpebgry	0,826666653	0,44444445	irrelevante
10	burwoldeast	0,799444437	0,4	irrelevante
11	burpengary	0,777692318	0,36363637	irrelevante
...	...	...	...	...
405	bacchusanarsh	0,2	0,012376238	irrelevante
406	bacchub narsh	0,2	0,012345679	irrelevante
407	caulfeeald	0,1	0,0123152705	irrelevante
...	...	...	...	...

Tabela 3.1: *Ranking* de nomes de bairros ordenado pelo escore original de acordo com a consulta $q =$*" bundaperg"*

resultados obtidos, serão relevantes.

Estas considerações podem ser generalizadas a quaisquer outras funções de similaridade que geram valores de similaridade consistentes. Ou seja, dada uma função de similaridade f, um valor q, um atributo A e um *ranking* $R(q)$ sobre A de acordo com q, pode-se construir uma tabela similar à 3.1, que possua os mapeamentos dos escores originais para os valores de precisão.

A tabela 3.1 foi construída utilizando um único valor q. Mas, para se ter um melhor entendimento do comportamento de uma função de similaridade, aplicada a um dado conjunto de valores de um atributo, o treinamento deve ser realizado utilizando um conjunto de valores representativos $Q = q_1, q_2, \ldots, q_n$ para gerar um *ranking* $R(q_i)$ para cada valor $q_i \in Q$. Neste caso, é apropriado ter uma única tabela que resuma o comportamento da função para todas as consultas. Assim, um simples mapeamento pode ser gerado para expressar a qualidade da função de similaridade considerando todos $q_i \in Q$.

Esta tabela única pode ser obtida simplesmente através do cálculo da média dos valores de precisão avaliada em cada $q_i \in Q$, conseguidos para o mesmo valor de escore original. A figura 3.1 apresenta três *rankings* resumidos criados como exemplos $R(q_1)$, $R(q_2)$ e $R(q_n)$. A qualidade da função de similaridade utilizada para construção destes *rankings* poderia ser calculada através da média aritmética dos valores de precisão para cada ponto de escore. Por exemplo, para o escore original 1 se obteria uma precisão média 1, para $g_i = 0,7$ se obteria a precisão média de $0,1731$. Porém, como pode ser observado na figura 3.1, os valores de escore original são distintos em cada *ranking*. Como exemplo, pode ser observado que o $R(q_1)$ e $R(q_n)$ possuem $g_i = 0,9$, porém no $R(q_2)$ este escore original não é encontrado. Para tratar esse problema, foi utilizado outro procedimento comum de Recuperação de Informação, que consiste em interpolar os resultados de precisão (BAEZA-YATES; RIBEIRO-NETO, 1999). Primeiramente, é definido um conjunto de valores arbitrários de escores originais, os quais são dispostos como pontos em uma escala dentro de um intervalo. Para os experimentos realizados neste trabalho foram utilizados onze valores $0,0; 0,1; \ldots; 0,9; 1$, como normalmente é utilizado na comunidade de RI. No entanto, um número maior de valores pode ser empregado em casos específicos.

Então, para cada *ranking* $R(q_i)$, será calculado um valor de precisão interpolada em

$R(q_1)$

Valor	Escore Original (g_i)	Precisão (p_i)
st andfes	1	1
st andzws	0,9467	1
st anerxws	0,936969697	1
...	...	...
stagdes	0,9	0,5714286
...	...	...
windsor	0,7	0,1764706
glendale	0,693636358	0,17143
...	...	...

$R(q_2)$

Valor	Escore Original (g_i)	Precisão (p_i)
kaimkillenbun	1	1
armile	0,735714257	0,5
armil	0,721428573	0,33333334
keilr	0,720634937	0,25
keilif	0,7	0,2
...	...	...
wororonaora	0,4393	0,007894737
hurston e park	0,4	0,007874016
...	...	...

$R(q_n)$

Valor	Escore Original (g_i)	Precisão (p_i)
cronullla	1	1
cronnua	0,918560624	1
cronvla	0,918560624	1
cronnnla	0,9	1
cronulla	0,8909	1
...	...	...
charlstonwn	0,7	0,1429
...	...	...

Figura 3.1: Rankings criados para exemplo do cálculo da precisão

cada um dos 11 pontos, baseado nos valores de precisão reais obtidos em tal *ranking*. Este processo é definido formalmente da seguinte forma:

Definição 3 *(Precisão interpolada em um ponto de escore original) Seja $R(q)$ um ranking original de acordo com um valor q (como na Definição 2) e considerando que para cada posição i em $R(q)$ existe um par $\langle g_i, p_i \rangle$, onde g_i representa o escore original em i e p_i representa a precisão nesta mesma posição. Considerando também que g_{MIN} e g_{MAX} são respectivamente o mínimo e o máximo valores de escores originais em $R(q)$. Seja $0 \leq \hat{g} \leq 1$ um valor arbitrariamente predefinido. A precisão interpolada para $\hat{g}$ é dada pela equação:*

$$\dot{p}(\hat{g}, R(q)) = \frac{p_{low}(\hat{g}, R(q)) + p_{high}(\hat{g}, R(q))}{2} \tag{3.2}$$

onde:

- *$p_{low}(\hat{g}, R(q))$ é o menor valor de precisão que ocorre com g_{low} no ranking $R(q)$ e g_{low} é o mais alto valor de escore original no ranking $R(q)$ tal que $\hat{g} \geq g_{low} \geq g_{MIN}$;*
- *$p_{high}(\hat{g}, R(q))$ é o menor valor de precisão que ocorre com g_{high} no ranking $R(q)$ e g_{high} é o mais baixo valor de escore original no ranking $R(q)$ tal que $\hat{g} \leq g_{high} \leq g_{MAX}$.*

A Definição 3 mostra como calcular a precisão interpolada para um valor arbitrário $0 \leq \hat{g} \leq 1$. A intuição por trás desta ideia é que, se $\hat{g}$ é um escore original existente no *ranking*, a precisão interpolada será o mesmo valor de precisão existente para esse $\hat{g}$, por exemplo, no *ranking* da tabela 3.1, a precisão interpolada para $\hat{g} = 0,2$ é $\dot{p} = 0,012345679$. Caso contrário, o valor da precisão interpolada será a média entre os valores de precisão existentes para os escores originais que estejam próximos a $\hat{g}$ (escore original mais alto e mais baixo próximos a $\hat{g}$). Ou seja, a precisão interpolada para $\hat{g} = 0,8$ é $\dot{p} = 0,422222225$, sendo a média entre os valores de precisão das posições

$i = 9$ e $i = 10$. Note que pode haver dois ou mais valores de precisão para um mesmo valor de escore original no *ranking*. Neste caso, o valor mais baixo de precisão é escolhido como precisão interpolada.

Finalmente, a média de todos valores de precisão interpolada é calculada para cada um dos 11 pontos predefinidos sobre todos os *rankings* $R(q_i)$, $q_i \in Q$. Neste trabalho, o termo escore ajustado ou *MeaningScore* são utilizados para referenciar estas médias. Nesta etapa, foi necessário lidar com o problema de não monotonicidade. Valores de precisão interpolada podem não ser monotômicos, por exemplo, em uma consulta específica pode acontecer que a precisão aumente com o escore original diminuindo, ou mais formalmente pode ocorrer que para alguns $0.1 \leq \hat{g} \leq 1$ se tenha $\dot{p}(\hat{g}, R(q)) < \dot{p}(\hat{g} - 0.1, R(q))$. Como se deseja que o escore ajustado seja monotônico, então, ao invés de pegar diretamente o valor da precisão interpolada para um específico escore original $\hat{g}$, se pega a máxima média da precisão interpolada sobre todos os escores originais iguais ou menores que $\hat{g}$. Mais precisamente, essas medias são calculadas como na Definição 4.

Definição 4 *(Tabela de Mapeamento de Escore - TaME) Seja $Q = \{q_1, \ldots, q_m\}, Q \subseteq A$, um conjunto de valores de um dado atributo A, $R(q_k)$ um ranking original gerado de acordo com $q_k \in Q$ (Definição 2). Considere que, para cada posição i de $R(q_k)$, existe um par $\langle g_{i,k}, p_{i,k} \rangle$ o qual representa o escore original e a precisão naquela posição.*

Uma TaME (Tabela de Mapeamento de Escore) é uma lista de pares $\langle \hat{g}_i, \hat{p}_i \rangle$ onde $\hat{g}_i \in \{0.0, 0.1, \ldots, 0.9, 1.0\}$ é um valor de escore representativo predefinido arbitrariamente e $\hat{p}_i$ é a média dos valores de precisão interpolada em $\hat{g}_i$ para cada ranking $R(q_k)$.

$$\hat{p}_i = \max_{j=0}^{i} \left(\frac{\sum_{k=1}^{|Q|} \dot{p}(\hat{g}_j, R(q_k))}{|Q|} \right) \tag{3.3}$$

onde $\dot{p}(\hat{g}_j, R(q_k))$ é a precisão interpolada para $\hat{g}_i$ como na Definição 3.

Cada $\hat{p}_i$ é chamado de valor de escore ajustado.

A tabela 3.2 apresenta um exemplo de tabela de mapeamento de escores construída após a geração de diversos *rankings* sobre a mesma base de dados.

Escore Original	**Escore Ajustado**
1	1
0,9	0,975
0,8	0,958333334
0,7	0,932916669
0,6	0,893041629
0,5	0,655449018
0,4	0,280767591
0,3	0,026283741
0,2	0,004710469
0,1	0,002428322
0,0	0,002365

Tabela 3.2: Um exemplo de uma tabela de mapeamento de escores

Através da tabela de mapeamento de escore (TaME) é possível definir uma função que mapeia o escore original para o escore ajustado. Essa função é definida a seguir.

Definição 5 *(Função de mapeamento do escore original para o escore ajustado) Seja* $M = \{\langle 0.0, p_0\rangle, \ldots, \langle 1.0, p_{10}\rangle\}$ *uma TaME para* f*, como na Definição 4, e um valor de escore original* $0 \leq g \leq 1$*. A função* $\alpha_M(g) \mapsto [0, 1]$ *que mapeia escores originais para escores ajustados é definida como:*

$$\alpha_M(g) = \begin{cases} p_i & \text{se } \langle g, p_i\rangle \text{ ocorre em } M \\ \frac{p_i + p_{i+1}}{2}, \text{ onde } g_i < g < g_{i+1} & \text{caso contrário} \end{cases} \tag{3.4}$$

De acordo com a equação 3.4, se o escore original g estiver presente na TaME, ele é mapeado diretamente para o correspondente escore ajustado, como por exemplo a TaME da figura 3.2, o escore original $0,7$ é mapeado diretamente para o escore ajustado $0,932916669$. No entanto, se o escore original a ser mapeado não pertencer a TaME, como por exemplo $g = 0,73$, é feita à média dos escores ajustados do maior e do menor escore original, no caso da tabela 3.2, a média entre os escore ajustados de $0,8$ e $0,7$, que resulta no valor $0,9456250015$.

Resumindo, para que se possa utilizar a função de mapeamento do escore original para o escore ajustado, em um determinado domínio D, com a utilização de determinada função de similaridade f, é necessário construir uma TaME (Tabela de Mapeamento de Escore) que represente o comportamento da função f no domínio D. Um treinamento deve ser realizado para a construção desta TaME. Este treinamento utiliza um conjunto de valores representativos $Q = \{q_1, q_2, \ldots, q_n\}$, no caso deste trabalho são utilizados 40 valores q distintos. Comparando cada consulta q_i com o conjunto de atributos A do domínio D utilizando a função f, é gerado um *ranking* original $R(q_i)$ para cada q, de acordo com a Definição 2. Então, a precisão é calculada para cada posição dos *rankings* $R(q_i)$, conforme a Equação 3.1. Com a precisão calculada, para cada *ranking* $R(q_i)$ é efetuado o cálculo da precisão interpolada em cada um dos 11 pontos, de acordo com a Definição 3. Como é necessário ter uma única tabela que resuma o comportamento da função f para todas as consultas q_i, a média de todos valores de precisão interpolada é calculada para cada um dos 11 pontos predefinidos sobre todos os *rankings* $R(q_i)$, $q_i \in Q$, gerando assim a TaME que mapeia o escore original para o escore ajustado da função f no domínio D (Definição 4).

Esta função de mapeamento do escore original para o escore ajustado é utilizada nos experimentos apresentados no próximo capítulo.

Como dito anteriormente, a abordagem de escore ajustado, apresentada aqui nesta seção, foi desenvolvida por uma aluna de doutorado da UFRGS. Porém, após a conclusão do seu trabalho ficaram alguns pontos em aberto, que são: (i) avaliar experimentalmente a utilização do escore ajustado, (ii) analisar o comportamento do escore ajustado utilizando diferentes funções de similaridade com a mesma base de dados, (iii) utilizar o escore ajustado no processo de casamento de registros, combinando resultados de diferentes funções e (iv) testar se a utilização de 11 pontos de precisão é o suficiente ou se, com a utilização de mais pontos, os resultados melhoram. Estas questões que ficaram em aberto do trabalho anterior são tratadas nos experimentos apresentados no próximo capítulo.

4 AVALIAÇÃO EXPERIMENTAL

Neste capítulo são descritos os experimentos realizados ao decorrer deste trabalho, com o objetivo de validar experimentalmente a abordagem descrita na seção 3. O objetivo desses experimentos é demonstrar que o escore ajustado pode ser utilizado em substituição ao escore original, calculado pela função de similaridade, permitindo ao usuário especificar a qualidade do processo de casamento através de um valor significativo.

Os experimentos realizados são divididos em 5 seções. O primeiro deles, descrito na seção 4.1, apresenta o processo de casamento de *strings*, mostrando os resultados da utilização do escore ajustado em substituição do escore original. Na seção 4.2 é apresentado um experimento que avalia o comportamento do escore ajustado com diferentes funções de similaridade para o mesmo atributo da base de dados.

Na seção 4.3 são apresentados os experimentos desenvolvidos utilizando o escore ajustado no casamento de registros, onde se faz necessária a combinação dos resultados de diferentes funções para se obter um único valor de similaridade. Estes experimentos utilizam quatro abordagens de combinação de escore: a primeira delas é a média aritmética, a segunda é a média ponderada, onde se define manualmente pesos para cada atributo do registro, a terceira é a média ponderada utilizando pesos calculados de forma automática e a quarta abordagem é árvore de decisão. A utilização dessas abordagens de combinação de escore tem como único objetivo comparar os resultados obtidos utilizando o escore ajustado e o escore original. Não tendo intenção alguma de propor funções de combinação e nem comparar as forma de combinar resultados.

Na seção 4.4 são apresentados os experimentos realizados com intuito de analisar a influência do tamanho da amostra no processo de casamento, analisando se, com uma amostra maior, os resultados melhoram ou não. E, por último, na seção 4.5, é apresentado um experimento desenvolvido para analisar a granularidade da escala de precisão, ou seja, se o número de pontos de precisão que está se utilizando é realmente suficiente para obter bons resultados, ou se, com o aumento do número de pontos, os resultados ficam mais precisos.

4.1 Precisão Como um Escore Significativo

Como descrito anteriormente, um dos pontos em aberto no trabalho anterior é a avaliação experimental da utilização do escore ajustado. Nesta seção, são apresentados os experimentos que visam realizar esta validação, comparando os resultados obtidos pela utilização direta de funções de similaridades (escore original) com os obtidos pela utilização do escore ajustado.

Como o escore ajustado é uma proposta de padronização do escore original retornado pelas funções de similaridade, no experimento desta seção os resultados do escore original e do ajustado são comparados para que se possa observar se o ajuste realmente melhora os resultados.

Os escores ajustados são estimados pelo processo de treinamento, que é executado a partir do mesmo domínio de dados. Neste experimento, foram avaliadas duas alternativas de bases de dados para o processo de treinamento: (i) usar a mesma base de dados que contém as instâncias a serem casadas (base de dados real), e (ii) utilizar um conjunto grande de valores representando o domínio de dados a serem casados (base de dados representativa). Usar a base de dados real para o treinamento, provavelmente, leva para melhores resultados, dado que a mesma base de dados é utilizada no processo de treinamento e no de casamento. Além disso, se a base de dados passou por uma grande atualização, um novo treinamento é necessário. Já para a segunda opção, base de dados representativa, um único treinamento poderia ser utilizado para qualquer conjunto de atributos do mesmo domínio, porém o escore ajustado pode não ser tão preciso como no caso da base de dados real.

Para ambos casos, o conjunto de consultas Q, usado para o treinamento, consiste em 40 valores distintos, $Q = (q_1, q_2, ..., q_{40})$, correspondentes a objetos do mundo real pertencentes à base de dados. Esses valores são usados para calcular a tabela de mapeamento de escores (TaME), para cada função de similaridade e com cada atributo da base de dados. Isto significa que o comportamento da função de similaridade estudada é estimado com base em um pequeno conjunto de consultas, o qual torna o custo do processo de treinamento viavel. Na seção 4.4, são feitos experimentos variando o número de consultas, com objetivo de analisar quanto o tamanho da amostra influi nos resultados e se 40 é um número aceitável.

4.1.1 Bases de Dados

O experimento foi realizado utilizando dois tipos de dados: dados reais e dados representativos. As bases de dados utilizadas são de três domínios diferentes, onde duas (Citações Bibliográficas e Filmes) foram coletadas em um trabalho anterior e a terceira (Nome) foi gerada neste trabalho.

- **Domínio de Citações Bibliográficas** (*Citation*). Cada registro desta base de dados possui três atributos: o título do artigo, o nome do periódico no qual o artigo foi publicado e a editora do periódico. Os valores deste domínio estão em Inglês e foram extraídos do CCSB[1] (*Collection of Computer Science Bibliographies*) e de arquivos BIBTEX coletados entre os membros do Grupo de Banco de Dados da UFRGS, totalizando 11.898 entradas de citações bibliográficas.

- **Domínio de Filmes** (*Movie*). Este domínio consiste em dados sobre filmes. Cada filme contém três atributos: o título do filme, representado pelo atributo *Título*, o

[1] `http://liinwww.ira.uka.de/bibliography/Database/index.html`

diretor, representado por *Diretor* e o gênero do filme, indicado pelo atributo *Gênero*. Os valores deste domínio estão em inglês e foram originados de múltiplas fontes de dados, como fontes de video locadoras, tais como: *Blockbuster US*, *Blockbuster UK* e *Blockbuster CA*. As instâncias foram extraídas manualmente da *Web* e totalizam 274 entradas de filmes.

- **Domínio de Nome** (*Name*). Esta é uma base de dados artificial, contendo nomes de pessoas gerados pela ferramenta *Febrl* (*Freely Extensible Biomedical Record Linkage*) (CHRISTEN; CHURCHES; HEGLAND, 2004). *Febrl* contém um gerador de dados no qual é possível produzir duplicatas de registros de pessoas com pequenas variações. Os dados produzidos pelo *Febrl* já vem avaliados, ou seja, contém um identificador que marca quais registros são duplicados. A base de dados produzida aqui contém 5.000 *strings* representando 500 nomes de pessoas distintas.

Além do conjunto de dados real, um segundo conjunto de dados também foi construído para cada domínio, chamado de conjunto de dados representativo. É importante salientar que as bases de dados representativas são usadas apenas no processo de treinamento (processo de estimativa do escore ajustado).

As bases de dados representativas possuem os mesmos domínios: Citação, Filmes e Nomes. As duas primeiras coletadas em um trabalho anterior e a terceira gerada durante este trabalho. Os valores para Citações e Filmes foram obtidos principalmente de sites *Web*, a partir de inúmeras fontes distintas. No domínio citações, os valores para os atributos *título*, *periódico* e *editora* foram extraídos da DBLP, Citeseer, sites de periódicos e conferências, além de sites de editoras, como Elsevier, ACM Press, IEEE pub., etc. Para o domínio filmes, os valores para os atributos *título*, *diretor* e *gênero* foram extraídos de várias listas de filmes, com seus títulos originais, na *Web*. Para o domínio nome, uma base de dados diferente, com 5.000 nomes, foi gerada utilizando o *Febrl*.

Na tabela 4.1 são apresentados dados sobre os conjuntos de dados reais; o número total de objetos, o número de objetos distintos (objetos sem redundância) obtidos pela inspeção manual em cada base de dados e o percentual de objetos distintos dividido pelo número total de objetos na base. Já a tabela 4.2 apresenta também os números totais de objetos, número de objetos distintos e o percentual dos distintos em relação ao total, só que para o conjunto de dados representativo.

Domínio	**Atributo**	**Total**	**Objetos Distintos**	**Distintos / Total**
Filmes				
	Título	147	75	51,02%
	Diretor	92	78	84,78%
	Gênero	35	29	82,86%
Citação				
	Título	10910	8416	77,14%
	Periódico	519	254	48,94%
	Editora	469	147	31,34%
Nome				
	Nomes	5000	500	10,00%

Tabela 4.1: Numero total de objetos, número de objetos distintos do total de objetos e percentual de objetos distintos no conjunto de dados reais

Domínio	Atributo	Total	Objetos Distintos	Distintos / Total
Filmes				
	Título	2626	1890	71,97%
	Diretor	10438	6754	64,71%
	Gênero	412	228	55,34%
Citação				
	Título	13204	5281	40,00%
	Periódico	1197	536	44,78%
	Editora	585	161	27,52%
Nome				
	Nomes	5000	500	10,00%

Tabela 4.2: Numero total de objetos, número de objetos distintos do total de objetos e percentual de objetos distintos no conjunto de dados representativos

Como mencionado anteriormente, são utilizadas funções de similaridades específicas para cada atributo, dependendo do seu domínio. Na seção 4.2 é apresentado um experimento que mostra o comportamento do escore ajustado com a utilização de diferentes funções de similaridade para o mesmo atributo.

O resultado do escore ajustado tende a ser melhor quando a função de similaridade é adequada para o domínio a ser utilizado. A tabela 4.3 apresenta as funções de similaridade utilizadas para cada atributo das bases de dados deste experimento. Tais funções estão disponíveis nos pacotes Java *SecondString* (COHEN; RAVIKUMAR; FIENBERG, 2009) ou *SimMetrics* (CHAPMAN, 2009). As funções utilizadas foram escolhidas baseadas em experimentos realizados anteriormente pelo grupo (STASIU; HEUSER; SILVA, 2005; DORNELES et al., 2004) e tentou-se escolher funções adequadas para os domínios dos atributos. Discussões e avaliações de diversas funções de similaridade podem ser encontradas na literatura (COHEN; RAVIKUMAR; FIENBERG, 2003; LEE, 2001; SILVA et al., 2007).

Domínio	Atributo - Função
Filme	Título - Q-grams
	Diretor - JaroWinklerTFIDF
	Gênero - Levenstein
Citação	Título - Q-grams
	Periódico - JaroWinkler
	Editora - MongeElkan
Nomes	Nome - Soundex

Tabela 4.3: Funções de similaridade utilizadas para cada atributo

4.1.2 Processo de Treinamento

O resultado do processo de treinamento, ou processo de estimativa do escore ajustado, é uma tabela chamada de Tabela de Mapeamento de Escore (TaME), que representa o relacionamento entre o escore original e o escore ajustado para uma função de similaridade específica aplicada em um atributo específico.

Para calcular a TaMe foi efetuado o processo de treinamento explicado no capítulo 3.

A seguir, são descritos os passos executados, os quais foram efetuados para cada atributo, tanto para os dados reais quanto para os dados representativos.

1. **Conjunto de Consultas.** O primeiro passo é sortear aleatoriamente uma amostra de 40 valores distintos $Q = (q_1, q_2, \ldots, q_{40})$, eliminando duplicatas idênticas, pois o objetivo é avaliar a habilidade da proposta em encontrar diferentes representações de um objeto do mundo real.

2. **Criação do *ranking* original (*ground ranking*).** Agora, para cada valor $q \in Q$, um *ranking* original $R(q) = \{a_1, a_2, \ldots, a_n\}$ foi construído baseado na definição 2.

3. **Cálculo da precisão interpolada.** Neste passo, um especialista marca cada valor a_i em cada *ranking* $R(q)$ como "relevante" (se representa o mesmo objeto no mundo real comparando com q) ou "irrelevante". Então, a precisão é calculada para cada posição i no *ranking*. Após calculada a precisão para todas posições i do *ranking* $R(q)$, o próximo passo é calcular a precisão interpolada (Definição 3) para cada um dos 11 pontos de escore $[0,0; 0,1; \ldots; 0,9; 1]$.

4. **Criação da tabela de mapeamento de escore.** Tendo como entrada as tabelas de precisão interpolada dos 40 *rankings*, calculadas no passo anterior, a tabela de mapeamento de escore para este atributo é calculada de acordo com a Definição 4.

   Na tabela 4.4, pode ser vista a TaME calculada para o atributo gênero da base de dados filmes real.

Escore Original	Escore Ajustado
1	1,000000000
0,9	0,833333333
0,8	0,788888889
0,7	0,777777778
0,6	0,743333333
0,5	0,712777778
0,4	0,680000000
0,3	0,441681097
0,2	0,210786436
0,1	0,067755027
0,0	0,064137417

Tabela 4.4: TaME do atributo gênero da base de dados filmes do conjunto real

O processo de treinamento resulta em duas TaME para cada atributo, uma tabela (M_a) correspondendo à base de dados real e outra (M_r) correspondendo à base de dados representativa.

4.1.3 Verificando a precisão do escore ajustado

Como dito anteriormente, os experimentos apresentados neste capítulo têm como objetivo comparar os resultados obtidos pela utilização do escore original com os obtidos utilizando o escore ajustado. Com o processo de treinamento concluído, foram calculadas as tabelas TaME (resultado do escore original) para cada atributo das bases de dados

utilizadas, então o próximo passo é efetuar o processo de casamento ajustando o escore retornado pelas funções de similaridade com base nas TaMEs calculadas. Este processo foi efetuado para cada atributo de cada base com os seguintes passos:

1. **Conjunto de Consultas.** O primeiro passo é sortear randomicamente outra amostra de 40 valores distintos $Q = (q_1, q_2, \ldots, q_{40})$, também eliminando duplicatas idênticas.

2. **Criação do *ranking* ajustado.** Agora, para cada valor $q \in Q$, um *ranking* ajustado $R(q) = \{a_1, a_2, \ldots, a_n\}$ é gerado. Este é similar ao *ranking* original (Definição 2), a única diferença é que, ao invés de ordenar os valores pelo escore original retornado pela função de similaridade ($f_i(q, a_i)$), a ordenação é feita pelo escore ajustado ($\alpha_{M_i}(f_i(q, a_i))$) correspondente.

3. **Cálculo da precisão interpolada.** Este passo é igual ao passo 3 do processo de treinamento. Um especialista analisa o *ranking* e marca cada valor a_i como "relevante" ou "irrelevante". Então é calculada a precisão para cada ponto a_i do *ranking*. Com as precisões de todas as posições a_i calculadas, o próximo passo é calcular a precisão interpolada (de acordo com a definição 3) para cada um dos 11 pontos de limiar $[0,0; 0,1; \ldots, 0,9; 1]$.

4. **Resultado do Casamento com escore ajustado.** Tendo como resultado do passo anterior às precisões interpoladas para cada uma das 40 consultas, o próximo passo é calcular a média das precisões dessas 40 consultas, a fim de obter uma tabela com o resultado do casamento (mesmo processo da formação da TaME, Definição 4).

Um exemplo de resultado deste processo pode ser visto na tabela 4.5, onde são apresentados os resultados para o atributo título na base de dados de filmes. A primeira coluna contém os onze pontos de limiar e a segunda apresenta os valores de precisão que foram calculadas com o uso do escore ajustado.

Limiar	Escore Ajustado
1	1,000000000
0,9	0,864583333
0,8	0,795000000
0,7	0,697351190
0,6	0,664672619
0,5	0,513189762
0,4	0,396844880
0,3	0,396844880
0,2	0,183303200
0,1	0,050235837
0	0,030669672

Tabela 4.5: Exemplo de resultado obtido através do processo de casamento utilizando escore ajustado para o atributo título na base de dados de filmes

É importante lembrar que um aspecto central desta abordagem é permitir que o usuário especifique a qualidade do processo de casamento através de um valor de limiar que expresse a precisão esperada nos resultados. Se a abordagem executar corretamente, a

precisão média obtida pelo escore ajustado deveria ser igual ao limiar fornecido. No entanto, como provavelmente o processo de estimação não é perfeito, diferenças entre esses dois valores podem existir. Um exemplo disso pode ser visto na tabela 4.5, a precisão média calculada para o limiar $0,4$ é $0,39684488$.

Para validar a abordagem proposta, é possível avaliar a diferença entre a precisão média obtida nos experimentos e a precisão fornecida pelo usuário como um limiar. Então, para isso, foram plotados gráficos com os resultados obtidos utilizando o escore original e o escore ajustado, conforme os passos explicados anteriormente.

As figuras 4.1 e 4.2 apresentam os valores de precisão média para os atributos no domínio de citação. Os gráficos na figura 4.1 contem os resultados utilizando a base de dados real, já na figura 4.2 são apresentados os gráficos com os resultados usando a base de dados representativa para o treinamento. Cada uma das duas figuras apresentam os dados para os atributos *título*, *Periódico* e *editora*.

As figuras 4.3 e 4.4 mostram os valores de precisão média para os atributos do domínio de filmes. Os gráficos na figura 4.3 apresentam os resultados obtidos com a utilização da bases de dados real, já os da figura 4.4 apresentam os resultados usando a base de dados representativa para o treinamento. As duas figuras apresentam os resultados para os atributos título, diretor e gênero.

E, por último, as figuras 4.5 e 4.6 apresentam os resultados para o domínio nome, a figura 4.5 contendo as precisões médias utilizando a base de dados real e a figura 4.6 contendo as precisões médias calculadas com o treinamento realizado nos dados representativos.

Nestes gráficos, o eixo x contém os valores de limiar e o eixo y contém a precisão média obtida tanto para o escore ajustado quanto para o escore original. Cada gráfico contém três linhas. A linha contínua (linha reta $y = -x$) representa o caso ideal, quando a precisão média é exatamente a precisão esperada, fornecida pelo usuário através do valor de limiar. A linha pontilhada corresponde aos valores de precisão média utilizando o escore ajustado e a linha tracejada apresenta as precisões médias obtidas com o uso do escore original. Em outras palavras, a linha pontilhada apresenta a aplicação da abordagem que está sendo testada neste trabalho, e a linha tracejada é utilizada como um *baseline* para a avaliação desta abordagem.

Observando os gráficos, é possível ver claramente que a abordagem aqui testada apresenta valores muito mais próximos da precisão esperada pelo usuário (linha contínua) do que o *baseline*. É importante enfatizar que estes resultados foram obtidos com um esforço pequeno de treinamento, envolvendo apenas 40 consultas por atributo. Analisando os resultados dos atributos *Periódico* e *Editora*, tanto da base real quanto da base representativa, é possível ver claramente a melhora que se obteve com a utilização da abordagem aqui testada, onde a curva do *baseline* apresenta resultado longe da curva ideal (linha contínua), e com o ajuste a curva passa a ficar bem próxima.

Para quantificar a diferença entre a precisão esperada, dada pelo usuário através de um limiar, e a precisão média obtida pelo uso do escore ajustado, foram calculados os desvio dos quadrados das diferenças (como efetuado no método dos mínimos quadrados) entre elas. O desvio é calculado pela equação:

$$d_a = \sum_{i=0}^{10} [apa_i - t_i]^2 \tag{4.1}$$

onde t_i é um dos onze pontos de limiar (a precisão esperada pelo usuário) e apa_i é a precisão média resultante para cada limiar usando o escore ajustado.

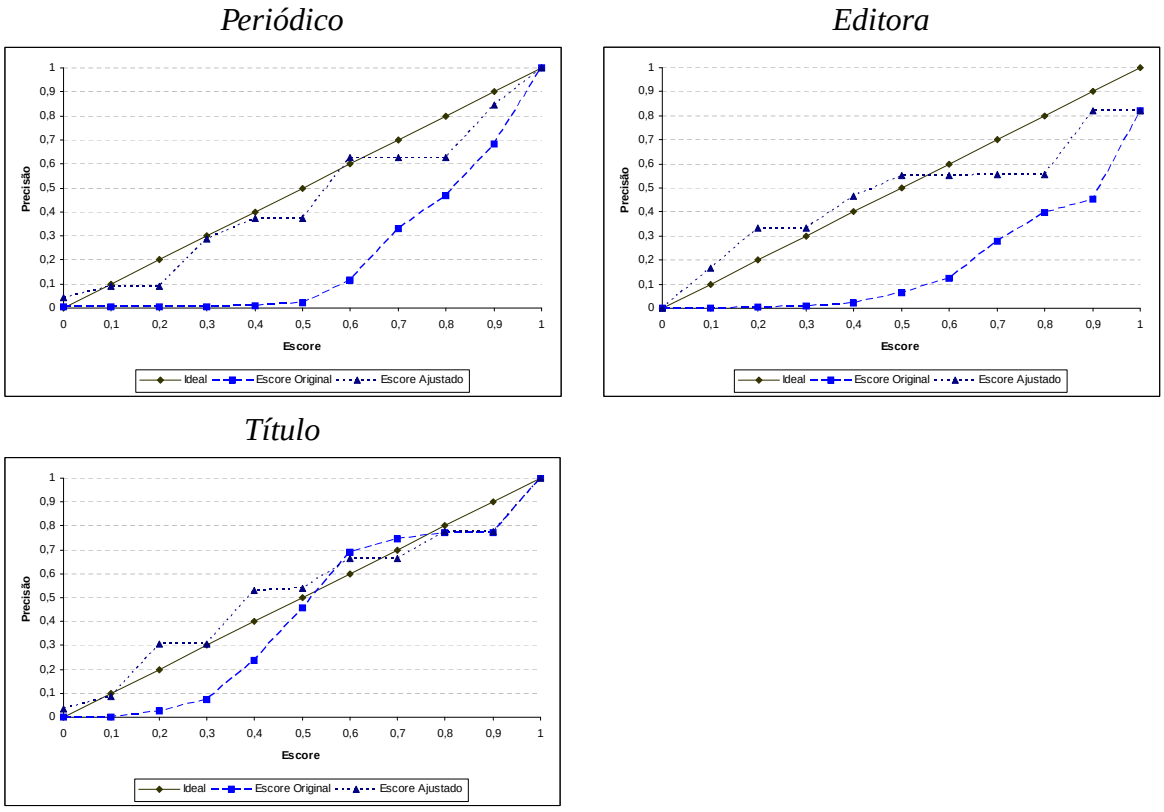

Figura 4.1: Precisão média calculada utilizando o escore ajustado e o escore original para o domínio *Citação*. Treinamento efetuado sobre a *base de dados real*. A linha contínua representa o caso ideal, a linha pontilhada representa o resultado com o escore ajustado, e a linha tracejada representa o resultado obtido utilizando o escore original.

Também foi calculado o desvio entre os resultados obtidos quando a abordagem de escore ajustado não é utilizada (quando os resultados das funções de similaridade são diretamente utilizados, apresentado nos gráficos pela linha tracejada) e o caso ideal (linha contínua). Esse desvio é calculado pela equação:

$$d_g = \sum_{i=0}^{10} [apg_i - t_i]^2 \tag{4.2}$$

onde t_i é um dos onze pontos de precisão $[0,0; 0,1; \ldots; 1]$ e apg_i é a precisão média calculada utilizando diretamente o valor do escore original.

Na tabela 4.6 são apresentados os valores de desvio para cada atributo. Nesta, a primeira coluna lista os domínios utilizados, a segunda identifica os atributos, a terceira coluna d_g contém o desvio entre o caso ideal e o resultado obtido com o uso do escore original e a quarta e quinta coluna d_a apresentam os valores de desvio entre o caso ideal e o resultado obtido com o uso do escore ajustado, porém a quarta apresentando os resultados com o treinamento realizado na *base representativa* e a quinta utilizando a *base real*.

Os resultados obtidos nesse experimento mostram que o uso da abordagem de escore ajustado proporciona resultados mais próximos do caso ideal. Esse fato pode ser visualmente observado nos gráficos das figuras 4.1, 4.2, 4.3, 4.4, 4.5 e 4.6 e confirmado pelo

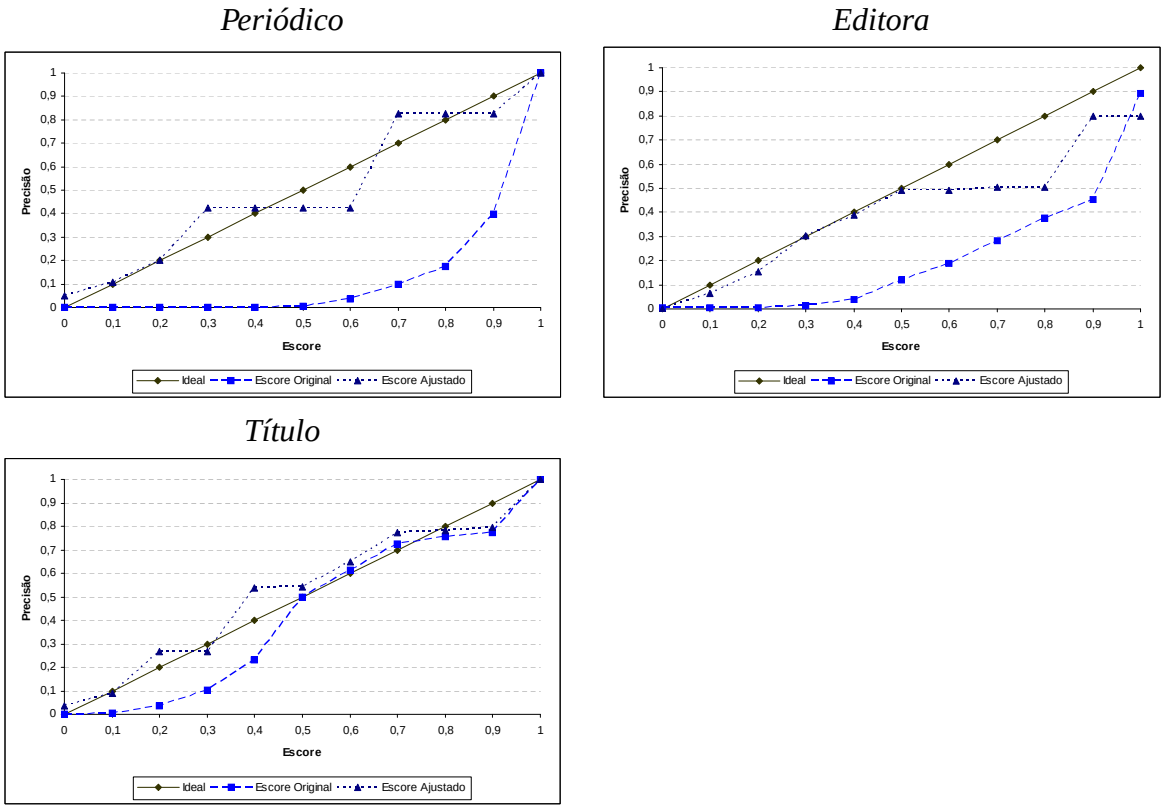

Figura 4.2: Precisão média calculada utilizando o escore ajustado e o escore original para o domínio *Citação*. Treinamento efetuado sobre a *base de dados representativa*. A linha contínua representa o caso ideal, a linha pontilhada representa o resultado com o escore ajustado, e a linha tracejada representa o resultado obtido utilizando o escore original.

seu pequeno valor de desvio mostrado na tabela 4.6. Os resultados da tabela 4.6 são de fato uma representação compacta dos resultados apresentados nos gráficos.

Além disso, os resultados obtidos com o treinamento realizado sobre a base de dados representativa foram próximos aos obtidos sobre a base de dados real. Isso significa que o treinamento pode ser realizado em uma base de um determinado domínio e ser utilizado em outras bases deste domínio, facilitando assim a utilização desta abordagem.

Ao analisar a tabela 4.6, algo muito curioso pode ser observado, a diferença resultante do atributo diretor da base de filmes e o atributo título da base citações, do conjunto representativo, é menor do que do conjunto real. Esse fato pode ter ocorrido por algumas das consultas sorteadas aleatóriamente (na base real) não representarem bem o restante da base. É importante lembrar se que trata de um processo de estimativa e a precisão dos resultados depende do quão bem o conjunto de consultas sorteados representa a base de dados. Para estes atributos, o conjunto de consultas sorteados da base de dados representativa representou melhor a base como um todo. Este também é um bom resultado, porque se pode dizer que o treinamento não precisa ser feito na própria base de dados, e pode ser feito apenas uma vez em uma base de dados com o mesmo domínio (neste caso a base representativa).

Em síntese, neste capítulo foi apresentado o experimento que compara a utilização do escore ajustado com o escore original. Para a sua realização, foram utilizados dois conjuntos de dados: reais e representativos. O processo de treinamento foi realizado para

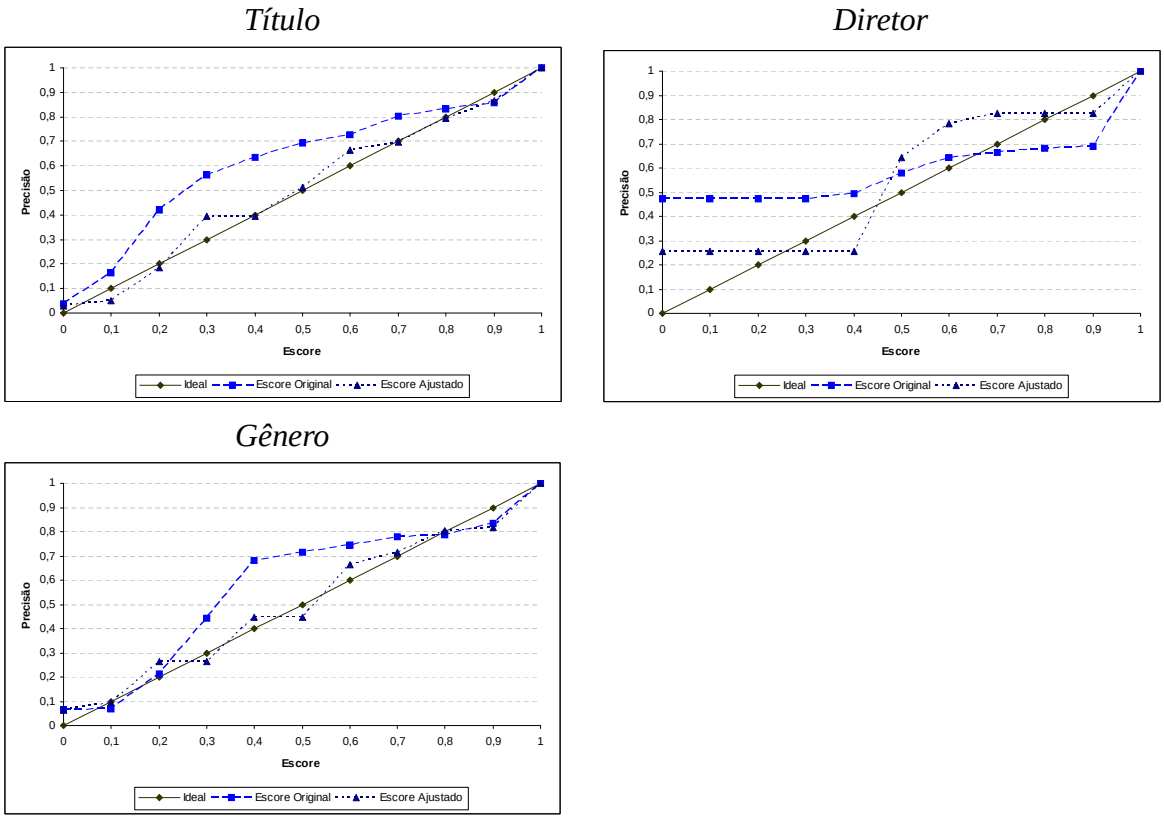

Figura 4.3: Precisão média calculada utilizando o escore ajustado e o escore original para o domínio *Filme*. Treinamento efetuado sobre a *base de dados real*. A linha contínua representa o caso ideal, a linha pontilhada representa o resultado com o escore ajustado, e a linha tracejada representa o resultado obtido utilizando o escore original.

todos atributos, tanto do conjunto de dados reais quanto do conjunto representativo. Foram utilizados esses dois tipos de dados com objetivo de demonstrar que o treinamento não precisa ser realizado na própria base de dados, mas que pode ser efetuado em qualquer outra base do mesmo domínio (utilizando a mesma função de similaridade). Após, o processo de casamento utilizando o escore ajustado foi realizado duas vezes com o conjunto de dados reais, uma com o treinamento real e a outra com o treinamento representativo. Os resultados foram plotados em gráficos para que fosse possível analisá-los visualmente. Tais gráficos são apresentados nas figuras 4.1, 4.2, 4.3, 4.4, 4.5 e 4.6 e possuem os resultados do escore ajustado, do escore original, e uma reta que seria um caso de precisão ideal. No entanto, para que fosse possível quantificar a diferença entre a precisão ideal e a precisão média obtida pelo uso do escore ajustado, foram calculados os desvio dos quadrados das diferenças entre eles. Tais resultados foram apresentados na tabela 4.6. Após analisar todos os resultados, é possível dizer que a utilização do escore ajustado proporciona melhora nos resultados e possibilita também utilizá-lo como um limiar intuitivo para o usuário. Além disso, o treinamento pode ser efetuado em outra base de dados do mesmo domínio e não necessariamente na própria base de dados.

No próximo capítulo, é apresentado um experimento realizado que tem como objetivo analisar o comportamento do escore ajustado utilizando diferentes funções de similaridade.

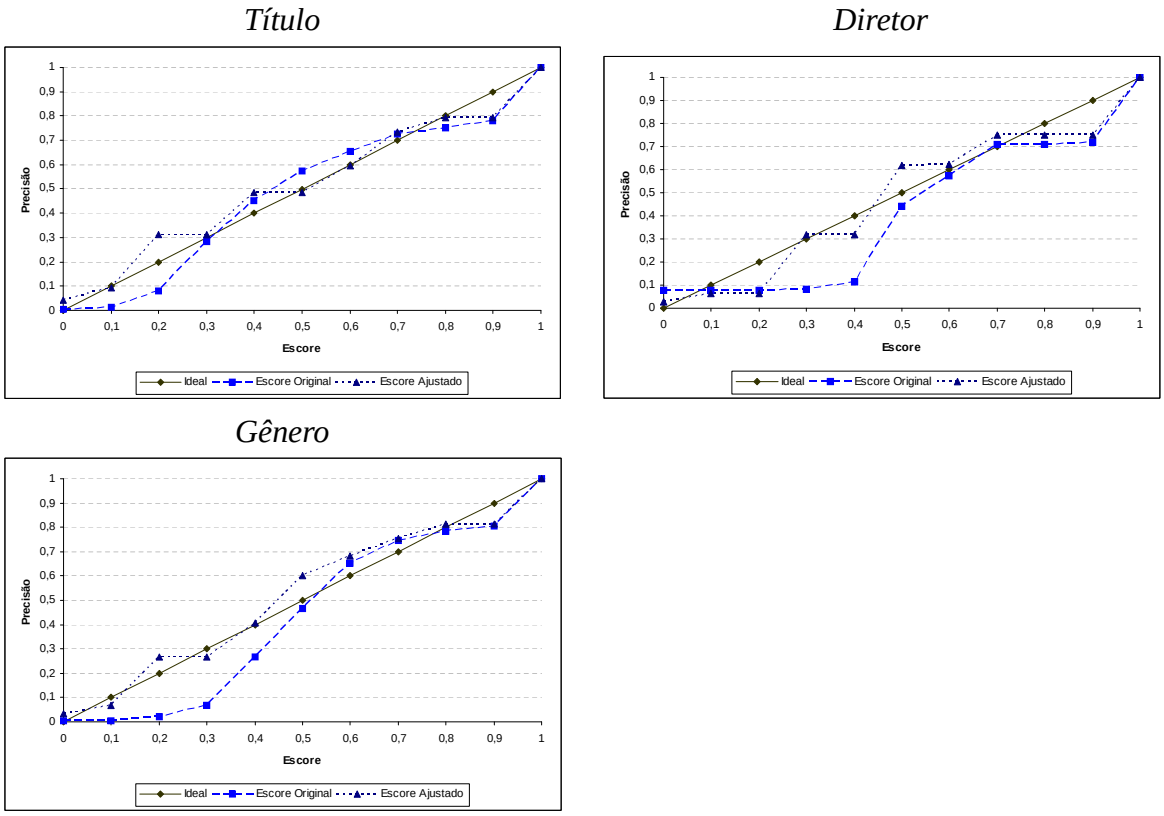

Figura 4.4: Precisão média calculada utilizando o escore ajustado e o escore original para o domínio *Filme*. Treinamento efetuado sobre a *base de dados representativa*. A linha contínua representa o caso ideal, a linha pontilhada representa o resultado com o escore ajustado, e a linha tracejada representa o resultado obtido utilizando o escore original.

4.2 Análise do Comportamento do Escore Ajustado com Diferentes Funções de Similaridade

Outro ponto em aberto no trabalho anterior é a análise do comportamento do escore ajustado utilizando diferentes funções de similaridade. Nesta seção são apresentados os experimentos desenvolvidos utilizando a abordagem de escore ajustado com vinte e duas funções de similaridade distintas. O objetivo é observar se os resultados permanecem próximos, ou seja, verificar se os escores ajustados são comparáveis.

É importante salientar que este experimento não está avaliando as funções de similaridade, mas, sim, analisando como o escore ajustado se comporta com a utilização de diferentes funções.

Para a realização deste experimento foram utilizadas duas bases de dados artificiais (base $nomes_1$ e $nomes_2$), criadas com a ferramenta *Febrl*. Cada base possui 5.000 registros contendo nomes de pessoas (nomes compostos). Destes cinco mil, quinhentos representam pessoas distintas e os outros quatro mil e quinhentos representam duplicatas.

A primeira base de dados criada ($nomes_1$) foi utilizada para o processo de treinamento e a segunda ($nomes_2$) para o processo de casamento. Durante o processo de casamento, o escore retornado pela função de similaridade é ajustado utilizando a TaME resultante do processo de treinamento. Neste experimento, foram utilizadas 21 funções de similaridade disponíveis no pacote Java *SimMetrics* (CHAPMAN, 2009) e mais uma desenvolvida por

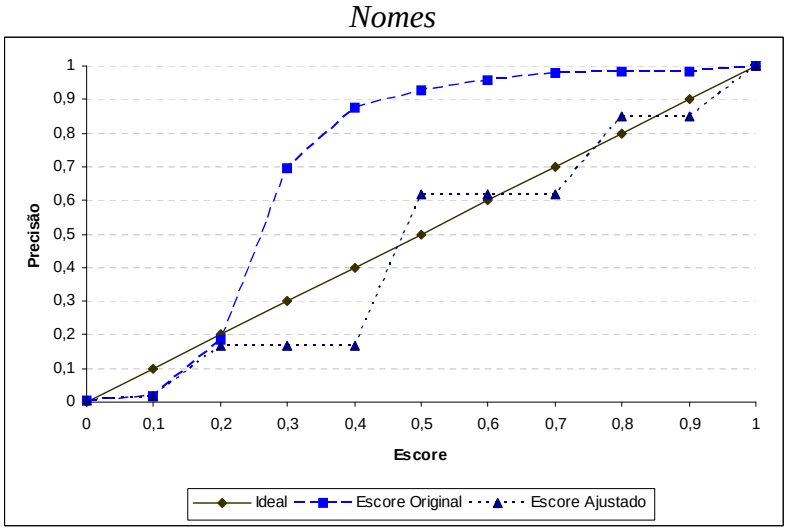

Figura 4.5: Precisão média calculada utilizando o escore ajustado e o escore original para o domínio de *Nome*. Treinamento efetuado sobre a *base de dados real*. A linha contínua representa o caso ideal, a linha pontilhada representa o resultado com o escore ajustado, e a linha tracejada representa o resultado obtido utilizando o escore original.

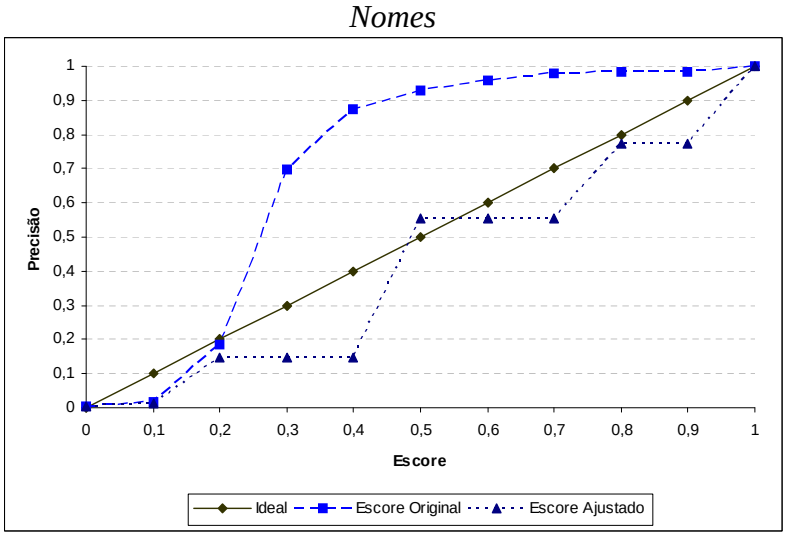

Figura 4.6: Precisão média calculada utilizando o escore ajustado e o escore original para o domínio *Nome*. Treinamento efetuado sobre a *base de dados representativa*. A linha contínua representa o caso ideal, a linha pontilhada representa o resultado com o escore ajustado, e a linha tracejada representa o resultado obtido utilizando o escore original.

um aluno de doutorado da UFRGS (MERGEN; HEUSER, 2005; RITT et al., 2008). Tais funções são listadas na tabela 4.7.

O processo de treinamento, descrito em detalhes no capítulo 3 e na subseção 4.1.2, foi realizado para esse experimento seguindo os seguintes passos:

1. Uma amostra de 40 consultas distintas é sorteada da base de dados $nomes_1$ $Q = (q_1, q_2, \ldots, q_{40})$.

2. Para cada uma das vinte e duas funções de similaridade, o seguinte processo foi seguido:

Domínio	Atributo	dg	da (Conjunto de Dados Representativos)	da (Conjunto de Dados Real)
Cinema				
	Título	0,245324404	0,035447116	0,018727643
	Diretor	0,548393691	0,070206345	0,191055858
	Gênero	0,180179338	0,035131700	0,025824542
Citação				
	Título	0,148328543	0,047467229	0,051953852
	Periódico	1,042834481	0,076918953	0,071390621
	Editora	1,269746357	0,191854430	0,150769219
Nome				
	Nomes	0,81719468	0,142045348	0,104677381

d_g – Desvio entre o caso ideal e os resultados obtidos quando o escore original é utilizado.
d_a – Desvio entre o caso ideal e os resultados quando o escore ajustado é aplicado.

Tabela 4.6: Valores de desvio

(a) Para cada consulta $q \in Q$, um *ranking* original foi construído baseado na definição 2.

(b) Como a base de dados $nomes_1$ já é avaliada, é calculada automaticamente a precisão para cada posição do *ranking* original.

(c) Com as precisões calculadas, o próximo passo é calcular a precisão interpolada (Definição 3) para cada um dos 11 pontos de escore $[0,0; 0,1; \ldots, 0,9; 1]$ de cada *ranking*.

(d) Tendo a precisão interpolada dos 11 pontos para cada um dos 40 *rankings*, é efetuada uma média aritmética dos pontos, por exemplo: $(ponto1ranking1 + ponto1ranking2 + \cdots + ponto1ranking40)/40$. Resultando na tabela de mapeamento de escore (TaME, Definição 4).

Com o processo de treinamento concluído, é possível efetuar os casamentos utilizando escores ajustados, com cada uma das 22 funções de similaridade. Para isso foi feito o processo descrito abaixo.

1. 40 consultas foram sorteadas randomicamente da base de dados $nomes_2$.

2. Após, para cada função de similaridade, foram seguidos os seguintes passos:

   (a) Um *ranking* ajustado é calculado utilizando cada consulta sorteada anteriormente. O processo é quase o mesmo do *ranking* original (Definição 2) só que, ao invés de utilizar diretamente o escore retornado pela função de similaridade, é utilizado o seu respectivo escore ajustado presente na TaME, calculada no treinamento.

   (b) Agora são calculadas as precisões em cada linha dos 40 *rankings* construídos. Com as precisões calculadas, é computada então a precisão interpolada para cada um dos 11 pontos de escore $[0,0; 0,1; \ldots, 0,9; 1]$.

   (c) A média das precisões interpoladas de cada uma das 40 consultas é calculada, a fim de obter o resultado final.

Funções de similaridade
Q-grams Distance
JaroWinkler
Levenshtein (Edit Distance)
Block Distance
Soundex
ChapmanLengthDeviation
ChapmanMatchingSoundex
Cosine Similarity
Dice Similarity
Euclidean Distance
Jaccard Similarity
Jaro
MatchingCoefficient
MongeElkan
NeedlemanWunch
OverlapCoefficient
SmithWaterman
SmithWatermanGotoh
SmithWatermanGotohWindowedAffine
TagLink
TagLinkToken
Carla

Tabela 4.7: Funções de similaridade utilizadas no experimento 4.2

3. Por fim, são plotados, em um gráfico, os resultados de todas as 22 funções de similaridade utilizadas.

Como estão sendo utilizadas muitas funções, os resultados foram separados em dois grupos, um com os resultados mais semelhantes e o segundo com os demais, para que fique mais fácil a observação dos gráficos. O primeiro, contendo os resultados semelhantes é apresentado na figura 4.7, e o segundo na figura 4.8.

Nos gráficos, a linha reta ($y = -x$) representa o caso ideal, onde a precisão média é exatamente a precisão esperada do processo de casamento, cada linha restante, representa o resultado de uma função de similaridade, as quais são identificadas na legenda localizada na parte inferior do gráfico. O eixo x contém o valor de escore e o eixo y representa a precisão obtida.

Analisando o gráfico da figura 4.7 é possível observar que as curvas permanecem com comportamento parecido, variando em torno da reta ideal. Com isso, pode-se dizer que estas funções são comparáveis. Porém, os resultados apresentados na figura 4.8 apresentam comportamentos distintos, algumas ainda variando em alguns pontos em torno da reta ideal. Algo que chama atenção nestas curvas são os resultados de *Monge Elkan*, *Soundex* e de *Chapman Length Deviation*, onde no ponto 1 de escore, não se tem 100% de precisão. Para *Monge Elkan* a precisão média no ponto 1 de escore é de $0,54$, para *Soundex* é $0,93$ e para *Chapman Length Deviation* é de $0,05$, esta última permanecendo o mesmo resultado até o ponto $0,1$ de escore. Para descobrir o motivo, foram analisados os resultados de todos os *rankings* formados durante o processo de casamento e foi descoberto que estas funções de similaridade retornam escore 1 para alguns atributos que não

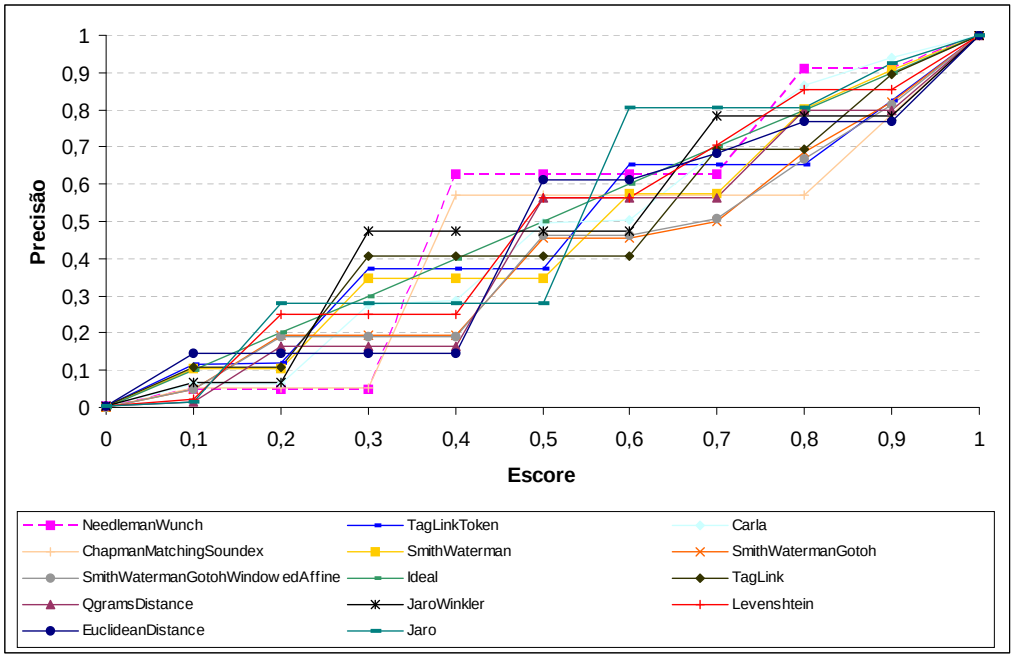

Figura 4.7: Primeiro conjunto de resultados com precisão média calculada utilizando o escore ajustado com diferentes funções de similaridade.

representam o mesmo registro. Então, na hora de calcular a precisão interpolada, o ponto 1 de escore não possui 100% de precisão.

Para analisar se as funções que ficaram com os resultados longe do ideal possuem realmente o resultado assim, ou se os resultados pioraram devido à utilização do escore ajustado, foi executado o mesmo processo de casamento descrito acima nesta seção, só que utilizando o escore original ao invés do ajustado (mesmo processo do treinamento, porém utilizando a base de dados $nomes_2$). Para quantificar a diferença entre a precisão esperada, dada pelo usuário através de um limiar, e as precisões médias obtidas pelo uso do escore ajustado e original, foram calculados os desvios dos quadrados das diferenças entre elas (como efetuado no método dos mínimos quadrados). O desvio para o escore ajustado é calculado pela equação 4.1, e para o escore original é calculado pela equação 4.2. As tabelas 4.8 e 4.10 apresentam o nome da função de similaridade utilizada e o seu respectivo valor de desvio. Nestas tabelas, as funções estão ordenadas de forma crescente de acordo com o desvio. A tabela 4.8 apresenta os resultados dos desvios dos quadrados utilizando o escore ajustado (DA) e a 4.10 apresenta os resultados utilizando o escore original (DG).

Analisando os resultados de DA e de DG é possível observar que as funções nas quais as curvas ficaram longe da ideal, ou seja, possuem valor de DA alto, também possuem um valor alto de DG. Estas funções podem ser analisadas na tabela 4.8, quando o valor de DA é maior que $0,4$. Outro aspecto, que pode ser observado comparando essas tabelas, é a melhora na qualidade dos resultados das funções de similaridade, onde todas obtiveram ganho quando utilizado o escore ajustado. Algumas se destacam por melhorarem muito, como por exemplo a *TagLinkToken* que possui um valor de desvio de $0,48$ e passou a ter um valor de $0,06$. A função *Levenshtein* também, com o valor $0,32$ de DG passou a ter o desvio ajustado de $0,04$, deixando a sétima posição do *ranking* e passando a ser a primeira. Um outro exemplo de grande melhora é a função *JaroWinkler* que possui um valor de desvio de $1,39$ e passou a ter um valor de $0,09$, onde mudou da décima oitava

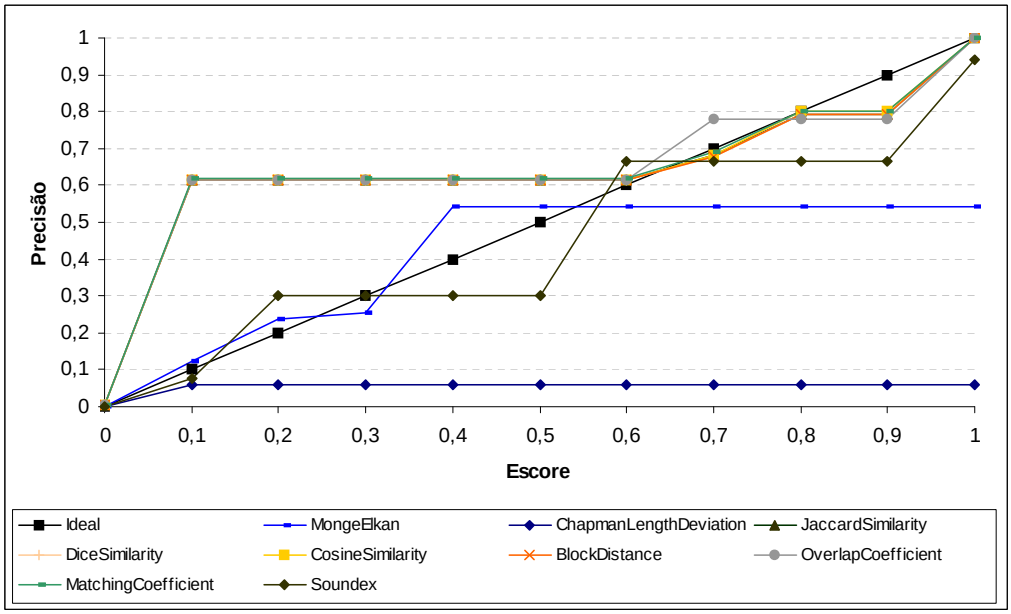

Figura 4.8: Segundo conjunto de resultados com precisão média calculada utilizando o escore ajustado com diferentes funções de similaridade.

posição do *ranking* para a sexta.

Um segundo teste realizado foi utilizar um software chamado *SimEval* (HEUSER; KRIESER; ORENGO, 2007) desenvolvido por um aluno de graduação da UFRGS, o qual avalia a qualidade de funções de similaridade. Foi submetida, neste software, a base de dados $nomes_2$, para que ele avaliasse as funções neste domínio, como resultado foi retornada uma tabela contendo um valor de similaridade média para cada função. Os resultados deste teste utilizando o *SimEval* são listados na tabela 4.11. Esta está ordenada de acordo com o valor de precisão média, ou seja, quanto mais no topo a função estiver, melhor são os seus resultados para esta base de dados.

Como pode ser observado, todas as funções que retornaram valores de DA maior que $0,4$ possuem precisão média baixa. Isso quer dizer que funções adequadas para o domínio retornam um bom escore ajustado e as funções que apresentam resultados de escore ajustado ruim são realmente funções ruins em termos de precisão média para o domínio.

Para comparar os dois *rankings*, o das precisões médias (figura 4.11) com o contendo os valores de DA (figura 4.8), foi utilizada uma medida estatística chamada *Coeficiente de Correlação de Pearson*. Neste caso utilizada para medir o grau da correlação entre os dois *rankings*. Este coeficiente r assume apenas valores entre -1 e 1. Onde, $r = 1$ significa uma correlação perfeita positiva, $r = -1$ significa uma correlação negativa perfeita e $r = 0$ significa que não existe correlação entre os *rankings*. Santos (2007) propõe uma classificação para a correlação r, apresentada na tabela 4.9.

O resultado de correlação obtido entre os dois *rankings* foi $r = -0,626$. De acordo com a classificação de Santos (2007), esses dois *rankings* possuem uma correlação moderada negativa.

Com isso é possível dizer que, à medida que a precisão aumenta, o desvio tende a diminuir. Ou seja, o escore ajustado depende de funções que sejam adequadas para o domínio de dados. Funções inadequadas (que possuem precisão média baixa) retornam escores ajustados ruins.

Função de Similaridade	DA
Levenshtein	0,043525715
Carla	0,048286607
SmithWaterman	0,054546006
TagLinkToken	0,061647959
TagLink	0,076285463
JaroWinkler	0,091161843
QgramsDistance	0,115406996
EuclideanDistance	0,125401033
Jaro	0,131351416
SmithWatermanGotoh	0,140224418
SmithWatermanGotohWindowedAffine	0,140883571
Soundex	0,142875217
NeedlemanWunch	0,175100705
ChapmanMatchingSoundex	0,200858195
MongeElkan	0,461887441
CosineSimilarity	0,599957808
JaccardSimilarity	0,601285933
BlockDistance	0,601526819
DiceSimilarity	0,601526819
OverlapCoefficient	0,610751808
MatchingCoefficient	0,619746763
ChapmanLengthDeviation	3,231800463

Tabela 4.8: Valores de desvio entre o caso ideal e os resultados obtidos quando o escore ajustado é utilizado

Coeficiente de Correlação	Correlação
$r = 1$	Perfeita positiva
$0,8 \leq r < 1$	Forte positiva
$0,5 \leq r < 0,8$	Moderada positiva
$0,1 \leq r < 0,5$	Fraca positiva
$0 < r < 0,1$	Ínfima positiva
0	Nula
$-0,1 < r < 0$	Ínfima negativa
$-0,5 < r \leq -0,1$	Fraca negativa
$-0,8 < r \leq -0,5$	Moderada negativa
$-1 < r \leq -0,8$	Forte negativa
$r = -1$	Perfeita negativa

Tabela 4.9: Classificação de valores de correlação.

Função de Similaridade	DG
SmithWatermanGotohWindowedAffine	0,192679508
SmithWatermanGotoh	0,192696295
Carla	0,217367775
SmithWaterman	0,220073361
EuclideanDistance	0,269099936
TagLink	0,32682349
Levenshtein	0,329531076
TagLinkToken	0,48457099
OverlapCoefficient	0,65764767
QgramsDistance	0,679861649
CosineSimilarity	0,68449664
BlockDistance	0,68684039
DiceSimilarity	0,68684039
MatchingCoefficient	0,708922106
NeedlemanWunch	0,869604543
JaccardSimilarity	0,890898289
Jaro	1,061960329
JaroWinkler	1,394025309
ChapmanMatchingSoundex	2,008895935
MongeElkan	2,216242138
Soundex	2,222813949
ChapmanLengthDeviation	3,676044415

Tabela 4.10: Valores de desvio entre o caso ideal e os resultados obtidos quando o escore original é utilizado

Função de Similaridade	Precisão Média
TagLink	0,925061896
Levenshtein	0,836548746
QGramsDistance	0,817099451
TagLinkToken	0,755492276
Carla	0,749924803
NeedlemanWunch	0,743707391
Jaro	0,71769069
JaroWinkler	0,714782859
SmithWaterman	0,707495832
SmithWatermanGotoh	0,662081818
SmithWatermanGotohWindowedAffine	0,662081818
ChapmanMatchingSoundex	0,641865975
Soundex	0,404850819
MongeElkan	0,360851814
BlockDistance	0,133195768
CosineSimilarity	0,133195768
DiceSimilarity	0,133195768
JaccardSimilarity	0,133195768
MatchingCoefficient	0,133195768
EuclideanDistance	0,132561165
OverlapCoefficient	0,131192958
ChapmanLengthDeviation	0,013324276

Tabela 4.11: Valores de precisão média calculados pelo software *SimEval*

4.3 Escore Ajustado em Casamento de Registros

Como descrito no final do capítulo 3, outro ponto em aberto, no trabalho anterior, é experimentar a combinação do escore ajustado no processo de casamento de registros.

Conforme foi discutido no capítulo 2, existem várias propostas para casamento de registros na literatura (para maiores detalhes sobre esse assunto, existe um *survey* (ELMAGARMID; IPEIROTIS; VERYKIOS, 2007)). Algumas destas abordagens (BILENKO et al., 2003; CARVALHO et al., 2006; CHAUDHURI et al., 2007; LEITÃO; CALADO; WEIS, 2007; SARAWAGI; BHAMIDIPATY, 2002; TEJADA; KNOBLOCK; MINTON, 2001) combinam os valores de escore de cada atributo (possivelmente calculados por diferentes funções de similaridade) em um único escore de similaridade para o registro com um todo. Métodos para combinação de escores de similaridade variam entre simplesmente fazer a média sobre os escores de similaridade dos atributos (DEY; SARKAR; DE, 1998; FELLEGI; SUNTER, 1969) até métodos mais sofisticados como os que utilizam redes bayesianas (LEITÃO; CALADO; WEIS, 2007) ou programação genética (CARVALHO et al., 2006).

Nesta seção, é apresentado um conjunto de experimentos com objetivo de demonstrar que, combinando os escores ajustados para o processo de casamento de registro, é possível melhorar a qualidade dos métodos existentes, que combinam os valores de escore de cada atributo.

A grande vantagem de utilizar a combinação dos escores ajustados é permitir a combinação correta dos escores originais, gerados por funções de similaridades diferentes, em um único escore. Pois, para cada atributo de um registro, é utilizada uma determinada função de similaridade que melhor se comporta no domínio. Como exemplo, para calcular a similaridade entre dois registros de filme, que contêm três atributos cada, $\langle d_1, t_1, g_1 \rangle$ e $\langle d_2, t_2, g_2 \rangle$, é necessário comparar atributo por atributo destes registros utilizando funções de similaridade f adequadas a cada um, $f_1(d_1, d_2)$, $f_2(t_1, t_2)$ e $f_3(g_1, g_2)$; cada comparação gera um escore e, $f(d1, d2) \mapsto e$, estes escores gerados necessitam ser combinados de alguma maneira, $(e_d; e_t; e_g)$. Então, como as funções de similaridade possuem distribuições diferentes, fica inviável combinar seus resultados. Por exemplo, a similaridade entre as *strings* $s_1 = Marcos$ e $s_2 = Marcio$, utilizando *JaroWinkler*, tem um escore de $g = 0,9333333$, já utilizando *Levenshtein*, retorna o escore $g = 0,6666666$. Utilizar escore ajustado é uma solução aceitável, pois estes são sempre (independente da função de similaridade) gerados a partir da mesma medida, a precisão; portanto, são considerados combináveis.

Foram escolhidos quatro métodos de combinação de escore: (i) média aritmética, (ii) média ponderada, (iii) peso automático e (iv) árvore de decisão. A qualidade do casamento quando escores retornados pelas funções de similaridades são diretamente combinados é comparada com a qualidade do casamento quando o escore ajustado é combinado. A qualidade é medida pela clássica curva de precisão x revocação (BAEZA-YATES; RIBEIRO-NETO, 1999; MANNING; RAGHAVAN; SCHÜTZE, 2008).

4.3.1 Bases de Dados e funções de similaridade utilizadas

Para estes experimentos foram utilizadas quatro bases de dados, todas contendo dados em inglês. Três bases Febrl (denominadas Febrl-2, Febrl-1 e Febrl-3) e uma base de dados Cora. As bases de dados Febrl, são coleções artificiais geradas pela ferramenta *Febrl*. As bases Febrl-1 e Febrl-3 foram criadas apenas para pequenos testes. Todas as três bases Febrl possuem registros com dados pessoais, Febrl-1 possui 1.000 registros distintos, onde

100 representam pessoas diferentes, tendo em média 10 duplicatas por registro. Cada registro contém quatro atributos: *given name* (nome da pessoa), *surname* (sobrenome), *date of birth* (data de nascimento) e *ssid* (id de segurança social). A base de dados Febrl-3 também possui 1.000 registros, sendo esses 90 originais e 910 duplicatas, cada registro também possui quatro atributos, porém esta base foi gerada com seis atributos onde alguns foram concatenados. Esta possui os seguintes atributos: *given name + surname*, *street number + address* (nome da rua e número da casa), *postal code* (código postal) e *phone number* (número do telefone).

Já a base Febrl-2, que foi utilizada na maior parte dos experimentos, contém 10.000 registros distintos representando 1.000 pessoas diferentes, possuindo assim uma média de 10 registros duplicados por pessoa. Cada registro desta base contém os seguintes atributos: *ssid*, *given name*, *surname*, *address* dividido em dois campos (*address1* e *address2*), *suburb* (nome do bairro), *street number*, *postal code* (código postal), *date of birth* e *phone number*.

Os processos de treinamento para as bases de dados Febrl (1, 2 e 3) foram feitos como descrito a seguir. Foi utilizada novamente a ferramenta *Febrl*, agora gerando dez pequenos conjuntos de dados (aproximadamente 500 valores), um para cada domínio dos atributos das bases. Para os atributos compartilhados entre as três bases de dados (*given name*, *surname*, *date of birth*, *ssid*, *postal code* e *phone number*), apenas um conjunto de dados foi gerado para o treinamento, pois como demonstrado nos experimentos anteriores, o treinamento pode ser efetuado em qualquer base de dados do mesmo domínio. Para a base Febrl-3 que contém os atributos *given name + surname* e *street number + address*, foram concatenadas as bases de treinamento *given name* com *surname* e *street number* com *address1*, com o objetivo de obter bases do mesmo domínio. O processo de treinamento descrito na subseção 4.1.2 foi executado para cada uma dessas bases de dados empregando funções de similaridades escolhidas por um especialista. Para a base de dados Febrl-2, foram utilizados dois conjuntos de funções de similaridade, para que se possa observar se o escore ajustado continua melhorando se forem utilizadas outras funções. Esses dois conjuntos de funções podem ser observadas: na tabela 4.13, o primeiro conjunto e, na tabela 4.14, o segundo conjunto de funções. As funções de similaridade utilizadas para a base de dados Febrl-1 são apresentadas na tabela 4.12 e a tabela 4.15 mostra as funções utilizadas para a base Febrl-3. Este processo resulta em uma TaME (Definição 4) para cada domínio de atributos das bases de dados. Para os atributos *given name*, *surname*, *date of birth*, *ssid*, *postal code* e *phone number*, que aparecem em mais de uma base de dados, apenas um treinamento foi efetuado, pois utilizam a mesma função de similaridade, com exceção do segundo conjunto de funções de similaridade da base Febrl-2 (tabela 4.14), pois as funções de similaridade de alguns atributos foram modificadas. Para estes, um novo treinamento foi efetuado.

Em resumo, para o treinamento das bases Febrl (1, 2 e 3), foram criadas 10 pequenas bases de dados para o treinamento, onde a partir destas, mais duas foram criadas pela concatenação das bases *given name* com *surname* e *street number* com *address1*, resultando em 17 tabelas TaME.

A base de dados Cora é um conjunto de dados reais de referências bibliográficas em inglês, obtidos do projeto Cora (MCCALLUM; NIGAM; UNGAR, 2000). A base de dados Cora contém 1.879 registros com 15 atributos, *author*, co-*author1*, *co-author2*, *title*, *year*, *journal*, *book title*, *volume*, *pages*, *publisher*, *address*, *editor*, *note*, *institution* e *type*.

Para os atributos *title* e *journal* foram utilizadas as TaME's calculadas no treinamento

Atributo	Função de Similaridade
ssid	Edit Distance
given name	Q-GramSimilarity
surname	Q-GramSimilarity
date of birth	Edit Distance

Tabela 4.12: Funções de similaridade utilizadas para os atributos da base de dados Febrl-1

Atributo	Função de Similaridade
ssid	Edit Distance
given name	Q-GramSimilarity
surname	Q-GramSimilarity
address 1	JaroWinkler
address 2	JaroWinkler
suburb	JaroWinkler
street number	Edit Distance
postal code	Edit Distance
date of birth	Edit Distance
phone number	Edit Distance

Tabela 4.13: Conjunto 1 de funções de similaridade utilizadas para cada atributo da base de dados Febrl-2

descrito na subseção 4.1.2 para estes atributos. Para o restante dos atributos, uma amostra de 40 consultas de cada atributo foi randomicamente sorteada e o processo de treinamento foi executado com estas amostras. Para *co-author1* e *co-author2* foram utilizadas as

Atributo	Função de Similaridade
ssid	Edit Distance
given name	JaroWinkler
surname	JaroWinkler
address 1	Q-GramSimilarity
address 2	Q-GramSimilarity
suburb	Q-GramSimilarity
street number	Edit Distance
postal code	Edit Distance
date of birth	Edit Distance
phone number	Edit Distance

Tabela 4.14: Conjunto 2 de funções de similaridade utilizadas para cada atributo da base de dados Febrl-2

Atributo	Função de Similaridade
given name + surname	Q-GramSimilarity
street number + address	JaroWinkler
postal code	Edit Distance
phone number	Edit Distance

Tabela 4.15: Funções de similaridade utilizadas para os atributos da base de dados Febrl-3

TaME's que resultaram do processo de treinamento do atributo *author*. As funções de similaridade utilizadas nesta base de dados são apresentadas na tabela 4.16.

Atributo	Função de Similaridade
author	Q-GramSimilarity
co-autor1	Q-GramSimilarity
co-author2	Q-GramSimilarity
title	Q-GramSimilarity
year	Edit Distance
journal	JaroWinkler
booktitle	Q-GramSimilarity
volume	Edit Distance
pages	Edit Distance
publisher	Q-GramSimilarity
address	JaroWinkler
editor	Q-GramSimilarity
note	Q-GramSimilarity
institution	JaroWinkler
type	Jaccard

Tabela 4.16: Funções de similaridade utilizadas para cada atributo da base de dados Cora

4.3.2 Métodos de combinação de escore para casamento de registros

Neste experimento, foram utilizados os seguintes métodos para calcular a similaridade entre dois registros r_1 e r_2:

1. *Média Aritmética*
   Este método corresponde ao método clássico de casamento de registros (FELLEGI; SUNTER, 1969), porém, ao invés de somar os escores dos atributos, é feita a média aritmética deles.
   Aqui, este método é utilizado para fazer a combinação do escore original e também do escore ajustado. Ele é definido a seguir:

   Definição 6 *(Similaridade média) Sejam* $r_1 = (a_{11}, a_{12}, \ldots, a_{1n})$ *e* $r_2 = (a_{21}, a_{22}, \ldots, a_{2n})$ *dois registros com* n *elementos cada um. Seja* f_i, $i = 1, 2, \ldots, n$ *a função de similaridade que é adequada para calcular a similaridade entre os valores* a_{1i} *e* a_{2i}*. Seja* α_M *a função de mapeamento do escore original para o escore ajustado, conforme descrito na definição 5.*

   O método de similaridade média utilizando escore original é definida como:

$$sim_{avgOriginal}(r_1, r_2) = \frac{\sum_{i=1}^{n} f_i(a_{1i}, a_{2i})}{n} \tag{4.3}$$

   O método de similaridade média utilizando escore ajustado é definida como:

$$sim_{avgAjustado}(r_1, r_2) = \frac{\sum_{i=1}^{n} \alpha_M(f_i(a_{1i}, a_{2i}))}{n} \tag{4.4}$$

A equação 4.3 faz a média aritmética dos escores originais retornados diretamente pelas funções de similaridade, já a equação 4.4 efetua a média substituindo o escore original pelo respectivo escore ajustado presente na sua TaME.

2. *Média Ponderada*
Este método é consistente com várias abordagens na literatura (por exemplo (BILENKO et al., 2003; CARVALHO et al., 2006)) que aplicam pesos para cada um dos atributos, com objetivo de dar maior ou menor importância para os mesmos.
Neste experimento, utilizaremos este método para combinar os escores originais e também os escore ajustados, e chamaremos esse método de abordagem de pesos manuais. Tal abordagem é definida abaixo.

Definição 7 *(Abordagem de pesos manuais)*

Sejam $r_1 = (a_1, a_2, \ldots, a_n)$ *e* $r_2 = (a_1, a_2, \ldots, a_n)$ *dois registros com* n *atributos cada um,* f_i*,* $i = (1, 2, \ldots, n)$*, a função de similaridade adequada para o domínio do atributo* a_n*,* α_M *a função de mapeamento do escore original para o escore ajustado (definição 5) e* p_i *o peso selecionado para o atributo* a_n*.*

A abordagem de peso utilizando escore original é definida como:

$$sim_{pesoOriginal}(r_1, r_2) = \frac{\sum_{i=1}^{n}(p_i * f_i(r_1 a_i, r_2 a_i))}{\sum_{i=1}^{n} p_i} \tag{4.5}$$

A abordagem de peso utilizando o escore ajustado é definida como:

$$sim_{pesoAjustado}(r_1, r_2) = \frac{\sum_{i=1}^{n}(p_i * \alpha_M(f_i(r_1 a_i, r_2 a_i)))}{\sum_{i=1}^{n} p_i} \tag{4.6}$$

Esses pesos são geralmente definidos por um especialista no domínio. No entanto, eles também podem ser obtidos através de alguma estatística sobre os dados ou por algum método de aprendizado de máquina, como descrito nos trabalhos relacionados (BILENKO et al., 2003; CARVALHO et al., 2006).

Neste experimento, o foco não é encontrar os melhores pesos para os atributos, o objetivo é simplesmente demonstrar que utilizando o escore ajustado, a qualidade do processo de casamento melhora.

Para o experimento com esta abordagem de pesos manuais, foram utilizadas as bases de dados Febrl-2 e Cora. Os pesos foram escolhidos arbitrariamente, como descritos a seguir:

- Para a base de dados Febrl-2:
  - Peso $p_i = 3$ para os atributos: *given name, surname, address1, suburb name* e *ssid*;
  - Peso $p_i = 1$ para os atributos: *address2, street number, postal code, date of birth* e *phone number*.

- Para a base de dados Cora:

  - Peso $p_i = 3$ para os atributos: *author, co-autor1* e *title*;
  - Peso $p_i = 2$ para os atributos: *co-author2* e *year*;
  - Peso $p_i = 1$ para os atributos: *journal, booktitle, volume, pages, publisher, address, editor, note, institution* e *type*.

3. *Peso Automático*
   Este método foi desenvolvido com base no IDF (*Inverse Document Frequency*) (BAEZA-YATES; RIBEIRO-NETO, 1999) e calcula pesos automaticamente para cada atributo da base de dados, beneficiando os atributos que contém maior número de objetos distintos.

   Definição 8 *(Abordagem de pesos automáticos)*

   Seja $R = \{r_1, r_2, \ldots, r_m\}$ um conjunto de m registros contidos em uma base dados e $r_i \in R | r_i = (a_1, a_1, \ldots, a_n)$ um registro pertencente a R com n atributos a, $v[a_i]$ o valor do i-ésimo atributo, f_i, $i = (1, 2, \ldots, n)$, a função de similaridade adequada para o atributo a_i e α_M a função de mapeamento do escore original para o escore ajustado (definição 5). A equação abaixo define a função $p_{aut}(a_i)$ que calcula automaticamente o peso de cada atributo $a_i \in R$.

$$p_{aut}(a_i) = \frac{v[a_i]}{|R|} \tag{4.7}$$

   O peso para cada atributo é definido pela razão entre número de valores distintos deste atributos e a totalidade de registros da base de dados.

   A abordagem de pesos automáticos utilizando escore original é definida como:

$$sim_{IDF-Original}(r_1, r_2) = \frac{\sum_{i=1}^{n}(p_{aut}(a_i) * f_i(r_1 a_i, r_2 a_i))}{\sum_{i=1}^{n} p_{aut}(a_i)} \tag{4.8}$$

   A abordagem de pesos automáticos utilizando escore ajustado é definida como:

$$sim_{IDF-Ajustado}(r_1, r_2) = \frac{\sum_{i=1}^{n}(p_{aut}(a_i) * \alpha_M(f_i(r_1 a_i, r_2 a_i)))}{\sum_{i=1}^{n} p_{aut}(a_i)} \tag{4.9}$$

   Esta abordagem é utilizada apenas para que se possa observar o comportamento do escore ajustado com a utilização de pesos calculados de forma automática, sem a intervenção do especialista, não tendo intenção alguma de comparar resultados com outras abordagens existentes.

4. *Árvore de Decisão*
   Um outro método para a ponderação da importância dos atributos é a árvore de decisão (SARAWAGI; BHAMIDIPATY, 2002; TEJADA; KNOBLOCK; MINTON, 2001). Neste experimento foram utilizadas as bases Febrl-2 e Cora. As árvores de

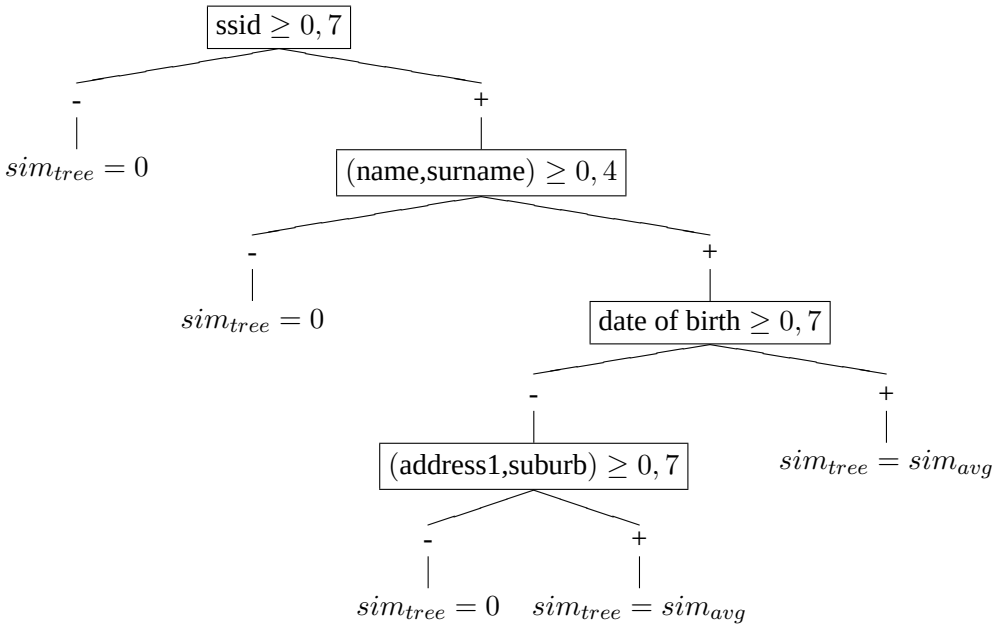

Figura 4.9: Árvore de decisão para calcular a similaridade entre registros na base de dados Febrl-2

decisão utilizadas são apresentadas na figura 4.9, para a base Febrl-2, e, na figura 4.10, para base de dados Cora. As árvores de decisão para o escore ajustado e para o escore original são quase iguais, muda somente no momento em que a média aritmética é calculada, sim_{avg} nas figuras 4.9 e 4.10. Neste momento, para escore original, utiliza-se a função $sim_{avgOriginal}$, definida na equação 4.3, e, para o escore ajustado, utiliza-se a função $sim_{avgAjustado}$, definida na equação 4.4.

A árvore de decisão define a similaridade entre dois registros iniciando pelo nó raiz e percorrendo a árvore para baixo até chegar a um nó folha, o qual especifica o escore de similaridade para o determinado par de registros. Cada nó da árvore contém um teste a ser executado em alguns atributos, e cada galho (*branch*) é classificado com um possível resultado do teste (+ e -, sendo verdadeiro ou falso respectivamente).

A árvore da figura 4.9 especifica que o atributo ssid dos registros é comparado primeiro. Se o escore de similaridade é menor que 0.7, a árvore diz que o registro representa duas pessoas diferentes e o resultados do escore de similaridade é retornado como zero. No entanto, se o valor de similaridade entre os valores de ssid é maior ou igual a 0.7, o processo continua pela comparação dos atributos *given name* and *surname*. Se o valor de similaridade média entre esses atributos é menor do que 0.4, as tuplas representam pessoas diferentes. Mas, se a média entre os escores de similaridade destes atributos for maior ou igual a 0.4, o processo continua testando o atributo *date of birth*. Se o escore de similaridade entre os atributos *date of birth* for maior ou igual a 0.7, a árvore decide que a similaridade entre os registros é sim_{avg} ($sim_{avgAjustado}$ para o escore ajustado ou $sim_{avgOriginal}$ para o escore original). E, por último, se *date of birth* for menor que 0.7, os atributos *address1* e *suburb name* são testados. Se o resultado da média da similaridade entre os atributos *address1* e *suburb name* for maior do que 0.7, a árvore decide que os dois registros representam a mesma pessoa, e o escore de similaridade e cal-

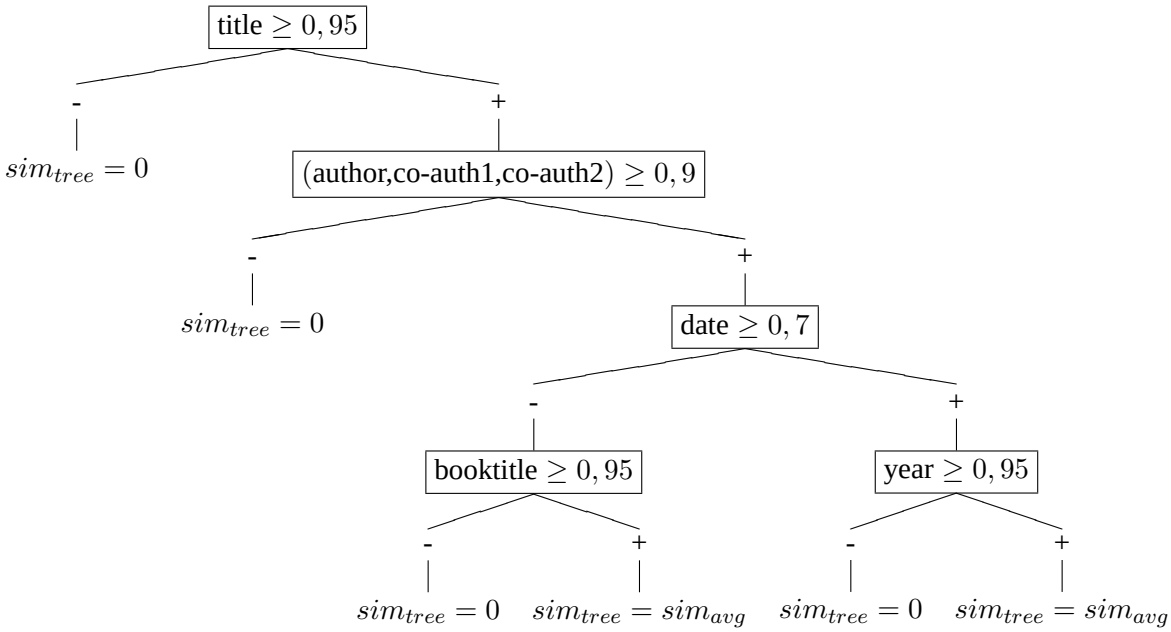

Figura 4.10: Árvore de decisão para calcular a similaridade entre registros na base de dados Cora

culado pela função sim_{avg}, ao contrário, se a similaridade média for inferior a 0.7 a árvore julga os dois registros como duas pessoas diferentes e retorna o escore de similaridade zero.

É importante lembrar que este processo é feito tanto para o escore original, quando para o escore ajustado, para que se possa comparar seus resultados. Quando o escore ajustado é utilizado, todos os testes nos nós da árvore são feitos com o escore ajustado, por exemplo, no primeiro nó testa se o ssid é maior ou igual a 0.7, esse teste é feito com o escore ajustado entre os atributos ssid que estão sendo testados e quando a árvore decide que os dois registros representam a mesma pessoa a função $sim_{avgAjustado}$ é utilizada. Quando o escore original é utilizado, os testes são feitos utilizando os seus resultados, e a função sim_{avg} a ser utilizada é a $sim_{avgOriginal}$.

O tipo de árvore de decisão utilizada aqui é diferente das apresentadas em (SARAWAGI; BHAMIDIPATY, 2002; TEJADA; KNOBLOCK; MINTON, 2001). Nestas abordagens, o resultado da árvore de decisão é um valor booleano (casa ou não casa), enquanto neste experimento, o resultado da árvore é um valor de similaridade. Optou-se por esta alternativa para que se possa ranquear os registros casados, a fim de obter valores mais precisos de revocação e precisão.

Observa-se ainda que uma árvore de decisão para ser empregada em uma aplicação depende de vários parâmetros: os atributos a serem testados, a ordem dos testes, o valor de limiar para cada teste e como calcular o valor que é utilizado como um escore de similaridade final entre os registros (média aritmética, média ponderada, etc). Para este experimento, foram manualmente definidas estas árvores porque refletem o conhecimento que se tem sobre quais atributos são mais significantes para identificar a pessoa ou a referência bibliográfica. Os valores de limiar foram definidos por tentativa e erro. É importante lembrar que, estes experimentos têm como objetivo demonstrar que os escores ajustados melhoram os resultados das técnicas de similaridade existentes, e não tentar encontrar a árvore de decisão mais

adequada. Em uma aplicação real, a árvore de decisão poderia ser construída através de métodos muito mais precisos e sofisticados, como utilizado em (TEJADA; KNOBLOCK; MINTON, 2001).

4.3.3 Combinando as similaridades dos atributos para casamento de registros

Nesta seção, serão apresentados os processos de casamento de registros utilizando cada uma das abordagens descritas na subseção anterior. Os processos de casamento e os resultados serão apresentados separadamente para cada método. Primeiramente, os experimentos realizados utilizando a média aritmética, seguidos pelos experimentos utilizando a média ponderada. Após, os resultados da utilização dos pesos automáticos e, por último, os experimentos utilizando árvore de decisão.

1. *Similaridade Média*
   Os experimentos com esta abordagem utilizaram as bases de dados Febrl-1, Febrl-2 e Cora. A base de dados Febrl-2 foi utilizada duas vezes com esta abordagem, a primeira utilizando o conjunto de funções de similaridade apresentadas na tabela 4.13 e a segunda utilizando outro conjunto de funções de similaridade apresentado na tabela 4.14. O objetivo de efetuar o processo de casamento, com funções de similaridades diferentes, é analisar se o escore ajustado melhora os resultados, mesmo utilizando outras funções de similaridade. Para cada base de dados, foram executados os seguintes passos:

   (a) 40 registros de consulta são sorteados randomicamente da base de dados $q = \{q_1, q_2, \ldots, q_{40}\}$.

   (b) Para cada consulta q_i (das 40 sorteadas), os registros da base de dados são ranqueados de acordo com a sua similaridade com o registro consulta. A similaridade é calculada pela função $sim_{avgOriginal}$ (equação 4.3), que faz a média aritmética dos escores retornados pelas funções de similaridade utilizadas. As funções utilizadas para cada atributo são apresentadas nas tabelas 4.13, 4.14, 4.12 e 4.16.

   (c) Outro *ranking* é calculado para cada uma das 40 consultas, só que agora ao invés de utilizar a função $sim_{avgOriginal}$ é utilizada a função $sim_{avgAjustado}$ (equação 4.4), que faz a média aritmética dos escores ajustados, obtidos pela TaME, calculada no processo de treinamento, de cada atributo.

   (d) Precisão e revocação são calculados pelo processo padrão de IR (recuperação de informação) (BAEZA-YATES; RIBEIRO-NETO, 1999; MANNING; RAGHAVAN; SCHÜTZE, 2008), e os resultados são plotados em duas curvas de precisão x revocação, uma para os resultados utilizando o escore original ($sim_{avgAjustado}$) e a outra utilizando escore ajustado ($sim_{avgAjustado}$).

   Este processo gerou um gráfico contendo duas curvas de precisão x revocação para cada base de dados. Uma curva com os resultados da utilização do escore ajustado e a outra com os resultados do escore original. Tais gráficos são apresentados, na figura 4.13, com os resultados para a base Febrl-1, na figura 4.11, com os resultados para a base Febrl-2, utilizando o conjunto de funções de similaridade da tabela 4.13, na figura 4.12, com as funções da tabela 4.14, e, na figura 4.14, apresentando os resultados obtidos utilizando a base de dados Cora. Nestes gráficos, a linha

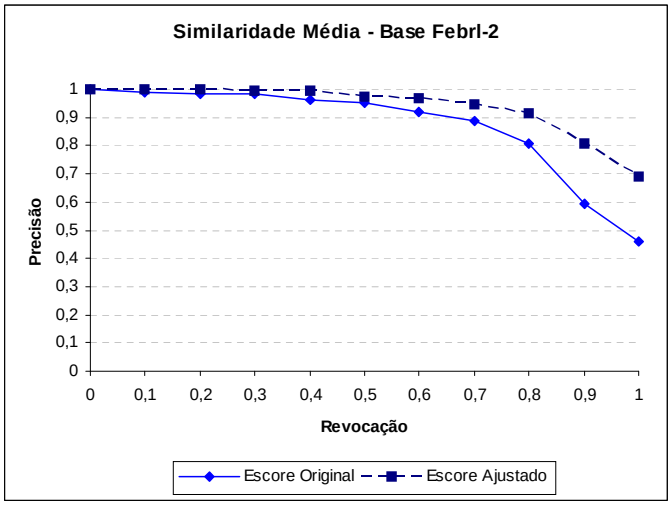

Figura 4.11: Curva de Precisão x Revocação utilizando similaridade média com a base de dados Febrl-2 e as funções de similaridade da tabela 4.11

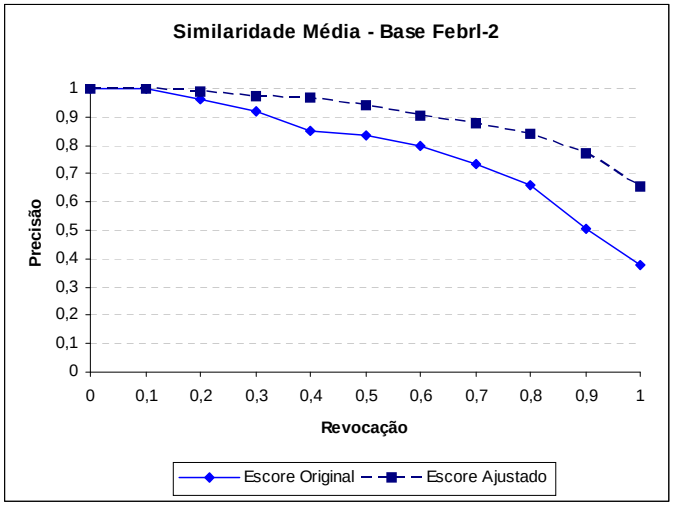

Figura 4.12: Curva de Precisão x Revocação utilizando similaridade média com a base de dados Febrl-2 e as funções de similaridade da tabela 4.12

tracejada representa a curva de precisão x revocação utilizando o escore ajustado, e a linha contínua representa a curva utilizando o escore original.

O primeiro experimento, utilizando combinação de escore para casamento de registros, foi realizado com a base de dados Febrl-1 (figura 4.13), com objetivo de observar se o escore ajustado funcionava bem neste processo. O resultado pode ser observado na figura 4.13. Onde a qualidade dos resultados utilizando escore ajustado (linha tracejada) é melhor do que com a utilização escore original (linha contínua). Tendo este primeiro resultado, o próximo passo foi criar uma base de dados maior, pois a Febrl-1 possui apenas mil registros e quatro atributos. Para isso, foi criada a base Febrl-2, contendo dez mil registros e dez atributos.

Os resultados deste segundo experimento, utilizando as funções de similaridade da tabela 4.13, também foram satisfatórios, como pode ser observado na figura 4.11, onde a linha pontilhada (escore ajustado) apresenta um resultado melhor do

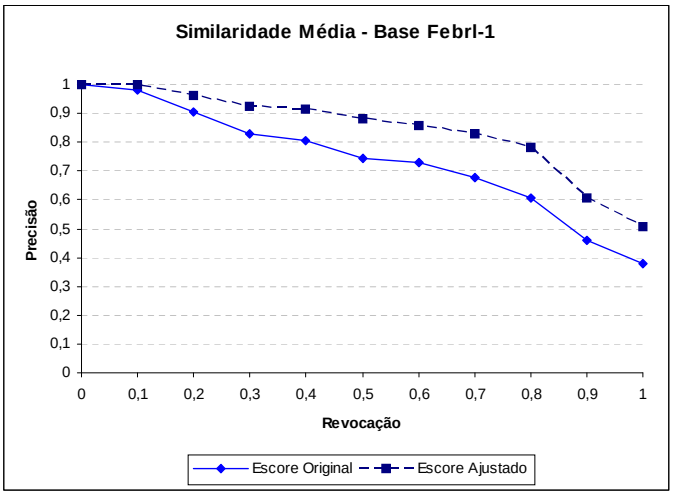

Figura 4.13: Curva de Precisão x Revocação utilizando a base de dados Febrl-1 com a abordagem de similaridade média

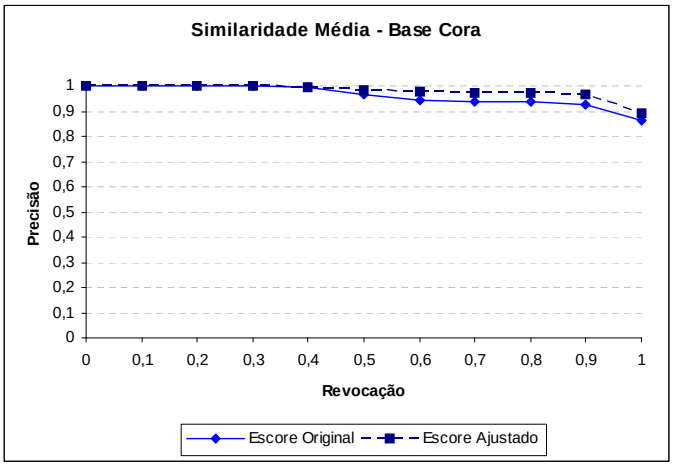

Figura 4.14: Curva de Precisão x Revocação utilizando a base de dados Cora com a abordagem de similaridade média

que a linha contínua (escore original). Neste momento, surgiu uma dúvida, se ao utilizar outras funções de similaridade, o escore ajustado segue retornando melhores resultados do que o escore original? Para isso, foram selecionadas outras funções de similaridade que funcionassem razoavelmente bem nos domínios dos atributos, tais funções são as apresentadas na tabela 4.14.

Analisando o gráfico 4.12, pode-se observar que mesmo com a utilização de outras funções de similaridade, o escore ajustado segue melhorando os resultados. Comparando os gráficos 4.11 e 4.12 percebe-se que os resultados são bem parecidos, porém as curvas do segundo gráfico apresentam resultados piores do que as mesmas do primeiro gráfico. Isto acontece porque as funções utilizadas no primeiro, são mais adequadas para estes domínios do que as funções utilizadas no segundo. No entanto, um ponto importante a observar é que mesmo com o escore original piorado, o escore ajustado segue melhorando o resultado, como pode ser visto no ponto $0,4$ de revocação do gráfico 4.12, a curva do escore original tem uma queda, mas a curva do escore ajustado segue bem próxima de 100% de precisão.

O último experimento, realizado com esta abordagem, foi utilizando a base contendo dados reais. Pois, nos experimentos anteriores, as bases utilizadas foram bases artificiais. Como dito anteriormente, a base com dados reais utilizada foi a Cora. Os resultados obtidos com esta base também foram satisfatórios, como pode ser visto no gráfico da figura 4.14. Analisando os resultados, percebe-se que o escore original já retorna resultados bons para essa base, não tendo muito que melhorar, mas mesmo assim, com a utilização do escore ajustado (linha tracejada) a qualidade dos resultados melhora.

2. *Abordagem de Pesos Manuais*
   Os experimentos realizados com esta abordagem utilizam as bases de dados Febrl-2 e Cora. A base de dados Febrl-2 foi utilizada com dois conjuntos de funções de similaridade, da mesma forma que no experimento com a similaridade média, tendo como objetivo analisar o comportamento do escore ajustado, utilizando esta abordagem com algumas funções de similaridades diferentes.

   Para este experimento, foi escolhido manualmente um peso para cada atributo das bases de dados (estes pesos foram apresentados na subseção anterior). Ao executar os experimentos com esta abordagem, foram seguidos os seguintes passos para cada base de dados:

   (a) 40 registros de consulta são sorteados randomicamente de cada base de dados $q = \{q_1, q_2, \ldots, q_{40}\}$.

   (b) Para cada uma das 40 consultas sorteadas, é calculada a similaridade dos registros da base de dados, utilizando a função $sim_{pesoOriginal}$ (equação 4.5) com os pesos selecionados manualmente. Então, os registros são ranqueados de acordo com o seu resultado de similaridade.

   (c) Novamente, para cada uma das 40 consultas, é feito o mesmo processo anterior, só que utilizando a função $sim_{pesoAjustado}$ (equação 4.6) ao invés da $sim_{pesoOriginal}$. Formando assim, *rankings* com resultados do escore ajustado.

   (d) Como as bases de dados já são avaliadas, a precisões e revocações são calculadas automaticamente para cada *ranking* criado.

   (e) Por fim, são geradas duas curvas de precisão x revocação para cada base de dados. Uma curva representando os resultados utilizando o escore ajustado, e a segunda representando os resultados do escore original.

   As curvas geradas neste processo foram plotadas em gráficos, e são apresentadas, na figura 4.15, os resultados com a base de dados Febrl-2, utilizando o conjunto 1 de funções de similaridade (tabela 4.13), na figura 4.16, também os resultados para a base Febrl-2, só que utilizando o conjunto 2 de funções de similaridade (tabela 4.14). E, na figura 4.17, são apresentados os resultados da utilização de pesos manuais na base de dados Cora.

   Ao analisar os gráficos com os resultados deste método de pesos manuais, é possível observar que o escore original segue melhorando a qualidade dos resultados obtidos pelas funções de similaridade. O gráfico da base Cora (figura 4.17) apresenta resultados onde o escore original possui uma qualidade muito boa. No entanto, mesmo com o escore original bom, o escore ajustado ainda consegue melhorar um pouco

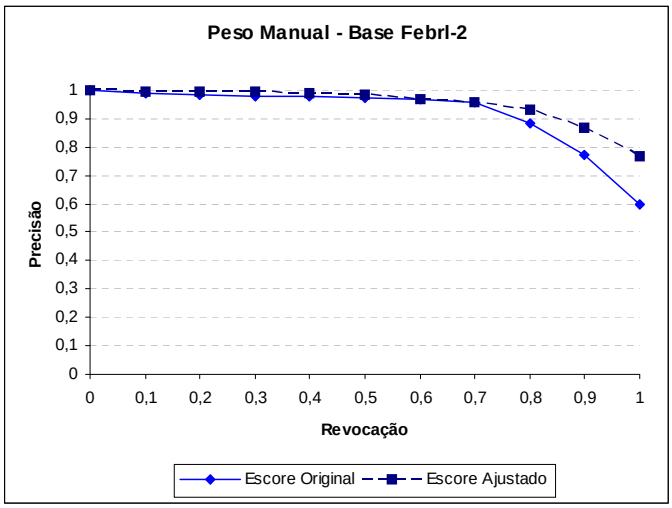

Figura 4.15: Curva de Precisão x Revocação utilizando a base de dados Febrl-2 com abordagem de peso manual e as funções de similaridade da tabela 4.11

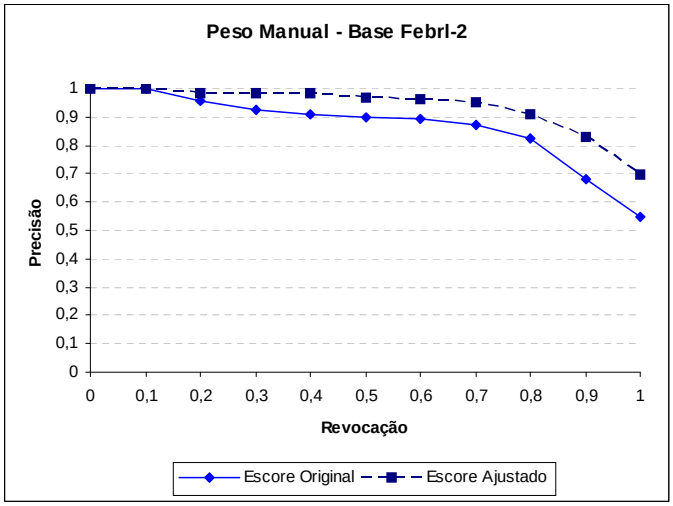

Figura 4.16: Curva de Precisão x Revocação utilizando a base de dados Febrl-2 com abordagem de peso manual e as funções de similaridade da tabela 4.12

sua qualidade. Para a base Febrl-2, os resultados utilizando tanto o primeiro conjunto de funções de similaridade (figura 4.17) quanto o segundo conjunto (figura 4.16) apresentam melhoras na qualidade dos resultados quando o escore ajustado é aplicado. Comparando estes dois gráficos, é possível perceber que os resultados obtidos com o escore original, do gráfico 4.16, pioram em comparação com o gráfico 4.15. Isso acontece porque as funções de similaridade escolhidas não são tão boas para o domínio. Porém, é possível observar que as curvas de escore ajustado permanecem bem próximas, demonstrando que a qualidade melhora mesmo quando as funções não são as mais adequadas.

3. *Abordagem de Pesos Automáticos*
   Para os experimentos utilizando pesos automáticos, foram utilizadas todas as quatro bases de dados, que são: Febrl-1, Febrl-2, Febrl-3 e Cora. É importante salientar bem que, com a utilização desta abordagem se tem apenas o objetivo de observar o

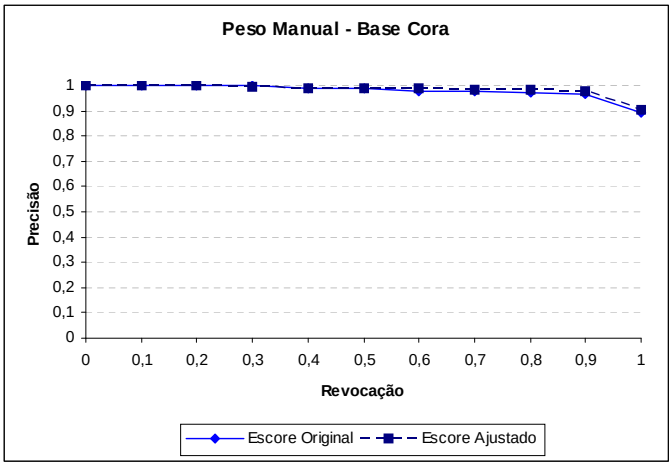

Figura 4.17: Curva de Precisão x Revocação utilizando a base de dados Cora com abordagem de peso manual

comportamento do escore ajustado com pesos calculados de forma automática, sem que ninguém precise interferir, não tendo nenhuma intenção de comparar resultados com outras técnicas.

As funções de similaridade utilizadas são discutidas na subseção 4.3.1. As quais são apresentadas, na tabela 4.12, para a base de dados Febrl-1, tabela 4.13, para a base de dados Febrl-2, tabela 4.15, para base Febrl-3 e, para a base Cora, as funções são apresentadas na tabela 4.16. Para a experimentação desta abordagem, foram seguidos os seguintes passos para cada base de dados:

(a) 40 registros de consulta são sorteados randomicamente $q = \{q_1, q_2, \ldots, q_{40}\}$.

(b) O cálculo do peso de cada atributo é calculado, conforme a equação 4.7.

(c) Para cada uma das 40 consultas sorteadas, é calculada a similaridade dos registros da base de dados, utilizando a função $sim_{IDF-Original}$ (equação 4.8) com os pesos calculados no passo anterior. Após, os registros são ranqueados de acordo com o seu resultado de similaridade.

(d) Novamente, para cada uma das 40 consultas, é feito o mesmo processo anterior, só que utilizando a função $sim_{IDF-Ajustado}$ (equação 4.9). Formando assim, *rankings* com resultados do escore ajustado.

(e) Como as bases de dados já são avaliadas, a precisões e revocações são calculadas automaticamente para cada *ranking* criado.

(f) Por fim, são geradas duas curvas de precisão x revocação para cada base de dados. Uma curva representando os resultados com escore ajustado, e a segunda representando os resultados com escore original.

Os resultados obtidos com a utilização dos pesos automáticos podem ser visualizados nas figuras 4.18, 4.19, 4.20 e 4.21. A figura 4.18 apresenta os resultados com a base de dados Febrl-1, a 4.19 contém os resultados para a base Febrl-2, já a 4.20 contém os resultados para a base Febrl-3, e por último, a figura 4.21 ilustra os resultados para a base de dados Cora.

Como pode ser observado nas figuras, o escore original melhora a qualidade dos resultados utilizando pesos calculados de forma automática. Nas figuras 4.18 e

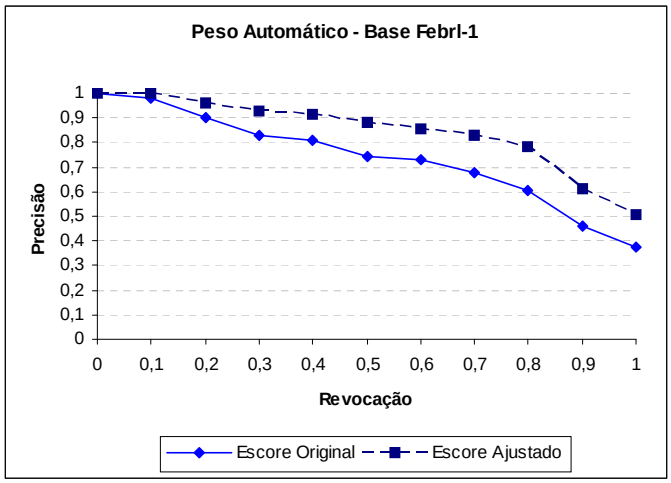

Figura 4.18: Curva de Precisão x Revocação utilizando a base de dados Febrl-1 com abordagem de peso automático

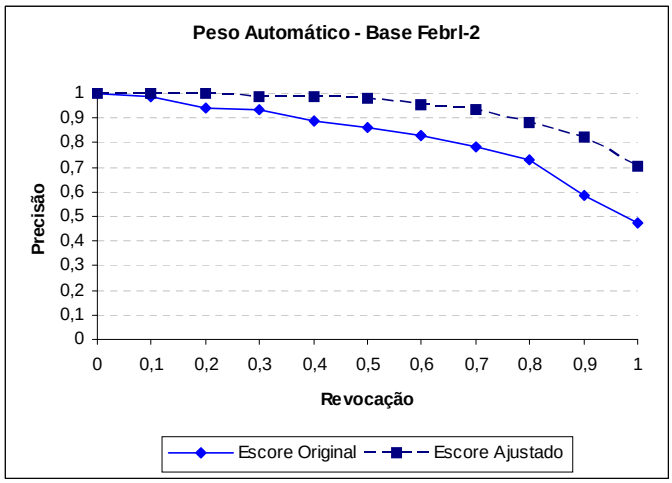

Figura 4.19: Curva de Precisão x Revocação utilizando a base de dados Febrl-2 com abordagem de peso automático

4.19, essa melhora pode ser vista claramente. Já, nas figuras 4.20 e 4.21, essa melhora é pequena, visto que, o resultado do escore original já está muito bom para estas bases. A curva com o resultado do escore original da base de dados Febrl-3 está quase sempre em 100% de precisão, baixando apenas do ponto $0,1$ de revocação em diante, não tendo realmente o que melhorar, porém, a curva do escore ajustado é uma curva de precisão x revocação quase ideal, onde a ideal seria uma reta sempre no ponto 1 de precisão, ou seja ter 100% de precisão com 100% de revocação.

Com estes resultados, se pode dizer que a utilização escore ajustado melhora a qualidade do processo de casamento, porém, em alguns casos a sua melhora pode não ser significante, visto que o resultado original já possui alta qualidade. Em resumo, o escore ajustado melhora os resultados quando possível, caso contrário fica igual, não diminuindo a qualidade.

4. *Árvore de Decisão*

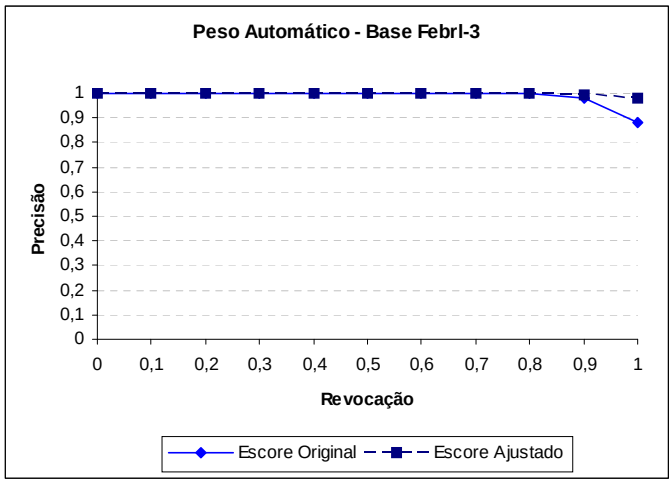

Figura 4.20: Curva de Precisão x Revocação utilizando a base de dados Febrl-3 com abordagem de peso automático

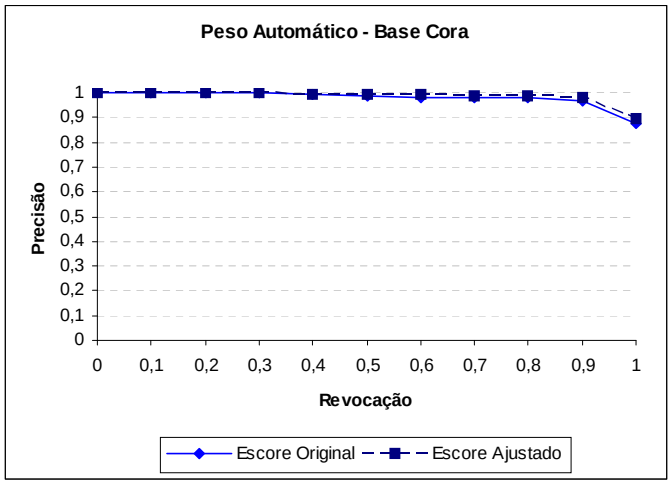

Figura 4.21: Curva de Precisão x Revocação utilizando a base de dados Cora com abordagem de peso automático

Os experimentos desenvolvidos com a utilização de árvore de decisão utilizaram as bases de dados Febrl-2 e Cora. Da mesma maneira que nos experimentos com precisão média e com pesos manuais, a base Febrl-2 foi utilizada com dois conjuntos de funções de similaridade, o conjunto 1 (tabela 4.13) e o conjunto 2 (tabela 4.14).

As árvores de decisão foram montadas manualmente, como apresentado na subseção 4.3.2. A árvore de decisão para a base de dados Febrl-2 pode ser visualizada na figura 4.9 e a árvore para a base cora na figura 4.10. É importante lembrar que estas árvores de decisões foram criadas apenas para observar o comportamento do escore ajustado com este tipo de abordagem, não tendo intenção alguma de criar a melhor árvore de decisão para os domínios, ou até mesmo comparar com os resultados de outros métodos.

Para a execução dos experimentos, foram seguidos os seguintes passos para cada uma das bases de dados:

(a) 40 registros de consulta foram sorteados randomicamente $q =$

$\{q_1, q_2, \ldots, q_{40}\}$.

(b) Para cada uma das 40 consultas sorteadas, é calculada a similaridade dos registros da base de dados, utilizando a árvore de decisão com os escores original. Todos os testes da árvore são efetuados com o valor do escore original entre os atributos, e quando a árvore decide que os dois registros são os mesmos, é utilizada a função $sim_{avgOriginal}$ para retornar escore de similaridade. Após os registros são ranqueados de acordo com o seu resultado de similaridade.

(c) Para cada uma das 40 consultas, é feito novamente o mesmo processo anterior, só que utilizando o valor do escore ajustado nos testes dos nós da árvore. No final do processo, se a árvore decidir que os registros casam, a função $sim_{avgAjustado}$ é utilizada para retornar um valor de similaridade entre os registros. Por fim, os registros são ranqueados pelo seu valor de similaridade.

(d) O cálculo das precisões e revocações são efetuados automaticamente para cada *ranking* criado.

(e) Por último, são geradas duas curvas de precisão x revocação (com média dos resultados dos 40 *rankings*). Uma curva representando os resultados com escore ajustado, e a outra representando os resultados com escore original.

Os resultados dos experimentos utilizando esta abordagem de árvore de decisão são apresentados nos gráficos das figuras 4.22, 4.23 e 4.24. Onde as linhas tracejadas representam a curva de precisão x revocação com o escore ajustado e as linha contínuas representam a curva com o resultado do escore original. Na figura 4.22, os resultados apresentados são da base de dados Febrl-2 com o primeiro conjunto de funções de similaridade (tabela 4.13), na figura 4.23, também são apresentados os resultados para a base de dados Febrl-2, porém utilizando o conjunto 2 de funções de similaridade (tabela 4.14) e, na figura 4.24, são apresentadas as curvas com resultado do experimento utilizando a base Cora.

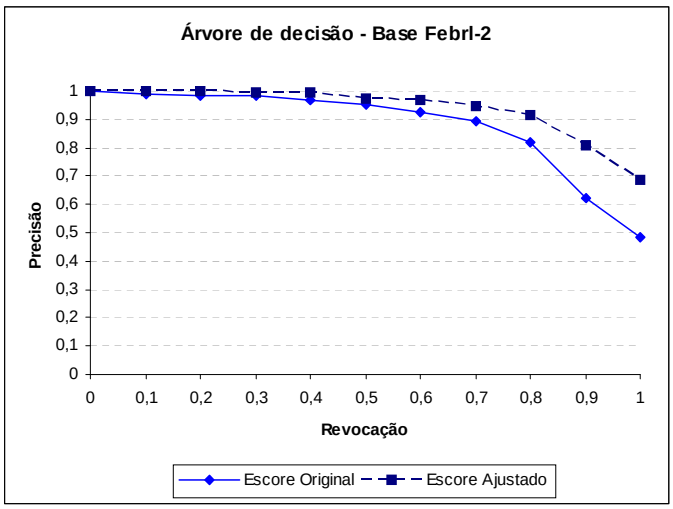

Figura 4.22: Curva de Precisão x Revocação utilizando a base de dados Febrl-2 com abordagem árvore de decisão e as funções de similaridade da tabela 4.11

As curvas destas figuras apresentam claramente que a utilização do escore ajustado com a abordagem árvore de decisão resulta em melhoras na qualidade do processo de casamento de registros. Nos experimentos anteriores, utilizando a média

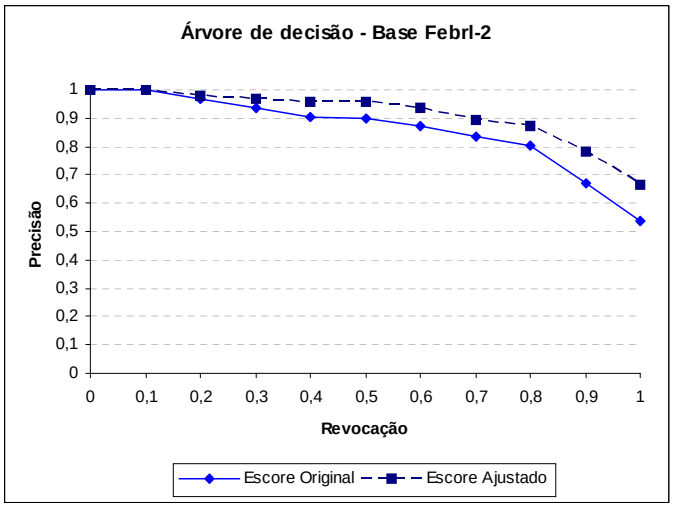

Figura 4.23: Curva de Precisão x Revocação utilizando a base de dados Febrl-2 com abordagem árvore de decisão e as funções de similaridade da tabela 4.12

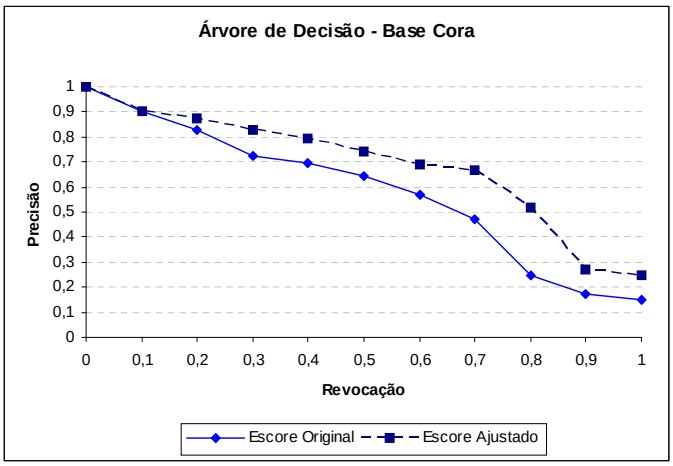

Figura 4.24: Curva de Precisão x Revocação utilizando a base de dados Cora com abordagem árvore de decisão

aritmética, pesos manuais e pesos automáticos, ficou difícil ver o comportamento do escore ajustado, pois os resultados do escore original já estavam bons, porém, no gráfico desta abordagem (figura 4.24) é possível visualizar melhor o comportamento, podendo dizer que o escore ajustado melhora os resultados também na base de dados Cora.

+

4.4 Influência do tamanho da amostra

Este experimento tem como objetivo demonstrar que o número de consultas utilizadas (quarenta) é aceitável, ou seja, é um valor que pode representar o comportamento de uma base de dados. Para isso, foram utilizadas as bases de dados $nomes_1$ e $nomes_2$, criadas para o experimento 4.2. Tais bases possuem 5.000 registros contendo nomes compostos de pessoas, onde 500 representam pessoas distintas e 4.500 representam duplicatas. A

base $nomes_1$ é utilizada para o processo de treinamento e a base $nomes_2$ para o processo de casamento. Para este teste, foram escolhidas quatro funções de similaridade, que são: (i) *Levenshtein*, (ii) *Carla*, (iii) *JaroWinkler* e (iv) *MatchingCoefficient*. As funções i, iii e iv são encontradas no pacote Java *SimMetrics* (CHAPMAN, 2009), e a função *Carla* foi desenvolvida por um aluno de doutorado da UFRGS (RITT et al., 2008; MERGEN; HEUSER, 2005).

Para analisar se a amostra com 40 consultas retorna um resultado aceitável, foram utilizadas outras sete, cada uma contendo um tamanho diferente (número diferente de registros consulta). Os tamanhos foram: 20,30,40,50,60,70,80 e 150. Foram sorteados 150 registros aleatoriamente de cada base de dados, e todas as 8 amostras saíram deste sorteio. Para a primeira, contendo 20 consultas, foram utilizados os 20 primeiros registros sorteados, para a que contém 30 consultas os 30 primeiros em assim por diante.

O processo de treinamento aqui efetuado é bem semelhante ao desenvolvido no experimento 4.2, a diferença é o número de objetos consultas utilizados. Neste, a base de dados utilizada também é a $nomes_1$, mas o treinamento é realizado oito vezes para cada função de similaridade, um para cada tamanho de amostra (sorteadas da base de dados $nomes_1$). No final deste processo, são formadas então oito TaMEs para cada função de similaridade (TaME-20, TaME-30, ... TaME-80 e TaME-150).

Com o processo de treinamento concluído, o próximo passo é realizar o casamento utilizando as TaMEs desenvolvidas no mesmo. O casamento também é semelhante ao do experimento 4.2, a diferença está no tamanho das amostras, aqui efetuado com oito tamanhos para cada função de similaridade. Estas, são sorteadas da base de dados $nomes_2$.

Como resultado deste processo, são geradas oito tabelas contendo os 11 pontos de precisão, para cada uma das funções utilizadas. Ou seja, é obtida uma tabela com os resultados de cada amostra. A tabela 4.17 apresenta, como exemplo, o resultado obtido no processo de casamento utilizando a função de similaridade *Levenshtein* e a amostra contendo 80 registros consulta. A primeira coluna contém os onze valores de limiar para os quais as precisões foram calculadas e a segunda (apa) contém os valores de precisão, os quais foram calculadas usando o escore ajustado. Trinta e duas tabelas como essa foram geradas no final do processo de casamento, pois são oito amostras e quatro funções de similaridade.

Limiar	**apa**
1	1
0,9	0,880812667
0,8	0,760688765
0,7	0,651809998
0,6	0,651809998
0,5	0,37310417
0,4	0,37310417
0,3	0,37310417
0,2	0,055901153
0,1	0,055901153
0	0,0023225

Tabela 4.17: Resultado obtido pelo processo de casamento utilizando escore ajustado com função de similaridade *Levenshtein* e amostra contendo 80 registros consulta.

Para que se possa ver claramente os resultados, foi calculado o desvio da diferença dos

quadrados entre os escores ajustados e os limiares, pois quanto mais próximo o escore ajustado for do limiar, mais preciso é seu resultado. Este cálculo é feito utilizando a equação 4.1.

Os resultados são plotados no gráfico da figura 4.25, onde cada linha representa o resultado de uma função de similaridade, o eixo x contém o tamanho da amostra e o eixo y apresenta o valor do desvio. A identificação das funções é dada pela legenda que se encontra na parte inferior da figura.

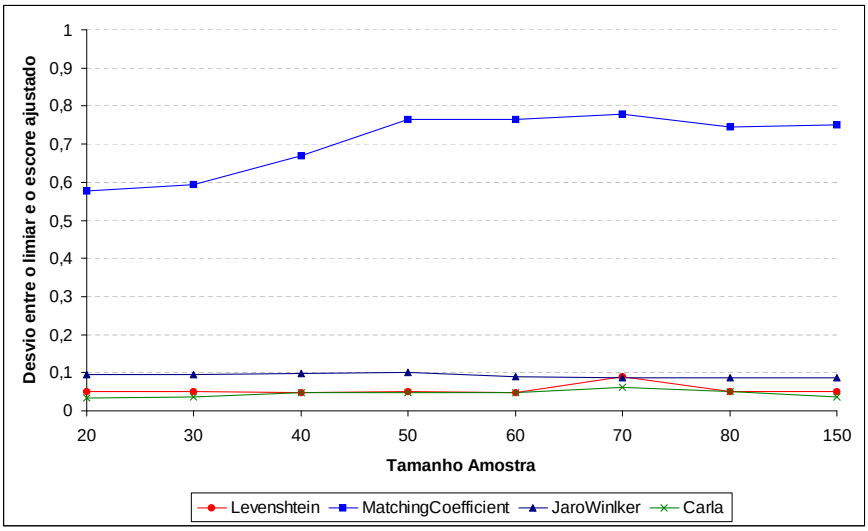

Figura 4.25: Gráfico contendo o resultados de desvio da diferença dos quadrados para as funções de similaridade *Levenshtein*, *Carla*, *JaroWinkler* e *MatchingCoefficient* variando o tamanho da amostra de consultas

Analisando este gráfico, é possível observar que os resultados tendem a permanecer próximos, mesmo aumentando o tamanho da amostra. Com exceção da função de similaridade *MatchingCoefficient*, onde aumentando o número de consultas o desvio tende a crescer, ou seja, com amostra maior os resultados ficaram mais distantes do ideal. As funções *Levenshtein*, *JaroWinkler* e *Carla* permanecem constantes com o aumento do tamanho das amostras, permanecendo sempre com resultado de desvio menor que $0,1$. Os valores no ponto 40 e no ponto 150 são bem próximos, isso quer dizer que, com a utilização de 40 consultas se consegue representar a base tão bem quanto com uma amostra maior. O resultado de *MatchingCoefficient* ficou ruim porque esta função não é adequada para o domínio de dados que está sendo utilizado. Como pode ser visto no experimento 4.2, esta função de similaridade apresenta resultados ruins para este domínio (tabela 4.8).

4.5 Granularidade da escala de precisão

Outra dúvida que pode existir, analisando o processo do escore ajustado, é se os 11 pontos de precisão, $[0.0, 0.1, \ldots, 0.9, 1.0]$, são realmente suficientes para se obter bons resultados, ou se com o aumento do número de pontos, os resultados ficam mais precisos, mais próximos do ideal.

Para esse fim, foi realizado um experimento com objetivo de testar se os resultados melhoram, utilizando um número maior de pontos de precisão no momento de efetuar os cálculos do escore ajustado. Este experimento compara os resultados da utilização de 11 pontos com a utilização de 21 pontos de precisão.

As bases de dados utilizadas neste experimento são as mesmas dos experimentos 4.2 e 4.4, que são a $nomes_1$ e $nomes_2$. A $nomes_1$ para o treinamento e $nomes_2$ para o processo de casamento. Foram escolhidas cinco funções de similaridade para a realização deste experimento. A escolha foi baseada nos resultados do experimento 4.2, pois foram selecionadas funções que não possuem valores de desvio alto. As funções utilizadas foram: (i) *Levenshtein*, (ii) *JaroWinkler*, (iii) *Q-grams Distance*, (iv) *NeedlemanWunch* e (v) *Carla*.

O processo de treinamento para os 11 pontos de precisão, aqui realizado, é o mesmo dos experimentos anteriores, só que utilizando uma amostra diferente de consultas. Foi realizado uma vez para cada uma das cinco funções de similaridade.

O treinamento utilizando 21 pontos é calculado sobre a mesma amostra de 40 consultas sorteadas no treinamento dos 11 pontos. O seu processo semelhante ao anterior, o que difere é que ao invés de calcular a precisão para 11 pontos de escore, é calculada para 21 pontos $[0.0, 0.05, 0.1 \ldots, 0.9, 0.95, 1.0]$.

Com o processo de treinamento finalizado, duas TaMEs foram criadas para cada função de similaridade, uma para os 11 pontos de precisão e a outra para os 21 pontos. Com isso, o próximo passo é realizar o processo de casamento utilizando estas tabelas.

O casamento também é efetuado duas vezes para cada função, uma com os 11 pontos de precisão, e a segunda com os 21 pontos, utilizando suas respectivas TaMEs.

O processo com a utilização dos 11 pontos é realizado da mesma forma explicada no experimento 4.2. Uma amostra de 40 consultas é sorteada da base de dados $nomes_2$, e o casamento é realizado.

O casamento utilizando 21 pontos de precisão é similar ao realizado anteriormente, a diferença está no cálculo da precisão interpolada, que ao invés de computar a precisão para 11 pontos de escore, é calculada para 21 pontos.

No final do processo de casamento, são formadas duas tabelas de resultado para cada função de similaridade, uma contendo os resultados com a utilização de 11 pontos de precisão e a outra com o resultados dos 21 pontos. Tais tabelas possuem o limiar e o respectivo valor de precisão média obtido pelo uso do escore ajustado. Lembrando que o valor de precisão média ideal é o valor do seu limiar. Para quantificar a diferença entre o limiar e a precisão média obtida em cada tabela (11 pontos e 21 pontos), foram calculados os desvios médios da diferença dos quadrados entre eles. Tal cálculo é realizado com a seguinte equação:

$$d_a = \frac{\sum_{i=1}^{n} [apa_i - t_i]^2}{n} \tag{4.10}$$

onde t_i é um dos pontos de limiar (a precisão esperada pelo usuário), apa_i é a precisão média resultante para cada limiar usando o escore ajustado e n é o número de pontos de precisão, no caso 11 ou 21.

O resultado deste experimento pode ser visto na tabela 4.18, a qual possui os valores de desvio obtidos tanto com a utilização de 11 como de 21 pontos de precisão. Na primeira coluna desta tabela, são apresentadas as funções de similaridade utilizadas, na segunda, os valores de desvio utilizando 21 pontos de precisão, na terceira, os desvios utilizando 11 pontos e, na quarta e última, são apresentadas as diferenças entre o desvio dos 21 pontos e o dos 11 pontos, ou seja, valores de desvio dos 21 pontos menos os dos 11 pontos. Essa diferença serve para observar se a utilização dos 21 pontos melhora ou não a qualidade dos resultados. Se o valor da diferença for negativo, significa que os resultados obtidos com 21 pontos são mais próximo do ideal, se for zero, significa que os resultados são

iguais e, se for positivo, significa que os resultados ficam mais longe do ideal do que com 11 pontos.

Função de Similaridade	**Desvio 21 pontos**	**Desvio 11 pontos**	**Diferença desvio 21 - 11 pontos**
Levenstein	0,002547	0,005357	-0,002810
JaroWinkler	0,018691	0,008938	0,009753
QGrams Distance	0,009290	0,004873	0,004417
Carla	0,004664	0,002540	0,002124
NeedlemanWunch	0,034944	0,019207	0,015737

Tabela 4.18: Resultado de desvio obtido utilizando 21 e 11 pontos de precisão para cada função de similaridade

Analisando esta tabela é possível observar que a utilização de 21 pontos de precisão melhorou os resultados apenas na função de similaridade *Levenshtein*, e a melhora foi pouca, tendo uma diferença de $-0,002810$ entre os desvios. Para as outras quatro funções de similaridade (*JaroWinkler*, *Q-grams Distance*, *NeedlemanWunch* e *Carla*) os resultados ficaram piores do que utilizando os 11 pontos, no entanto, a diferença também é bem pequena. Com essa análise, pode-se concluir que não foi encontrada diferença significativa entre a utilização de 11 e de 21 pontos de precisão.

5 CONCLUSÕES E TRABALHOS FUTUROS

O presente trabalho apresenta um estudo experimental realizado sobre uma técnica de padronização de escores de similaridade, aqui chamada de *MeaningScore*, a qual foi desenvolvida em um trabalho anterior por uma aluna de doutorado do grupo de banco de dados da UFRGS (DORNELES et al., 2007) e está apresentada no capítulo 3 deste trabalho.

Dorneles et al. (2007) propuseram e desenvolveram tal abordagem, no entanto, ficaram em aberto alguns pontos em sua avaliação experimental. Um experimento foi realizado com intuito de avaliar sua abordagem, porém ocorreram erros como arredondamento nos cálculos do processo de padronização de escores. Então, alguns pontos ficaram em aberto com a sua conclusão, os quais são citados a seguir: (i) avaliar experimentalmente a utilização do escore padronizado, chamado de escore ajustado, (ii) analisar o comportamento do escore ajustado utilizando diferentes funções de similaridade com a mesma base de dados, (iii) utilizar o escore ajustado no processo de casamento de registros, onde se faz necessária a dos escores de diferentes funções de similaridade e, por último, (iv) durante todo o processo do cálculo do escore ajustado, a precisão é calculada em 11 pontos de escore ($[0, 0.1, \ldots, 0.9, 1.0]$), então é necessário testar se a utilização dos 11 pontos de precisão é o suficiente, ou se com a utilização de mais pontos os resultados melhoram.

A técnica *MeaningScore* foi estudada e avaliada experimentalmente neste trabalho, cobrindo os pontos de avaliação que estavam em aberto. Os experimentos realizados são apresentados no capítulo 4 e com eles é possível perceber que com a utilização desta abordagem os resultados ficam mais próximos do ideal, além disso, percebe-se que o treinamento, utilizando determinada função de similaridade, pode ser realizado em uma base de domínio x e ser utilizado em outras bases do mesmo domínio, facilitando assim a utilização desta abordagem. Com os resultados obtidos, também é possível observar que o escore ajustado depende de funções que sejam adequadas para o domínio de dados, então a escolha da função de similaridade é essencial para que se obtenha bons resultados. Porém, analisando os experimentos da subseção 4.3.3 é possível perceber que mesmo não utilizando funções de similaridade ideais para o domínio de dados, os resultados com o escore ajustado ainda são melhores do que utilizando a função de similaridade original.

Um ponto importante a observar é que a função de padronização de escore pode ser utilizada em conjunto com abordagens existentes de casamento de registro, visando melhorar seus resultados, como apresentado nos experimentos da seção 4.3.

Como produção científica foram publicados os seguintes artigos:

- (DORNELES et al., 2009) A strategy for allowing meaningful and comparable scores in approximate matching. **Information Systems**, Oxford, UK, UK, v.34, n.8, p.740-756, 2009. Este artigo apresenta a abordagem *MeaningScore* e os experi-

mentos das seções 4.1 e 4.3.

- (DORNELES et al., 2007) A Strategy for Allowing Meaningful and Comparable Scores in Approximate Matching. In: ACM CONFERENCE ON INFORMATION AND KNOWLEDGE MANAGEMENT, CIKM, 16., 2007. Proceedings... New York: ACM, 2007. p.303-312. Este artigo apresenta a abordagem *MeaningScore* e os experimentos da seção 4.1.

Com a conclusão deste trabalho, alguns experimentos interessantes de realizar foram identificados.

Um deles seria executar experimentos utilizando uma base de dados real com maior números de registros, para que se possa verificar se o comportamento do escore ajustado permanece o mesmo com o aumento do escopo de testes. Porém, um grande problema para realizar este trabalho é obter uma base de dados avaliada com milhares de registros, pois para realizar testes é necessário que se tenha conhecimento dos objetos duplicados. Se a base não for avaliada, um especialista deve estudá-la e marcar manualmente quais registros são duplicatas.

Seria interessante realizar também um experimento utilizando a abordagem de Carvalho et al. (2006) que utiliza programação genética para deduplicar objetos. Para que se possa observar se o escore ajustado consegue melhorar os resultados de abordagens complexas como esta.

Um terceiro experimento que poderia ser realizado é utilizar o trabalho de Santos et al. (2008) para efetuar o processo de treinamento de forma automática, pois a sua proposta é realizar o processo de estimativa da precisão e da revocação automaticamente.

REFERÊNCIAS

BAEZA-YATES, R.; RIBEIRO-NETO, B. **Modern Information Retrieval**. [S.l.]: ACM Press / Addison-Wesley, 1999.

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

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

BORGES, E. N.; GALANTE, R. M. **MD-PROM**: um mecanismo para deduplicação de objetos provenientes de bibliotecas digitais. Porto Alegre, 2008.

BOSER, B. E.; GUYON, I. M.; VAPNIK, V. N. A training algorithm for optimal margin classifiers. In: ANNUAL WORKSHOP ON COMPUTATIONAL LEARNING THEORY, COLT, 5., 1992. **Proceedings...** New York: ACM, 1992. p.144–152.

BRUNO, N.; CHAUDHURI, S.; GRAVANO, L. Top-k selection queries over relational databases: mapping strategies and performance evaluation. **ACM Trans. Database Syst.**, New York, NY, USA, v.27, n.2, p.153–187, 2002.

BUENO, R.; TRAINA, A. J. M.; TRAINA, C. Accelerating approximate similarity queries using genetic algorithms. In: ACM SYMPOSIUM ON APPLIED COMPUTING, SAC, 2005, New York, NY, USA. **Proceedings...** ACM, 2005. p.617–622.

CARVALHO, J. C. P.; SILVA, A. S. da. Finding similar identities among objects from multiple web sources. In: ACM INTERNATIONAL WORKSHOP ON WEB INFORMATION AND DATA MANAGEMENT, WIDM, 5., 2003. **Proceedings...** New York: ACM, 2003. p.90–93.

CARVALHO, M. G.; GONÇALVES, M. A.; LAENDER, A. H. F.; SILVA, A. S. Learning to deduplicate. In: JOINT CONFERENCE ON DIGITAL LIBRARIES, JCDL, 6., 2006. **Proceedings...** New York: ACM, 2006. p.41–50.

CHAPMAN, S. **SimMetrics**: a java & c#. net library of similarity metrics. Disponível em: `http://sourceforge.net/projects/simmetrics/`. Acesso em: 27 jan. 2009.

CHAUDHURI, S.; CHEN, B.-C.; GANTI, V.; KAUSHIK, R. Example-driven design of efficient record matching queries. In: VERY LARGE DATA BASES, VLDB, 33., 2007. **Proceedings...** VLDB Endowment, 2007. p.327–338.

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, SIGMOD, 2003. **Proceedings...** New York: ACM, 2003. p.313–324.

CHAUDHURI, S.; GRAVANO, L. Evaluating Top-k Selection Queries. In: INTERNATIONAL CONFERENCE ON VERY LARGE DATA BASES, VLDB, 25., 1999. **Proceedings...** San Francisco: CA: Morgan Kaufmann Publishers, 1999. p.397–410.

CHEN, F.; FARAHAT, A.; BRANTS, T. Multiple Similarity Measures and Source-Pair Information in Story Link Detection. In: HLT-NAACL 2004, HUMAN LANGUAGE TECHNOLOGY CONFERENCE OF THE NORTH AMERICAN CHAPTER OF THE ASSOCIATION FOR COMPUTATIONAL LINGUISTICS, MAY 2-7, BOSTON, MASSACHUSETTS, 2004. **Anais...** [S.l.: s.n.], 2004. p.313–320.

CHOWDHURY, G. G. **Introduction to Modern Information Retrieval**. [S.l.]: Neal-Schuman Publishers, 2003.

CHRISTEN, P.; CHURCHES, T.; HEGLAND, M. Febrl - A Parallel Open Source Data Linkage System. In: PACIFIC ASIA KNOWLEDGE DISCOVERY AND DATA MINING, PAKDD 2004 (LNAI 3056), 2004. **Proceedings...** Springer, 2004. p.638–647.

COHEN, W. W.; RAVIKUMAR, P.; FIENBERG, S. E. A Comparison of String Distance Metrics for Name-Matching Tasks. In: WORKSHOP ON INFORMATION INTEGRATION, IJCAI, 2003. **Proceedings...** [S.l.: s.n.], 2003. p.73–78.

COHEN, W. W.; RAVIKUMAR, P.; FIENBERG, S. E. **SecondString**: open source java-based package of approximate string-matching. Disponível em: `http://secondstring.sourceforge.net/`. Acesso em: 27 jan. 2009.

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

CULOTTA, A.; MCCALLUM, A. Joint deduplication of multiple record types in relational data. In: ACM INTERNATIONAL CONFERENCE ON INFORMATION AND KNOWLEDGE MANAGEMENT, CIKM, 14., 2005. **Proceedings...** New York: ACM, 2005. p.257–258.

DALVI, N. N.; SUCIU, D. Efficient Query Evaluation on Probabilistic Databases. In: VERY LARGE DATA BASES, VLDB, 30., 2004. **Proceedings...** [S.l.: s.n.], 2004. p.864–875.

DEY, D.; SARKAR, S.; DE, P. Entity Matching in Heterogeneous Databases: a distance based decision model. In: ANNUAL HAWAII INTERNATIONAL CONFERENCE ON SYSTEM SCIENCES-VOLUME 7, HICSS, 31., 1998, Washington, DC, USA. **Proceedings...** IEEE Computer Society, 1998. p.305.

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. F.; HEUSER, C. A.; LIMA, A. E. N.; SILVA, A. S. da; MOURA, E. S. de. Measuring similarity between collection of values. In: ANNUAL ACM INTERNATIONAL WORKSHOP ON WEB INFORMATION AND DATA MANAGEMENT, WIDM, 6., 2004. **Proceedings...** New York: ACM, 2004. p.56–63.

DORNELES, C. F.; HEUSER, C. A.; ORENGO, V. M.; SILVA, A. S. da; MOURA, E. S. de. A strategy for allowing meaningful and comparable scores in approximate matching. In: ACM CONFERENCE ON INFORMATION AND KNOWLEDGE MANAGEMENT, CIKM, 16., 2007. **Proceedings...** New York: ACM, 2007. p.303–312.

DORNELES, C. F.; NUNES, M. F.; HEUSER, C. A.; MOREIRA, V. P.; SILVA, A. S. da; MOURA, E. S. de. A strategy for allowing meaningful and comparable scores in approximate matching. **Information Systems**, Oxford, UK, UK, v.34, n.8, p.740–756, 2009.

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.

FELLEGI, I. P.; SUNTER, A. B. A Theory for Record Linkage. **Journal of the American Statistical Association**, [S.l.], v.64, n.328, p.1183–1210, December 1969.

FERGUSON, A.; BRIDGE, D. Generalised Prioritisation: a new way of combining similarity metrics. In: INTERNATIONAL CONFERENCE ON ARTIFICIAL INTELLIGENCE AND COGNITIVE SCIENCE, AICS., 1999. **Proceedings...** [S.l.: s.n.], 1999. p.137–142.

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

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...** [S.l.: s.n.], 2004. p.636–647.

GUHA, S.; KOUDAS, N.; SRIVASTAVA, D.; YU, X. Reasoning About Approximate Match Query Results. In: INTERNATIONAL CONFERENCE ON DATA ENGINEERING, ICDE., 22., 2006, Atlanta, GA, USA. **Proceedings...** [S.l.: s.n.], 2006. p.8.

HEUSER, C. A.; KRIESER, F. N. A.; ORENGO, V. M. SimEval: a tool for evaluating the quality of similarity functions. In: TUTORIALS, POSTERS, PANELS AND INDUSTRIAL CONTRIBUTIONS AT THE INTERNATIONAL CONFERENCE ON CONCEPTUAL MODELING, ER., 26., 2007, Darlinghurst, Australia, Australia. **Proceedings...** Australian Computer Society: Inc., 2007. p.71–76.

JARO, M. A. Advances in record linking methodology as applied to the 1985 census of Tampa Florida. **Journal of the American Statistical Society**, [S.l.], v.64, p.1183–1210, 1989.

KOUDAS, N.; MARATHE, A.; SRIVASTAVA, D. Flexible String Matching Against Large Databases in Practice. In: VERY LARGE DATA BASES, VLDB, 30., 2004, Toronto, Canada. **Proceedings...** [S.l.: s.n.], 2004. p.1078–1086.

LAENDER, A. H. F.; GONÇALVES, M. A.; ROBERTO, P. A. BDBComp: building a digital library for the brazilian computer science community. In: ACM/IEEE-CS JOINT CONFERENCE ON DIGITAL LIBRARIES, JCDL, 4., 2004. **Proceedings...** New York: ACM, 2004. p.23–24.

LEE, L. On the Effectiveness of the Skew Divergence of Statistical Language Analysis. **Artificial Intelligence and Statistics**, [S.l.], p.65–72, 2001.

LEITÃO, L.; CALADO, P.; WEIS, M. Structure-based inference of xml similarity for fuzzy duplicate detection. In: ACM CONFERENCE ON CONFERENCE ON INFORMATION AND KNOWLEDGE MANAGEMENT, CIKM, 16., 2007, New York, NY, USA. **Proceedings...** ACM, 2007. p.293–302.

LEVENSHTEIN, V. I. Binary codes capable of correcting deletions, insertions and reversals. **Soviet Physics Doklady.**, [S.l.], v.10, n.8, p.707–710, February 1966.

MANNING, C. D.; RAGHAVAN, P.; SCHÜTZE, H. **Introduction to Information Retrieval**. [S.l.]: Cambridge University Press, 2008.

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

MERGEN, S. S.; HEUSER, C. A. Carla: uma técnica para comparação de cadeias de caracteres. In: I ESCOLA REGIONAL DE BANCOS DE DADOS, ERBD, 2005. **Anais...** SBC, 2005. p.55–70.

MOTRO, A. VAGUE: a user interface to relational databases that permits vague queries. **ACM Transactions on Office Information Systems**, [S.l.], v.6, n.3, p.187–214, July 1988.

QUINLAN, J. R. Induction of Decision Trees. **Mach. Learn.**, Hingham, MA, USA, v.1, n.1, p.81–106, 1986.

RISTAD, E. S.; YIANILOS, P. N. Learning String Edit Distance. **IEEE Transactions on Pattern Recognition and Machine Intelligence**, [S.l.], v.20, n.5, p.522–532, May 1998.

RITT, M.; COSTA, A. M.; MERGEN, S. S.; ORENGO, V. M. An integer linear programming approach for approximate string comparison. **European Journal of Operational Research**, [S.l.], 2008.

SÍLABO (Ed.). **Estatística Descritiva - Manual de Auto-Aprendizagem**. [S.l.: s.n.], 2007.

SANTOS, J. B. dos; HEUSER, C. A.; ORENGO, V. M.; WIVES, L. K. Automatic threshold estimation for data matching applications. In: BRAZILIAN SYMPOSIUM ON DATABASES,SBBD, 23., 2008. **Proceedings...** Campinas: SBC, 2008. p.106–119.

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

SILVA, R. da; STASIU, R.; ORENGO, V. M.; HEUSER, C. A. Measuring quality of similarity functions in approximate data matching. **Journal of Informetrics**, [S.l.], v.1, n.1, p.35–46, 2007.

STASIU, R. K.; HEUSER, C. A.; SILVA, R. da. Estimating Recall and Precision for Vague Queries in Databases. In: CONFERENCE ON ADVANCED INFORMATION SYSTEMS ENGINEERING, CAISE., 17., 2005. **Proceedings...** Berlin: Springer, 2005. p.187–200. (Lecture Notes in Computer Science, v.3520).

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

WINKLER, W. E. String Comparator Metrics and Enhanced Decision Rules in the Fellegi-Sunter Model of Record Linkage. In: SECTION ON SURVEY RESEARCH, 1990. **Proceedings...** [S.l.: s.n.], 1990. p.354–359.

WINKLER, W. E. **The State of Record Linkage and Current Research Problems**. [S.l.]: U.S. Bureau of the Census, Washington, D.C., 1999. (Statistical Research Report Series RR99/04).