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

GUILHERME DAL BIANCO

Redução do Esforço do Usuário na Configuração da Deduplicação de Grandes Bases de Dados

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

Profa. Dra. Renata Galante
Orientador

Prof. Dr. Carlos A. Heuser
Co-orientador

Porto Alegre, janeiro de 2014

CIP – CATALOGAÇÃO NA PUBLICAÇÃO

Guilherme Dal Bianco,

Redução do Esforço do Usuário na Configuração da Deduplicação de Grandes Bases de Dados / Guilherme Dal Bianco. – Porto Alegre: PPGC da UFRGS, 2014.

112 f.: il.

Tese (doutorado) – Universidade Federal do Rio Grande do Sul. Programa de Pós-Graduação em Computação, Porto Alegre, BR–RS, 2014. Orientador: Renata Galante; Co-orientador: Carlos A. Heuser.

1. Integração de dados. 2. Deduplicação. 3. Deduplicação por assinatura. I. Renata Galante, . II. Carlos A. Heuser, . III. Título.

“Muitos são os obstinados que se empenham no caminho que escolheram,
poucos os que se empenham no objetivo.”
— FRIEDRICH NIETZSCHE

AGRADECIMENTOS

Meus mais sinceros agradecimentos a todos que me apoiaram durante esta longa caminhada. Em especial, gostaria de agradecer as seguintes pessoas:

À minha família, pelo apoio incondicional durante todas as batalhas conquistadas desde minha formação acadêmica até a conclusão do doutorado.

À minha mãe que sempre me incentivou fazendo suas orações para iluminar o meu caminho aos mais diversos santos. Mesmo nos seus momentos mais difíceis, sempre manteve a fé e a esperança. Muito obrigado mãe!

Ao meu pai, pelo incentivo aos estudos muito importante para minha formação e do meu amadurecimento como pessoa.

Às minhas irmãs, pelo apoio nos momentos mais difíceis. Agradeço imensamente pelo fortalecimento do núcleo familiar que vai ser levado durante todas as nossas vidas.

À minha namorada, Bruna Almeida, pelo apoio e a paciência nos momentos mais difíceis e desoladores, principalmente durante os áridos meses finais do doutorado.

À minha orientadora, Renata Galante, por acreditar, apoiar e incentivar durante todos esses anos. Ao meu co-orientador, Carlos A. Heuser, por contribuir com toda sua experiência, conhecimento e visão.

Ao Professor Marcos André Gonçalves pelas contribuições fundamentais no desenvolvimento desta tese. Agradeço também por possibilitar a experiência de trabalhar junto ao grupo de Banco de Dados da UFMG.

Aos meus colegas de sala, pelas amizades criadas que serão certamente carregadas para toda a vida. Especialmente, gostaria de agradecer aos meus colegas Bruno Rezende Laranjeira, Edimar Manica, Gustavo Zanini Kantorski, Solange Pertile e Mirian Colpo que colaboram na revisão do texto da tese. Muito Obrigado!

Ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) e ao Programa de formação de Recursos Humanos da Petrobras (PRH-PB17), pelo imprescindível apoio financeiro para o fomento de minha pesquisa. Mais ainda, à Professora Taisy Weber pela atenção e dedicação junto à coordenação do projeto de pesquisa que fomentou esta tese.

Aos professores membros da banca da banca: Profa. Dra. Carina Dorneles, Prof. Dr. Ricardo Torres e o Prof. Dr. Leandro Krug Wives, por terem aceitado participar desta banca e contribuir para o aprimoramento do texto final.

SUMÁRIO

LISTA DE FIGURAS

LISTA DE TABELAS

RESUMO

A deduplicação consiste na tarefa de identificar quais objetos (registros, documentos, textos, etc.) são potencialmente os mesmos em uma base de dados (ou em um conjunto de bases de dados). A identificação de dados duplicados depende da intervenção do usuário, principalmente para a criação de um conjunto contendo pares duplicados e não duplicados. Tais informações são usadas para ajudar na identificação de outros possíveis pares duplicados presentes na base de dados.

Em geral, quando a deduplicação é estendida para grandes conjuntos de dados, a eficiência e a qualidade das duplicatas dependem diretamente do "ajuste" de um usuário especialista. Nesse cenário, a configuração das principais etapas da deduplicação (etapas de blocagem e classificação) demandam que o usuário seja responsável pela tarefa pouco intuitiva de definir valores de limiares e, em alguns casos, fornecer pares manualmente rotulados. Desse modo, o processo de calibração exige que o usuário detenha um conhecimento prévio sobre as características específicas da base de dados e os detalhes do funcionamento do método de deduplicação.

O objetivo principal desta tese é tratar do problema da configuração da deduplicação de grandes bases de dados, de modo a reduzir o esforço do usuário. O usuário deve ser somente requisitado para rotular um conjunto reduzido de pares automaticamente selecionados. Para isso, é proposta uma metodologia, chamada FS-Dedup, que incorpora algoritmos do estado da arte da deduplicação para permitir o processamento de grandes volumes de dados e adiciona um conjunto de estratégias com intuito de possibilitar a definição dos parâmetros do deduplicador, removendo os detalhes de configuração da responsabilidade do usuário. A metodologia pode ser vista como uma camada capaz de identificar as informações requisitadas pelo deduplicador (principalmente valores de limiares) a partir de um conjunto de pares rotulados pelo usuário.

A tese propõe também uma abordagem que trata do problema da seleção dos pares informativos para a criação de um conjunto de treinamento reduzido. O desafio maior é selecionar um conjunto reduzido de pares suficientemente informativo para possibilitar a configuração da deduplicação com uma alta eficácia. Para isso, são incorporadas estratégias para reduzir o volume de pares candidatos a um algoritmo de aprendizagem ativa. Tal abordagem é integrada à metodologia FS-Dedup para possibilitar a remoção da intervenção especialista nas principais etapas da deduplicação.

Por fim, um conjunto exaustivo de experimentos é executado com objetivo de validar as ideias propostas. Especificamente, são demonstrados os promissores resultados alcançados nos experimentos em bases de dados reais e sintéticas, com intuito de reduzir o número de pares manualmente rotulados, sem causar perdas na qualidade da deduplicação.

Palavras-chave: Integração de dados, Deduplicação, Deduplicação por assinatura.

Reducing the User Effort to Tune Large Scale Deduplication

ABSTRACT

Deduplication is the task of identifying which objects (e.g., records, texts, documents, etc.) are potentially the same in a given dataset (or datasets). It usually requires user intervention in several stages of the process, mainly to ensure that pairs representing matchings and non-matchings can be determined. This information can be used to help detect other potential duplicate records.

When deduplication is applied to very large datasets, the matching quality depends on expert users. The expert users are requested to define threshold values and produce a training set. This intervention requires user knowledge of the noise level of the data and a particular approach to deduplication so that it can be applied to configure the most important stages of the process (e.g. blocking and classification).

The main aim of this thesis is to provide solutions to help in tuning the deduplication process in large datasets with a reduced effort from the user, who is only required to label an automatically selected subset of pairs. To achieve this, we propose a methodology, called FS-Dedup, which incorporates state-of-the-art algorithms in its deduplication core to address high performance issues. Following this, a set of strategies is proposed to assist in setting its parameters, and removing most of the detailed configuration concerns from the user. The methodology proposed can be regarded as a layer that is able to identify the specific information requested in the deduplication approach (mainly, threshold values) through pairs that are manually labeled by the user.

Moreover, this thesis proposed an approach which would enable to select an informative set of pairs to produce a reduced training set. The main challenge here is how to select a "representative" set of pairs to configure the deduplication with high matching quality. In this context, the proposed approach incorporates an active learning method with strategies that allow the deduplication to be carried out on large datasets. This approach is integrated with the FS-Dedup methodology to avoid the need for a definition of threshold values in the most important deduplication stages.

Finally, exhaustive experiments using both synthetic and real datasets have been conducted to validate the ideas outlined in this thesis. In particular, we demonstrate the ability of our approach to reduce the user effort without degrading the matching quality.

Keywords: Data integration, Deduplication, Signature-Based deduplication.

1 INTRODUÇÃO

Nas últimas décadas, um massivo volume de informações tem sido produzido de variadas fontes de dados, tais como: dispositivos móveis, redes sociais, transmissões em tempo real (ou *streamming*), entre outros. Tal volume de informações oferece uma vasta oportunidade para extrair novos conhecimentos e, ao mesmo tempo, impulsiona a demanda por novas soluções para as tarefas de armazenamento, integração e recuperação de informações. Muitos desses serviços estão incorporados ao nosso cotidiano, como, por exemplo: Web sites de comparação de preços (Buscapé, Bom de Faro); bibliotecas digitais (DBLP, CiteSeer); sistemas de recomendação (Amazon.com); locadoras virtuais (Netflix); armazenamento virtual de dados (*Amazon Web Service*); entre outros. De um modo geral, os serviços dependem de bases de dados concisas e sem erros, ou seja, com uma boa qualidade.

Erros ou anomalias nas bases de dados podem ser facilmente identificados em situações reais. As anomalias podem estar presentes de diferentes formas como, por exemplo, a partir de erros léxicos e de sintaxe, erros de formato, irregularidades, violação de integridade, informações duplicadas, ausência e inversão de campos, entre outras (OLIVEIRA; RODRIGUES; HENRIQUES, 2005). Os problemas da baixa qualidade dos dados podem afetar diretamente o funcionamento de uma organização nos seus mais variados níveis, desde o aumento no tempo de processamento para a execução de uma determinada consulta até a dificuldade na tomada de decisões estratégicas, principalmente, devido à análise enviesada ou imprecisa das informações presentes nas bases de dados.

A biblioteca digital CiteSeer (LAWRENCE; GILES; BOLLACKER, 1999), que promove um serviço autônomo de indexação de publicações científicas, sofre diretamente com o problema da qualidade dos dados. Na biblioteca CiteSeer, são utilizados coletores de documentos (*Web crawlers*) para promover uma varredura em busca de novas informações dispersas na Web. No entanto, a extração automática de campos, nos documentos recuperados, pode resultar em informações imprecisas que devem ser corretamente identificadas antes de serem armazenadas no repositório de dados. Por exemplo, a Figura 1-(A) apresenta o *ranking* retornado pela consulta à publicação "*A New Consistency Metric for Scalable Monitoring*". Note-se que os dois primeiros registros apresentam informações aparentemente idênticas, ilustrando uma anomalia na base de dados. Mais detalhes dos registros recuperados pela consulta podem ser visualizados nas Figuras 1-(B) e 1-(C). Em um dos registros retornados (Figura 1-(C)), o nome dos autores é inserido erroneamente junto ao título, resultando em uma duplicata. Caso as anomalias nos dados não sejam tratadas, a biblioteca digital poderá comprometer os serviços de busca e de recomendação.

Uma melhora substancial na qualidade dos dados pode ser realizada a partir da identificação de objetos similares. A similaridade de objetos é estudada em uma variada gama de áreas, desde o contexto da detecção de registros, documentos ou Web sites similares,

até nas áreas de plágio, identificação de *spams* e reconhecimento de padrões. Esta tese tem como foco o problema de identificação de registros em bases de dados que correspondem a um mesmo objeto no mundo real, chamada de deduplicação. A deduplicação é conhecida por diversos nomes na literatura: *casamento de registros* (GILL, 2001; CHRISTEN; GOISER, 2007); *identificação de duplicatas* (HERNáNDEZ; STOLFO, 1995a); *identificação de incertezas* (PASULA, 2003); *casamento de objetos* (KIM; LEE, 2007); *casamento aproximado* (GRAVANO et al., 2001; ELMAGARMID; IPEIROTIS; VERYKIOS, 2007); entre outros.

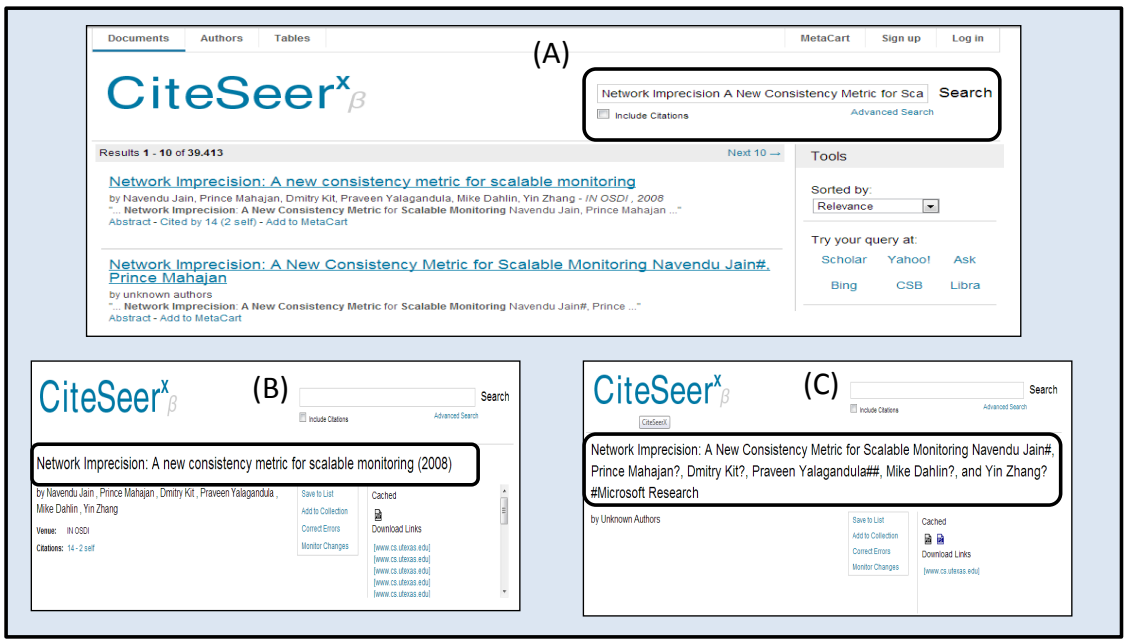


Figura 1.1: Exemplo de registros duplicados retornados por uma consulta à biblioteca digital CiteSeer (A). Um detalhamento dos registros duplicados é apresentado nas figuras (B) e (C).

Uma primeira proposta para a deduplicação é apresentada por NEWCOMBE et al. (1959). No entanto, devido à ausência de estudos estatísticos, o método proposto por NEWCOMBE et al. (1959) é considerado empírico (ELMAGARMID; IPEIROTIS; VERYKIOS, 2007). A primeira fundamentação teórica foi realizada por FELLEGI; SUNTER (1969). Nela os autores formularam uma série de teorias para a formalização do problema da deduplicação. Tal embasamento teórico é utilizado até os dias atuais em inúmeros trabalhos científicos (CHRISTEN; CHURCHES; HEGLAND, 2004; KIM; LEE, 2007; DAL BIANCO et al., 2013). Apesar de os primeiros trabalhos para a deduplicação de dados terem sido propostos em meados do século passado, a deduplicação tem recebido uma atenção especial da comunidade científica devido à necessidade de soluções eficientes, com uma reduzida intervenção do usuário, e capazes de processar grandes volumes de dados (ELMAGARMID; IPEIROTIS; VERYKIOS, 2007).

As abordagens referentes à deduplicação operam em três fases distintas: (*i*) *pré-processamento*, na qual são removidos caracteres indesejáveis, realizando um processo de padronização nos valores de atributos (por exemplo, padronizar o atributo data no formato internacional); (*ii*) *blocagem*, que visa reduzir substancialmente o número de pares a serem comparados a partir do agrupamento de registros com características em comum; e (*iii*) *classificação*, na qual, tipicamente, uma função de similaridade é aplicada para

Tabela 1.1: Exemplo de registros da base de dados IMDBe NetFlix.

Fonte	Título	Ano	Diretor	Tempo
IMDB	Brazil - O Filme	1985	Terry Gilliam	110 min
NetFlix	Brazil	1985	Terry Gilliam	120 min

quantizar o grau de semelhança entre pares de registros (por exemplo, Jaccard, Levenshtein, Jaro (ELMAGARMID; IPEIROTIS; VERYKIOS, 2007)) e limiares ou modelos de classificação, utilizados para a predição dos pares.

Tipicamente, a deduplicação depende de um usuário para a "configuração" de suas principais etapas internas (por exemplo, a blocagem e a classificação). A intervenção do usuário é fundamental para identificar, em cada contexto, quais pares se referem a uma duplicata. A Tabela 1.1 ilustra dois exemplos de registros que armazenam informações referentes ao domínio de mídias. Tais registros possuem os atributos "ano" e "diretor" com valores idênticos, já o atributo "título" apresenta uma variação no conteúdo ("Brazil - O Filme" vs. "Brazil"). No contexto de um catálogo virtual, no qual o objetivo é disponibilizar informações genéricas sobre mídias (por exemplo, "Internet Movie DataBase-IMDB"[1]), a correspondência entre os valores dos atributos "ano de lançamento" e "diretor" define tais registros como uma duplicata. Por outro lado, caso tais registros pertençam a uma locadora virtual de mídias (por exemplo, NetFlix[2]), provavelmente outras informações devam ser analisadas para identificar se os registros representam uma duplicata ou não, como, por exemplo, a qualidade de armazenamento, o tempo de duração, a mídia utilizada, entre outras informações que, em um catálogo virtual, não são relevantes. De fato, para que o método de deduplicação seja capaz de identificar corretamente o que representa uma duplicata, é necessária a intervenção do usuário para direcionar ou contextualizar a configuração do processo.

Além disso, a deduplicação depende da configuração da etapa de blocagem para a geração dos pares candidatos. O elevado número de pares que devem ser analisados para identificar as duplicatas é um dos fatores mais impactantes para a complexidade da deduplicação (CHRISTEN, 2012). Em grandes bases de dados (compostas por milhões de registros), uma comparação quadrática torna-se inviável devido ao consumo acentuado de recursos computacionais. Nesse contexto, a blocagem é utilizada com o objetivo de aliviar o acentuado custo computacional da deduplicação, produzindo pares candidatos (possíveis duplicatas) somente entre registros pertencentes a um mesmo bloco. No entanto, a maximização qualidade dos pares gerados depende de um ajuste de acordo com o nível de ruído da base de dados. Tal configuração, por exemplo, pode ser representada a partir da escolha de evidências (definição de quais atributos serão utilizados para a geração dos blocos) e limiares de similaridade que determinam os ajustes para configurar o tamanho dos blocos.

De um modo geral, a intervenção do usuário para a configuração da deduplicação pode ser realizada de duas formas: direta ou indireta. Na intervenção direta (também conhecida como não supervisionada), um "usuário especialista" define um conjunto de valores de limiares utilizados para configurar o deduplicador (CHAUDHURI; GANTI; KAUSHIK, 2006; SARAWAGI; KIRPAL, 2004; BAYARDO; MA; SRIKANT, 2007; XIAO et al., 2011; WANG; LI; FE, 2011). Os pares são mapeados para valores de similaridade (a partir de funções de similaridade) e comparados aos limiares determinados pelo usuário. Um

[1]IMDB: www.imdb.com.br.

[2]NetFlix: www.netflix.com.br

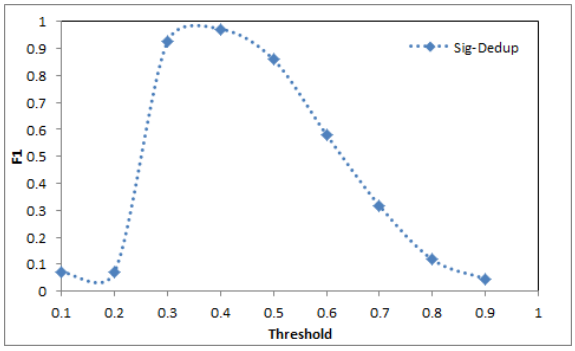


Figura 1.2: Exemplo do processo de identificação do valor de limiar para configuração da deduplicação de uma base de dados sintética.

limiar ideal se caracteriza pela separação precisa dos pares duplicados e não duplicados. A definição de valores de limiares ideais é uma tarefa complexa, restrita a "usuários especialistas", devido à dependência de múltiplos fatores, como o nível de ruído da base de dados, o funcionamento interno da abordagem de deduplicação e do contexto do usuário. Em bases de dados reais, cujo gabarito é raramente conhecido, o processo de calibração exige que o usuário realize inúmeras execuções, variando o valor do limiar. Cada um dos limiares é avaliado, utilizando uma amostra rotulada de pares, remetendo o usuário a um dispendioso processo de validação manual. No entanto, esse processo pode ser extremamente impreciso em grandes bases de dados, devido à dificuldade da seleção de um conjunto de treinamento representativo. Além disso, o tempo gasto para promover a execução com diferentes configurações (variando o valor dos limiares) pode ser impraticável.

A Figura 1.2 ilustra a dificuldade de se identificar o valor do limiar ideal para as etapas de blocagem e classificação em uma base de dados sintética[3]. No experimento, a qualidade da deduplicação foi avaliada pela métrica F1[4]. Observe que, na figura, o limiar ideal pode ser identificado com o valor 0,4. A escolha de um limiar próximo ao ideal (com valor 0,3 ou 0,5, por exemplo) pode reduzir em mais de 10% a eficácia da deduplicação. Além disso, devido aos diferentes padrões de ruído, presentes nas bases de dados (especialmente de domínios distintos), dificilmente um valor de limiar ideal pode ser reutilizado pelo deduplicador.

Na intervenção indireta (também conhecida como supervisionada), o usuário não especialista é responsável pela tarefa de selecionar e rotular pares com intuito de produzir um conjunto de treinamento da base de dados (BILENKO; MOONEY, 2003; CARVALHO et al., 2006, 2008, 2012). Nesse contexto, o usuário não depende do conhecimento prévio para a tarefa de rotulação dos pares. O problema surge em como selecionar um conjunto reduzido de pares sem perdas de características relevantes, ou seja, o conjunto de treinamento deve conter uma informatividade similar ao conjunto de pares não rotulados. De fato, o conjunto de treinamento, criado pelo usuário, pode conter um número acentuado de pares pouco informativos (pares que, quando rotulados, não acrescentam informações relevantes ao conjunto de treinamento, aumentando o custo manual da deduplicação).

Para contornar o problema da criação manual do conjunto de treinamento, métodos

[3]Neste experimento, o deduplicador empregado foi proposto por VERNICA; CAREY; LI (2010), como detalhado na seção 2.3.2.1. A base de dados utilizada foi criada sinteticamente pelo gerador de dados Dsgen (CHRISTEN; CHURCHES, 2002), como detalhado na seção 5.1.

[4]A métrica F1 promove uma ponderação da precisão e da revocação (detalhada na Seção 5.2))

de aprendizagem ativa são propostos com o objetivo de promover a seleção do conjunto de treinamento, mantendo um certo controle sobre os pares que são rotulados pelo usuário (ERTEKIN et al., 2007; FREITAS et al., 2010; SARAWAGI; BHAMIDIPATY, 2002; ARASU; GOTZ; KAUSHIK, 2010; BELLARE et al., 2012; SETTLES, 2010). A aprendizagem ativa pode reduzir em até duas ordens de magnitude o número de pares manualmente rotulados, se comparada com abordagens randômicas de seleção de pares, produzindo uma eficácia competitiva (SARAWAGI; BHAMIDIPATY, 2002). De modo geral, os métodos de aprendizagem ativa promovem um modelo incremental em que o usuário é interativamente requisitado para rotular pares que são considerados relevantes para complementar a informatividade do conjunto de treinamento. No entanto, em grandes bases de dados, o bom funcionamento dos métodos de aprendizagem ativa depende da etapa de blocagem para a geração de um conjunto factível de pares candidatos, evitando a geração quadrática de pares (ARASU; GOTZ; KAUSHIK, 2010; BELLARE et al., 2012). Em outras palavras, novamente o usuário é responsável pelo ajuste manual das variáveis que configuram o processo de blocagem possibilitando a posterior execução do método de aprendizagem ativa.

Com base nos problemas apresentados anteriormente, duas grandes questões são motivadoras para o desenvolvimento da tese aqui proposta:

- ausência de métodos que abordem as etapas internas da deduplicação (ou seja, as etapas de blocagem e classificação) como uma única tarefa, na perspectiva da intervenção do usuário;
- a necessidade de que um usuário "especialista" no domínio participe do processo de configuração da deduplicação, especialmente no contexto de grandes bases de dados.

1.1 Objetivos e Contribuições

Devido ao crescente volume de informações, cria-se a necessidade de novas soluções para a deduplicação, que sejam capazes de processar a dimensão de informações presentes nas bases de dados atuais. Esta tese tem como foco reduzir o esforço do usuário no processo de calibração da deduplicação de grandes bases de dados.

Especificamente, são explorados os seguintes desafios:

- como identificar os valores de limiares capazes de "configurar" idealmente a deduplicação de grandes bases de dados, removendo a necessidade de um usuário especialista nas principais etapas da deduplicação (blocagem e classificação);
- como limitar a intervenção do usuário a somente um "conjunto" reduzido de pares manualmente rotulados.

Para o primeiro desafio, é proposta uma solução capaz de identificar os valores ideais de limiares para a configuração do processo de deduplicação. O volume acentuado de dados naturalmente exige que as soluções sejam especificamente sintonizadas para cada conjunto de dados. Tal sintonia fina restringe a configuração da deduplicação somente a usuários com conhecimento prévio do domínio em questão e do funcionamento da abordagem de deduplicação a ser empregada. Portanto, pretende-se reduzir o custo manual da deduplicação de grandes bases de dados, possibilitando que um usuário não especialista possa configurar a deduplicação sem a necessidade de interagir com as etapas internas do

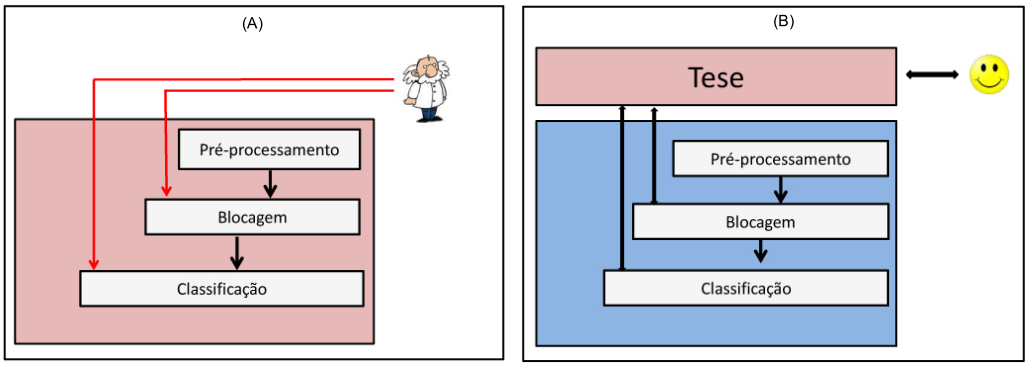


Figura 1.3: Exemplo do esforço do usuário na configuração da deduplicação (A) e da redução da intervenção do usuário proposta na presente tese (B).

processo. Para isso, é projetada uma metodologia capaz de mapear as informações presentes em um reduzido conjunto de pares, rotulados pelo usuário, para as configurações requisitadas pelo deduplicador. Na metodologia, uma heurística é proposta com o intuito de produzir um conjunto controlado de pares candidatos sem a intervenção do usuário. Além disso, é proposta uma estratégia para a identificação dos pares considerados mais desafiadores para o processo de classificação, ou seja, pares que possuem um alto grau de ambiguidade). Dessa forma, é possível aplicar métodos de classificação mais específicos, visando facilitar a predição dos pares mais críticos.

Para o segundo desafio, é proposto um novo método para a seleção de pares informativos a partir de dois passos. No primeiro passo, são selecionadas amostras randômicas de pares em diferentes porções da base com o intuito de produzir conjuntos reduzidos e balanceados de pares. Em seguida, cada amostra é incrementalmente processada por um algoritmo de aprendizagem ativa, de modo que sejam rotulados manualmente somente os pares que acrescentam ganho de informação ao conjunto de treinamento. O primeiro passo é fundamental para reduzir o alto custo computacional do método de aprendizagem ativa, permitindo a deduplicação em grandes bases de dados. A partir do reduzido conjunto de treinamento, criado pelos dois passos, são extraídas as informações necessárias para identificar os pares mais desafiadores no conjunto de pares candidatos e configurar as abordagens de classificação.

De modo geral, a presente tese tem como objetivo preencher a lacuna de como reduzir a intervenção do usuário, o tanto quanto possível, sem depreciar a qualidade da deduplicação. A Figura 1.3-(A) ilustra como o usuário interage tradicionalmente na deduplicação de grandes conjuntos de dados. Note-se que o usuário deve ser capaz de determinar a configuração das etapas de blocagem e classificação. A Figura 1.3-(B) ilustra o objetivo da tese, no qual o usuário é isolado da tarefa de configurar as etapas da deduplicação. Mais ainda, a configuração da deduplicação é realizada a partir da rotulação manual de um conjunto reduzido de pares selecionados de forma automática pelas abordagens propostas na tese.

1.2 Organização da tese

O restante deste texto está organizado como descrito a seguir:

- no Capítulo 2, é apresentada uma revisão bibliográfica que aborda os trabalhos relacionados com a presente tese. Uma nova classificação dos trabalhos presentes na

bibliografia é proposta para facilitar o entendimento e, da mesma forma, evidenciar as possíveis lacunas presentes na bibliografia. É importante salientar que diversos trabalhos apresentados neste capítulo podem fazer uso do método proposto nesta tese e que serão indicados no momento oportuno;

- no Capítulo 3, é apresentada uma descrição detalhada da metodologia proposta para a calibração do processo de deduplicação em grandes bases de dados. O capítulo apresenta as estratégias descritas para possibilitar que o usuário seja liberado da tarefa de conhecer detalhes internos das principais etapas da deduplicação. Primeiramente, uma heurística é utilizada para identificar o limiar de blocagem capaz de maximizar a geração dos pares duplicados, sem a intervenção do usuário. Em seguida, é apresentada uma estratégia para identificar o conjunto de pares candidatos mais desafiadores de serem classificados, fazendo uso de um modelo iterativo para a seleção e, posterior, rotulação manual dos pares pertencentes ao conjunto de treinamento. Por fim, são apresentados dois métodos para a classificação dos pares duplicados;

- no Capítulo 4, é detalhada uma nova abordagem para a seleção de uma amostra informativa de pares, que visa a redução do esforço do usuário. Esse capítulo apresenta os passos dessa abordagem responsáveis pela seleção de uma amostra reduzida e altamente informativa na presença de grandes bases de dados. Além disso, é demostrado como a abordagem proposta nesse capítulo é integrada à metodologia apresentada no Capítulo 3 com intuito de reduzir, ainda mais, a redução da intervenção do usuário;

- no Capítulo 5, é promovida uma extensiva avaliação experimental a fim de validar as propostas apresentadas por esta tese. Esse capítulo tem como objetivo avaliar as propostas apresentadas nos Capítulos 3 e 4;

- no Capítulo 6, são apresentadas as conclusões, destacando os resultados obtidos por esta tese. São apresentadas, também, as publicações resultantes do doutorado. Por fim, são descritas as possíveis direções de pesquisa a serem desenvolvidas no futuro.

2 TRABALHOS RELACIONADOS

Este capítulo apresenta um estudo bibliográfico relativo aos trabalhos relacionados ao escopo da tese. Na Seção 2.1, são apresentados os principais conceitos relevantes para um melhor entendimento dos trabalhos relacionados a esta tese. A Seção 2.2 descreve a classificação, na qual os trabalhos relacionados estão organizados. Na Seção 2.3, são detalhados os principais trabalhos relacionados, segundo a classificação proposta. Por fim, as considerações finais são apresentadas na Seção 2.4, visando ressaltar os principais desafios de pesquisa identificados por este estudo.

2.1 Conceitos Fundamentais

Os métodos de deduplicação podem ser divididos em dois grandes grupos: supervisionados e não supervisionados (COSTA et al., 2011). Os métodos supervisionados assumem a presença amostras rotuladas para inferir uma função de classificação. Os pares não rotulados são submetidos a uma função de classificação para a predição dos rótulos. Já os métodos não supervisionados utilizam o grau de similaridade entre os registros como uma evidência para promover a classificação dos pares. Um detalhamento dos métodos supervisionados e não supervisionados podem ser encontrado em diferentes levantamentos bibliográficos (ELMAGARMID; IPEIROTIS; VERYKIOS, 2007; KOPCKE; THOR; RAHM, 2010).

De um modo geral, a intervenção do usuário, nos métodos não supervisionados, é caracterizada pela definição de valores de variáveis ou limiares que conduzem o processo de deduplicação. Em outras palavras, o usuário interage diretamente na calibração do deduplicador. Os valores de limiares são empregados, entre outras funções, como critério para o mapeamento dos pares como: duplicata, caso o par apresente um grau de similaridade acima do valor definido pelo limiar; ou não duplicata, se a similaridade do par não atingir o limiar estipulado. A similaridade de cada par pode ser calculada a partir de uma variada gama de medidas ou funções de similaridade, que podem ser divididas em dois grandes grupos (mas não se restringem a estes) (COHEN; RAVIKUMAR; FIENBERG, 2003; NAVARRO, 2001; HADJIELEFTHERIOU; SRIVASTAVA, 2011)[1]:

- baseadas em caracteres - utilizam como evidência o posicionamento dos caracteres para avaliar a similaridade entre os registros. Tais funções são projetadas para computar o número mínimo de variações (inserção, remoção ou alteração) necessárias para transformar uma primeira *string* em uma segunda *string*. No entanto, as funções baseadas em caracteres são consideradas computacionalmente custosas,

[1] O detalhamento de cada função, em específico, foge do escopo desta tese e pode ser encontrado em diferentes trabalhos, tais como ELMAGARMID; IPEIROTIS; VERYKIOS (2007); GUSFIELD (1997).

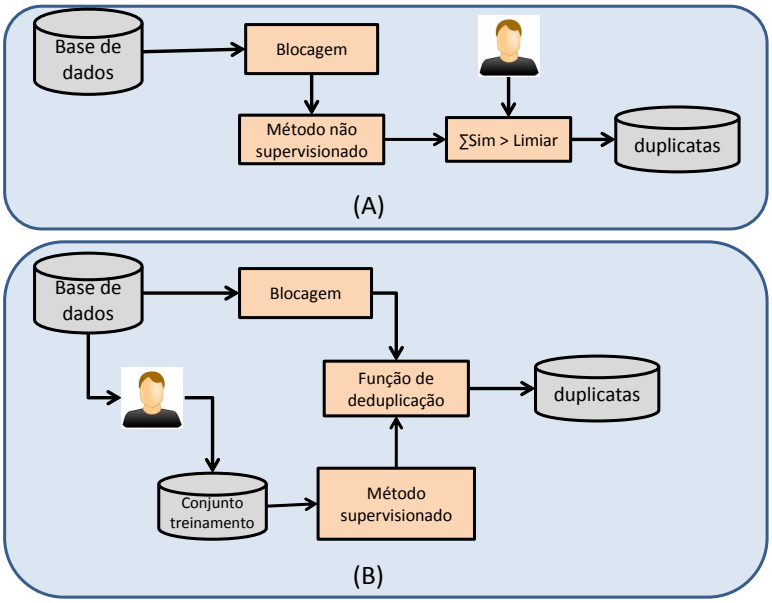


Figura 2.1: Ilustração da intervenção do usuário nos métodos não supervisionados (A) e nos métodos supervisionados (B). Na figura, a configuração da etapa de blocagem foi abstraída para facilitar a visualização.

devido à necessidade de comparar todos os caracteres para a quantização da similaridade. Nesse grupo, é possível citar as seguintes funções de similaridade: *distância de edição*, *Levenshtein*, *Jaro*, *Jaro Winkler*, etc;

- baseadas em *tokens*[2] - têm como característica a fragmentação da base de dados em conjuntos de *tokens* para quantização da similaridade utilizando o relacionamento entre conjuntos (interseção ou união de elementos). De uma forma simplista, o valor de similaridade é calculado a partir do casamento exato entre os *tokens*. No entanto, o casamento exato impede a detecção de pequenas variações, como, por exemplo, erros de tipografia, plural, abreviações, etc. Para minimizar esse problema, são adicionadas técnicas de tokenização para a fragmentação dos *tokens* em unidades menores. A tokenização mais intuitiva leva em consideração a segmentação em nível de termos, ou seja, os registros são particionados utilizando como critério a presença do caractere de espaços entre os termos. Na tokenização em nível de NGram[3], os termos são fragmentados em pequenas unidades, chamadas NGram, para melhorar a precisão do processo de quantização da similaridade entre os *tokens*. Na tokezinação NGram, é utilizada a sobreposição de caracteres (ou seja, a redundância de dados) para evitar possíveis perdas de informações na fragmentação dos termos. Notavelmente, a inserção da redundância resulta em um custo adicional de processamento que pode ser relevante em grandes conjuntos de dados. Alguns exemplos de funções de similaridade pertencentes a esse grupo são: *Jaccard*, *Dice*, *Cossine*, *Inverse document frequency (IDF)*, etc.

[2]Nesta tese, o termo *token* é utilizado de forma abstrata para descrever o conteúdo de um atributo, um termo (ou palavra), ou uma *substring*.

[3]Um NGram representa uma sequência contínua de N caracteres em uma *string*.

Na perspectiva da intervenção do usuário, os métodos supervisionados assumem a existência de uma amostra capaz de representar, idealmente, os padrões presentes nas bases de dados. O usuário interage indiretamente com o método, fornecendo uma amostra rotulada. A qualidade da classificação, em tais métodos, está intimamente relacionada ao grau de informatividade da amostra de treinamento. Em outras palavras, uma amostra, contendo um número substancialmente alto de pares, pode configurar a tarefa de classificação com uma qualidade inferior a uma outra amostra, contendo um número reduzido de pares, mas altamente "representativo".

A Figura 2.1 ilustra graficamente como o usuário interage com cada um dos métodos descritos anteriormente. Observe que, nos métodos não supervisionados (Figura 2.1-(A)), o usuário é responsável diretamente pela qualidade dos pares recuperados a partir da definição de valores de limiares. No caso dos métodos supervisionados, é demandado uma amostra previamente rotulada para a construção de um modelo de classificação (ou função de classificação), como pode ser observado na Figura 2.1-(B). Como alternativa, nos métodos supervisionados, a amostra rotulada pode ser criada ou complementada utilizando métodos de aprendizagem ativa para a seleção dos pares mais informativos. Mais detalhes dos métodos de aprendizagem ativa são descritos no decorrer deste capítulo.

2.2 Classificação da Literatura

Esta seção apresenta a classificação proposta, na qual, os trabalhos relacionados são categorizados. O objetivo é facilitar a descrição e, por consequência, o entendimento do levantamento bibliográfico descrito neste capítulo. Mais especificamente, os trabalhos relacionados estão divididos segundo dois níveis hierárquicos: *(i)* escalabilidade, que representa a capacidade dos métodos em processar grandes volumes de dados; e *(ii)* o grau de intervenção do usuário, ou seja, como o usuário interage para configurar o método de deduplicação. A classificação proposta permite identificar a necessidade de projetar novas soluções que complementam o atual estado da arte da deduplicação.

Tipicamente, a deduplicação de um volume acentuado de dados exige a presença de métodos de blocagem para a redução do custo computacional. A blocagem visa produzir grupos de registros que compartilham características em comum. Pares candidatos são produzidos somente entre registros pertencentes a um mesmo bloco (CHRISTEN, 2007; BAXTER; CHRISTEN; CHURCHES, 2003).

Idealmente, a blocagem deve ser capaz de evitar que registros não duplicados sejam erroneamente inseridos nos blocos, garantindo que cada bloco seja composto somente por pares duplicados. Por exemplo, em uma tabela de dados cadastrais referentes a pessoas, o critério de blocagem pode ser definido como o "ano de nascimento", fragmentando os registros em cerca de 100 blocos. Assim, a blocagem permite reduzir substancialmente o número de pares candidatos, possibilitando o processamento de grandes volumes de dados em um tempo factível (CHRISTEN, 2012). No entanto, não é possível garantir que um deduplicador dependa exclusivamente da efetividade da blocagem para alcançar um alto grau de eficiência e escalabilidade. Abordagens ineficientes, mesmo que empregando a blocagem, podem apresentar uma escalabilidade restrita a pequenos conjuntos de dados. Uma comparação direta entre as abordagens de deduplicação presentes na bibliografia, com foco na escalabilidade, é inviável devido à ausência de uma padronização experimental (KOPCKE; THOR; RAHM, 2010).

Dessa forma, no primeiro nível da *classificação* proposta, é assumido como pressuposto que o emprego de técnicas blocagem oferece um indício sobre a capacidade do

método em processar grandes montantes de dados. No segundo nível, os trabalhos são categorizados utilizando como critério o grau de intervenção do usuário. Foge do escopo desta tese discutir, em profundidade, abordagens que tratam exclusivamente da etapa de blocagem devido ao foco desta tese ser direcionado para o problema da deduplicação de dados como um todo.

A intervenção do usuário foi fragmentada em quatro níveis distintos, como descrita a seguir:

1. *definição de valores de limiares* - o usuário, especialista no domínio, intervém a partir da definição de valores de limiares. Tais limiares são responsáveis, entre outras funções, pela definição da similaridade mínima que um par deve apresentar para ser considerada uma duplicata.

2. *definição de valores de limiares e rotulação de pares* - a intervenção, nesta categoria, ocorre em duas etapas. Em uma primeira etapa, um limiar é utilizado para a configuração da etapa de blocagem ou para a calibração das configurações internas do deduplicador. Em uma segunda etapa, a rotulação do usuário é requisitada para criar ou ampliar o conjunto de treinamento, utilizado pelo processo de classificação. O conjunto rotulado (também chamado de conjunto de treinamento) é fundamental para que o deduplicador identifique os padrões ou comportamentos presentes na base de dados.

3. *rotulação de pares* - a intervenção do usuário se manifesta somente a partir da rotulação de um conjunto de pares. Diferentemente do nível descrito anteriormente, a intervenção a partir da rotulação de pares é capaz de configurar por completo o processo de deduplicação. Dessa forma, é removida a necessidade de que o usuário detenha o conhecimento prévio no domínio.

4. *automática* - a deduplicação ocorre sem nenhuma intervenção do usuário. Utilizando-se de heurísticas, são extraídos os padrões para a classificação automática dos pares presentes no conjunto de dados.

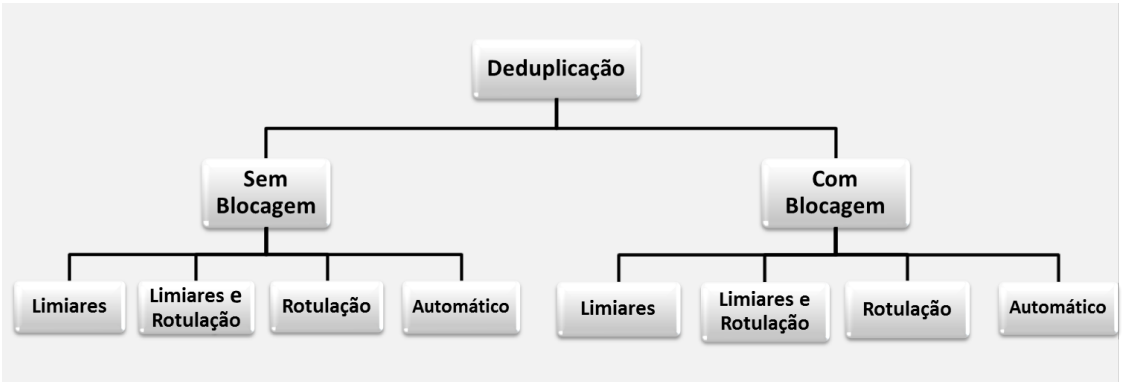


Figura 2.2: Representação da *Classificação* proposta para os trabalhos diretamente relacionados à tese.

Observe que na *classificação* proposta, a intervenção do usuário foi dividida a partir de níveis mais complexos (ou seja, restrito a usuários especialistas capazes de definir os corretos valores de limiares) até a ausência completa da dependência do usuário. A intervenção a partir da *rotulação de pares* representa o nível que mais se aproxima da

automatização, dependendo somente de um usuário não especialista (sem conhecimento prévio no domínio) para a configuração da deduplicação. A Figura 2.2 ilustra graficamente a *classificação* proposta.

2.3 Revisão da Literatura

A seguir, os principais trabalhos relacionados são categorizados, segundo a *classificação* proposta, e apresentados com objetivo de evidenciar os principais conceitos e contribuições. Detalhes de mais baixo nível de abstração (ou no nível de implementação), apesar de representarem uma contribuição relevante, são sucintamente apresentados para facilitar o entendimento. É importante salientar que esta seção tem como objetivo fundamentar, contextualizar e projetar o leitor para a discussão da motivação da presente tese.

2.3.1 Métodos de deduplicação que não exploram a *Blocagem*

A aplicação do produto cartesiano entre os registros da base de dados assegura que todos pares duplicados sejam gerados. No entanto, tal garantia é computacionalmente viável somente em pequenos conjuntos de dados, como apresentado nos trabalhos desta subseção.

2.3.1.1 Intervenção do usuário a partir da definição de limiar

FELLEGI; SUNTER (1969) - F&S

FELLEGI; SUNTER (1969) desenvolveram o primeiro modelo formal consolidado na bibliografia para deduplicação baseado em probabilidades, previamente esboçado por NEWCOMBE et al. (1959). O modelo de F&S é utilizado como fundamentação teórica para a deduplicação até os dias atuais (CHRISTEN; CHURCHES; HEGLAND, 2004; WINKLER, 1999; DAL BIANCO et al., 2013).

O modelo probabilístico de F&S visa otimizar a classificação dos pares de duas bases de dados (por exemplo, A e B) em três conjuntos: (*i*) *M*, composto somente por pares duplicados; (*ii*) P, composto por possíveis duplicatas; e (*iii*) *U*, composto por pares não duplicados. Tipicamente, pares pertencentes ao conjunto *M* possuem campos relevantes em comum, tal como, o título da publicação, autor(es), ano de publicação (no contexto de bibliotecas digitais). Pares pertencentes ao conjunto U apresentam poucos indícios de representarem uma duplicata, como, por exemplo, no contexto de uma biblioteca digital, um par pode apresentar somente o ano de publicação semelhante. Por fim, o conjunto P é composto pelos pares com alto grau de ambiguidade, ou seja, não apresentam indícios suficientes para serem definidos como duplicatas ou para serem descartados. Novamente, utilizando o exemplo de uma biblioteca digital, um par pertencente ao conjunto P pode conter o título da publicação similar, mas com variações substanciais nos nomes dos autores.

No modelo proposto, os *padrões de concordância*(γ) são aplicados para identificar se campos de registros (a_k, b_k) apresentam algum indício de casamento ou correspondência. Um *padrão de concordância* simplista, por exemplo, identifica se um determinado campo apresenta valores idênticos (γ=1). Um modelo de probabilidade é utilizado para determinar um peso w para cada γ, como definido a seguir:

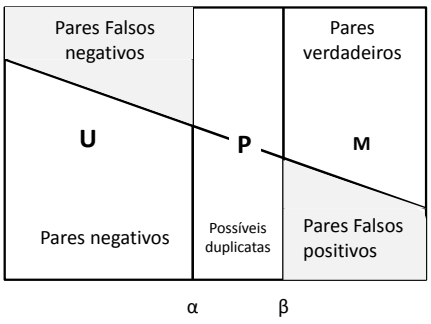


Figura 2.3: Exemplificação da divisão dos pares candidatos seguindo o modelo F&L.

$$\begin{aligned} w_k &= \frac{\ln(m_k)}{\ln(u_k)} \qquad se(\gamma_k = 1) \\ w_k &= \frac{\ln(1-m_k)}{\ln(1-u_k)} \qquad se(\gamma_k = 0) \end{aligned} \tag{2.1}$$

as variáveis m_k e u_k representam a probabilidade condicional do campo *k* possuir um valor correspondente (ou não correspondente), como definido a seguir:

$$\begin{aligned} m_k &= Prob(\gamma = 1 | r_{ij} \in \boldsymbol{M}) \\ u_k &= Prob(\gamma = 0 | r_{ij} \in \boldsymbol{U}) \end{aligned} \tag{2.2}$$

O peso w_k (descrito na Equação 2.1) resultará em um valor positivo, caso o par seja considerado uma duplicata e um valor negativo para pares não duplicados. O modelo de decisão é construído a partir da soma dos pesos $(w_k, w_j, ..., w_n)$ e comparado a dois valores de limiares α e β. F&S propuseram que as regras de decisão são definidas pelas seguintes condições:

- se $\sum w_{k..n} > \beta$, então o par é inserido no conjunto ***M***;
- se $\alpha < \sum w_{k..n} < \beta$, então o par é inserido no conjunto ***P***;
- se $\sum w_{k..n} < \alpha$, então o par é inserido no conjunto ***U***;

Os limiares α e β são diretamente responsáveis pela inserção dos pares nos conjuntos: U, P ou M. O conjunto P deve ser enviado para um processo de rotulação manual devido à impossibilidade de classificação pelo modelo proposto. Segundo o modelo probabilístico de F&S, o conjunto P contém os pares mais desafiadores em relação ao processo de classificação. Neste trabalho, o conjunto P é chamado de *Região Crítica*.

A Figura 2.3 ilustra as três regiões (ou conjuntos) resultantes do processo de deduplicação, seguindo o modelo de F&L. Os valores dos limiares são definidos respeitando taxas de erros aceitáveis (como ilustrado na região cinza da Figura 2.3). Como esperado, os pares não duplicados estão concentrados na região U. Note na região U, a presença de pares duplicados (chamados de pares falsos negativos) devido uma esperada taxa de erro na definição dos limiares. Os pares duplicados estão concentrados na região M, adicionados de pares não duplicados (chamados de pares falsos positivos). Na região entre α e β se concentram os pares de possíveis duplicatas, caracterizando a região crítica da base de dados.

GRAVANO et al. (2001)

GRAVANO et al. (2001) têm como objetivo melhorar a eficiência da deduplicação explorando a utilização de operadores primitivos oferecidos pelos bancos de dados relacionais (por exemplo, *group by*, *having*, *count*, etc.). O método promove o produto cartesiano da base de dados (ou tabela de dados) para a identificação dos pares, cujos valores de similaridade[4] sejam menores que um limiar, previamente definido. No entanto, o produto cartesiano da tabela de dados resulta em um número quadrático de pares candidatos. Nesse contexto, a principal contribuição da abordagem é a inserção dos *filtros de posição, tamanho e contagem* para a redução do número de pares candidatos produzido pelo produto cartesiano.

GRAVANO et al. (2001) utilizam o método de tokenização baseado em NGram (CAVNAR; TRENKLE, 1994) para segmentar as *string* em *substrings* (ou *tokens*), possibilitando que pequenos erros de tipografia não inviabilizem a identificação de pares duplicados. No entanto, tal tokenização produz um número relativamente alto de combinações de *substrings* podendo causar um impacto substancial na eficiência do método. Para contornar esse problema, é adicionada uma etapa de pré-processamento para o armazenamento das *substrings* em tabelas adicionais. As tabelas possibilitam acessar rapidamente o mesmo *token* inúmeras vezes, com a penalidade do aumento do volume de informações armazenadas no banco de dados.

Uma substancial melhora na eficiência do método é alcançada a partir da inserção dos *filtros* após a geração dos pares candidatos. Três *filtros* são propostos, como descrito a seguir:

- *filtro de contagem* - todos os pares candidatos devem apresentar um número mínimo de NGram em comum, definido por um valor de limiar.

- *filtro de posição* - a variação no posicionamento dos NGram dos pares deve ser menor que um valor de limiar.

- *filtro de tamanho* - a diferença de tamanho entre os registros não pode ser maior que um valor de limiar.

Os pares que sobreviverem aos *filtros* são enviados para a etapa de cálculo da similaridade. A intuição da abordagem é que todos (ou quase todos) os pares falsos positivos possam ser removidos pelos filtros, reduzindo o número de pares candidatos analisados pela função de similaridade.

A análise experimental, conduzida pelos autores, demonstrou que a combinação dos filtros resultou na remoção substancial de pares candidatos, se aproximando do número real de duplicatas. Os experimentos foram conduzidos em três bases de dados reais de 30 mil a 40 mil registros. Não foram discutidas abordagens para a blocagem dos pares. O usuário intervém a partir da definição de valores de limiares que são utilizados para configuração do processo de filtragem.

2.3.1.2 Intervenção do usuário a partir da definição de limiares e da rotulação de pares

CARVALHO et al. (2006, 2008, 2012)

[4] A função de similaridade utilizada neste trabalho foi a *distância de edição* (ELMAGARMID; IPEIROTIS; VERYKIOS, 2007). Em tal função, quanto menor for o número de edições ou variações mais similar é considerado o par.

Em CARVALHO et al. (2006), é projetado um deduplicador que explora a Programação Genética[5](PG) para a identificação de evidências relevantes para melhorar a eficácia da classificação dos pares. A proposta, baseada em PG, tem como vantagem a capacidade de identificar os atributos mais representativos, a partir de um conjunto de treinamento, descartando atributos que não agregam informações ao processo de identificação de duplicatas.

O deduplicador baseado em PG combina os atributos utilizando operadores matemáticos básicos (por exemplo: soma, divisão, subtração e multiplicação) para a criação de uma estrutura em formato de árvore. A cada geração (ou *rodada*) um novo conjunto de árvores é criado, a partir do cruzamento das árvores anteriores. Uma função de poda é aplicada para remover árvores menos eficazes, segundo critérios previamente definidos (por exemplo, um valor de precisão mínimo). A árvore resultante, chamada de função de deduplicação, tem como característica reduzir a importância (ou descartar) de atributos pouco informativos e adicionar pesos mais elevados aos atributos mais relevantes para a classificação.

A avaliação do deduplicador proposto por Carvalho foi realizada utilizando uma base de dados sintética contendo 1.000 registros e uma base de dados real composta por 2.000 registros. A técnica baseada em PG foi comparada com a abordagem F&S com objetivo de avaliar a eficácia. Para fins de comparação, os limiares do deduplicador Carvalho e F&S foram fixados com valores de 0 e 30. A avaliação experimental comprovou que o uso de um número reduzido de atributos é suficiente para melhorar a eficácia do processo, se comparado com a abordagem F&S. A intervenção do usuário é demandada para a criação de um conjunto de treinamento (ou seja, uma amostra rotulada), dois limiares (seguindo o modelo F&S, descrito na seção 2.3.1.1) e da definição de quais funções de similaridade são utilizadas por cada atributo.

Em CARVALHO et al. (2008, 2012), a proposta de CARVALHO et al. (2006) é estendida com objetivo de automatizar a seleção das funções de similaridade. As funções são avaliadas pelo deduplicador baseado em PG buscando identificar, durante o processo evolucionário, quais são as mais adaptáveis a cada atributo (por exemplo, atributos contendo textos, datas, nomes, números, entre outros). Adicionalmente, foi realizada uma avaliação empírica da seleção automática do limiar de classificação, a partir de um conjunto de treinamento.

A avaliação experimental foi conduzida em bases de dados reais e sintéticas variando de 500 a cerca de 1.300 registros. Na experimentação, foi avaliado o potencial da PG em discernir, entre um conjunto de funções de similaridade, qual é a função que melhor quantifica a similaridade de cada atributo. A avaliação do autor concluiu que a PG é capaz de melhorar a eficácia, se comparando com a seleção manual de funções de similaridade. A avaliação experimental concluiu também que é possível maximizar métrica F1[6] a partir da definição de um baixo valor de limiar. O valor de limiar é explicado pela forma com que o algoritmo de PG constrói a topologia da árvore, ou seja, a partir de múltiplas combinações de operações matemáticas de multiplicação com pesos abaixo do valor 1,0. Desse modo, é possível alcançar uma aproximação do limiar ideal, utilizando somente uma amostra de treinamento.

[5]A Programação Genética pode ser vista como uma técnica de aprendizagem de máquina, na qual a ideia básica é inspirada na teoria da *seleção natural* (KOZA; POLI, 2005).

[6]A métrica *F1* ou *F-measure* é uma medida que calcula a média harmônica ponderada dos valores de precisão e de revocação. Um detalhamento das métricas utilizadas nesta proposta é feito na seção 5.2.

FREITAS et al. (2010) - AGP

O objetivo do deduplicador proposto por FREITAS et al. (2010) é reduzir a intervenção manual sem o detrimento da eficácia. Segundo FREITAS et al. (2010), o deduplicador proposto por CARVALHO et al. (2006) tem como desvantagem a necessidade da seleção e rotulação manual de um conjunto de treinamento. É proposto um deduplicador, intitulado de AGP, que estende a proposta de CARVALHO et al. (2006) com o objetivo de selecionar um conjunto reduzido de pares para compor o conjunto de treinamento, a partir da aprendizagem ativa.

Inicialmente, o AGP constrói um *ranking* utilizando como critério o grau de similaridade de cada par. Duas amostras são selecionadas a partir do *ranking*: uma primeira amostra contendo os pares mais similares (na parte superior do *ranking*); e uma segunda amostra composta pelos pares mais dissimilares (ou seja, na parte inferior do *ranking*). Em seguida, as amostras são enviadas para a rotulação manual.

A partir do conjunto de treinamento inicial criado pelas duas amostras, o algoritmo de classificação baseado em PG inicia o processo de treinamento. Um conjunto de *funções de deduplicação* (descrita na Seção 2.3.1.2) é escolhido para compor um comitê de classificação. Tal comitê é responsável pela seleção dos pares considerados relevantes para complementar a informatividade do conjunto de treinamento. Mais especificamente, caso os membros do comitê concordem com a classificação de um par (ou seja, todos identifiquem como pertencente a uma mesma classe), é assumido que o par não apresenta ambiguidades e pode ser automaticamente anexado ao conjunto de treinamento. No caso contrário, se os membros do comitê resultarem em uma divergência, o par candidato é manualmente avaliado para corrigir as possíveis divergências. A realimentação é utilizada para corrigir as *funções de deduplicação* que resultarem em avaliações incorretas, segundo a rotulação do usuário.

Os experimentos propostos por FREITAS et al. (2010) analisaram a eficácia e o número de amostras rotuladas para o treinamento do algoritmo de classificação. Os experimentos foram conduzidos em três bases de dados de dados sintéticas e reais, contendo um montante de 250 a 1.000 registros. O AGP foi comparado com os métodos propostos por Carvalho et al. e ALIAS. Na experimentação, o AGP manteve a eficácia na deduplicação, mas convergiu[7] com um número reduzido de amostras, manualmente rotuladas, se comparado aos *baselines* analisados.

2.3.1.3 Intervenção do usuário a partir da rotulação de pares

SARAWAGI; BHAMIDIPATY (2002) - ALIAS

Em SARAWAGI; BHAMIDIPATY (2002), é projetada uma abordagem para deduplicação com objetivo de reduzir a intervenção manual. O método, intitulado de ALIAS, utiliza a aprendizagem ativa para identificar os pares mais representativos para a criação do conjunto de treinamento. ALIAS parte da premissa de que pares com alto grau de semelhança (ou alto grau de divergência) podem ser facilmente classificados, portanto, têm uma baixa influência no ganho de informatividade do conjunto de treinamento. Por outro lado, os pares mais ambíguos (ou seja, aqueles pertencentes à região crítica da base de dados) são os mais informativos e possibilitam discernir de uma forma mais precisa os pares duplicados dos pares não duplicados com um número reduzido de amostras.

[7] O termo convergir, neste contexto, é utilizado para descrever que foram disponibilizadas amostras suficientes para que o método atingisse o critério de parada, previamente definido.

Primeiramente, na abordagem ALIAS, é selecionado e rotulado um pequeno conjunto inicial de pares (cerca de 10 pares). Função (ou funções) de similaridade, definidas pelo usuário, quantificam a similaridade de cada par. O conjunto inicial e os respectivos valores de similaridade são utilizados para o treinamento inicial dos algoritmos de classificação. Um comitê (ou conjunto) de classificadores é utilizado para identificar os pares mais informativos ou ambíguos para a classificação. Pares com alto/baixo grau de similaridade são facilmente identificados pelos membros do comitê. Em contrapartida, pares com certo grau de ambiguidade, possivelmente resultam em uma divergência entre os classificadores. São propostos três métodos de constituir o comitê de classificadores, como descrito a seguir:

1. *randomização de parâmetros* - são gerados diferentes formatos de árvores de decisão para a composição do comitê. A variação na configuração das árvores é feita a partir de variações ou perturbações dos parâmetros durante a criação das árvores de decisão. Assim, a árvore ganha um novo desenho (ou configuração) que influência diretamente no processo de classificação;

2. *divisão do conjunto de treinamento* - o conjunto de treinamento é dividido em diferentes subconjuntos que são utilizados para o treinamento do classificador. O treinamento pode ser efetuado a partir da combinação de subconjuntos isolados ou sobrepostos;

3. *partição de atributos* - é baseado na remoção de alguns atributos e na aplicação dos métodos de *Randomização de parâmetros e divisão do conjunto de treinamento* (como descritos nos itens anteriores). Dessa forma, é possível definir classificadores com diferentes formações para a criação de um comitê. A remoção de determinados atributos pode resultar na perda de informações relevantes, resultando na divergência da predição entre os membros do comitê.

Por fim, os pares classificados pelo comitê como ambíguos são enviados para a avaliação do usuário. O usuário pode corrigir a rotulação do comitê, caso não concorde com a predição dos pares. A possibilidade de correção da amostra rotulada, durante a execução, determina a interatividade da abordagem. O subconjunto, que foi validado pelo usuário, é adicionado ao conjunto rotulado e o classificador é novamente treinado. Se o classificador não produzir uma saída esperada pelo usuário, é feita uma nova seleção de pares para serem incorporados ao conjunto de treinamento. Caso a predição do classificador persista na incoerência, a função de similaridade pode ser ajustada ou alterada manualmente. O processo de rotulação ativa prossegue até atingir o nível de qualidade esperado pelo usuário.

A abordagem ALIAS discute métodos de blocagem para reduzir o número quadrático de pares candidatos. No entanto, ALIAS não incorpora nenhuma técnica de blocagem, restringindo sua experimentação a pequenos conjuntos de dados contendo 80 a 300 registros. Segundo a experimentação dos autores, o método proposto obteve uma redução de duas ordens de magnitude no número de pares manualmente rotulados, se comparado com métodos tradicionais de classificação supervisionada (como exemplo, SVMs (MANNING; RAGHAVAN; SCHTZE, 2008) e as árvores de decisão (QUINLAN, 1996)).

ERTEKIN et al. (2007)

ERTEKIN et al. (2007) propõem uma solução genérica para a classificação de dados utilizando a *aprendizagem ativa*. O objetivo é facilitar o processo de aprendizagem com a redução da rotulação manual, em domínios com alto grau de desbalanceamento de classes[8]. O desbalanceamento de classes prejudica o processo de aprendizagem devido à difícil tarefa de selecionar os pares informativos pertencentes à classe menos representativa (classe de pares duplicados).

ERTEKIN et al. (2007) partem do princípio de que pares mais próximos do *hiperplano*[9] criado pelo algoritmo de Máquinas de Vetores de Suporte (SVMs, do inglês *Support Vector Machines*) (MANNING; RAGHAVAN; SCHTZE, 2008) são mais representativos que os pares mais distantes. Dessa forma, é proposta uma alteração no algoritmo SVM para a adequação à *aprendizagem ativa*. Mais especificamente, a cada rodada o método coleta aleatoriamente uma amostra contendo 59 pares e, a partir dos quais, é selecionado o par mais próximo do hiperplano para ser manualmente rotulado. A intuição do método é que os pares mais próximos do hiperplano são mais ambíguos e oferecem um desafio maior para o processo de classificação. Cada par rotulado é treinado individualmente e adicionado ao modelo existente, reduzindo custos de processamento.

A convergência do método (ou o critério de parada) ocorre quando os novos pares selecionados estão mais distantes do hiperplano do que os pares previamente rotulados. Isto demonstra que o método atingiu um bom estado de aprendizagem, ou seja, os novos pares rotulados não agregam novas informações ao conjunto de treinamento.

Os experimentos foram conduzidos em diferentes domínios (por exemplo, bases de documentos e imagens). O volume de dados analisado variou de 8.000 a 60.000 registros. Uma das discussões conduzida nos experimentos está relacionada ao tamanho da amostra utilizada no treinamento. Foi concluído que a adição descontrolada de pares rotulados pode reduzir a eficácia do classificador. Assim, o critério de parada proposto foi importante para identificar a necessidade ou não do aumento do número de pares rotulados.

2.3.1.4 Métodos Automáticos para a deduplicação

CHRISTEN; GOISER (2007); CHRISTEN (2008a)
Em CHRISTEN; GOISER (2007), é proposto um método com objetivo de automatizar a seleção e a rotulação dos pares para criação da amostra de treinamento. A proposta é dividida em duas etapas: (*i*) seleção dos pares, e (*ii*) o treinamento do algoritmo de classificação. A principal contribuição da abordagem está focada na etapa (*i*), detalhada a seguir.

A abordagem parte da premissa de que pares duplicados contêm atributos com uma alta similaridade (registros aproximadamente idênticos) e pares não duplicados contêm atributos com uma baixa similaridade. Mais especificamente, dois conjuntos são criados para armazenar as duplicatas (*Wm*) e as não duplicatas (*Wn*). Primeiramente, pares contendo atributos exatamente iguais são inseridos no conjunto *Wm*. Em seguida, pares contendo atributos totalmente dissimilares são inseridos no conjunto *Wn*. Para complementar o conjunto de amostras, pares na vizinhança de *Wm* e *Wn* são selecionados.

Uma estimativa da proporção entre os pares duplicados e não duplicados é definida para melhorar o aprendizado do algoritmo de classificação. Uma importante observação

[8] O desbalanceamento de classes ocorre quando o número de elementos de uma classe é substancialmente superior ao número de elemento das outras classes.

[9] O hiperplano, no algoritmo SVM, é responsável pela separação das classes no espaço de entrada. Idealmente, o hiperplano criado pelo algoritmo SVM é capaz de separar corretamente as classes definindo uma fronteira de decisão (MANNING; RAGHAVAN; SCHTZE, 2008).

apontada em CHRISTEN; GOISER (2007) é a dificuldade de selecionar automaticamente pares contidos na região crítica. Tais pares são importantes para um treinamento efetivo do algoritmo de classificação, devido ao alto grau de ambiguidade.

Os experimentos foram conduzidos em 6 bases de dados reais e sintéticas, variando de 1.000 a 10.000 registros. Apesar da alta qualidade (alta precisão) dos pares obtidos pelo processo automático de seleção, os experimentos demonstraram que a ausência de pares pertencentes à região crítica resulta na depreciação da eficácia do método, se comparado com as técnicas tradicionais ((FELLEGI; SUNTER, 1969), k-médias (PIERRE; MICHAUD, 1997) e com a função de distância de edição (ELMAGARMID; IPEIROTIS; VERYKIOS, 2007)).

Em CHRISTEN (2008a), é estendido o deduplicador proposto em CHRISTEN; GOISER (2007) com objetivo de aprimorar o processo de seleção automática dos pares. Foi identificado que a omissão da seleção dos pares pertencentes à região crítica resulta na depreciação acentuada da eficácia.

1. é adotada a premissa que a vizinhança em que o par se encontra é um fator determinante para a definição do rótulo dos pares. A vizinhança dos pares pode ser criada por limiares de similaridade ou pela distância Euclidiana no espaço dimensional. Assim, pares cujos vizinhos mais próximos representam duplicatas também são considerados duplicatas. Da mesma forma, os pares cujos vizinhos próximos representam não duplicatas são considerados não duplicatas. Essa estratégia é empregada recursivamente até que os conjuntos contenham um número suficiente de amostras;

2. são identificados os pares "altamente" *positivos e negativos* utilizando-se do algoritmo SVM. Mais especificamente, a partir de um treinamento inicial, o algoritmo SVM é utilizado para a predição de um conjunto de pares. Os pares são ranqueados, levando em consideração a distância até a fronteira do hiperplano criado pelo SVM. É assumido que pares mais distantes do hiperplano são mais seguros de serem definidos como duplicatas (pares positivos) ou não duplicatas (pares negativos). Incrementalmente, novas rodadas são executadas para complementar o conjunto de treinamento.

Os experimentos foram conduzidos em 7 bases de dados (3 reais e 4 sintéticas), contendo de 1.000 a 10.000 registros. Os métodos propostos foram comparados com a SVM tradicional, treinado com toda a base de dados rotulada. A experimentação comparativa do autor evidenciou que os métodos propostos obtiveram uma eficácia inferior a SVM tradicional. Segundo CHRISTEN (2008a), a deficiência na eficácia, especialmente em bases de dados reais, é devido à ausência de padrões que dificilmente são identificados pela seleção automática de pares, devido ao seu alto grau de ambiguidade.

SANTOS et al. (2008, 2011)

Em SANTOS et al. (2008, 2011), é apresentada uma abordagem para deduplicação com intuito de remover a intervenção do usuário. A intuição da abordagem é avaliar a qualidade interna dos *clusters* (agrupamentos) para extrair indícios em busca do limiar ideal. A abordagem é dividida em duas etapas: *(i)* geração dos grupos; e *(ii)* avaliação da qualidade interna dos grupos.

Na primeira etapa, são gerados grupos a partir do algoritmo de agrupamento hierárquico (HAC)(MANNING; RAGHAVAN; SCHTZE, 2008). Diferentemente de outras

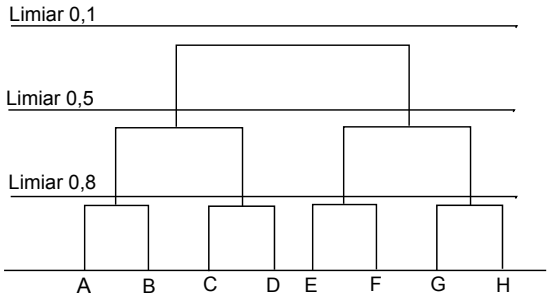


Figura 2.4: Exemplo do agrupamento hierárquico (HAC). Os valores de limiares 0,1, 0,5 e 0,8 são responsáveis pelas geração dos grupos.

abordagens de agrupamento (k-médias (PIERRE; MICHAUD, 1997)), o HAC não depende da especificação prévia do número de grupos, ou seja, evita a necessidade de se especificar valores de limiares. Como a Figura 2.4 ilustra, o HAC resulta em uma estrutura hierárquica em forma de uma árvore binária. Os grupos de registros são construídos a partir da variação do valor da similaridade (0,1, 0,2, 0,3,...,1,0). Por exemplo, todos os pares associados com uma similaridade inferior ao valor de limiar 0,1 são inseridos em um mesmo grupo. Aumentando o limiar para 0,5, a tendência é que os pares sejam mais similares e em menor número. Quando o limiar for definido com o valor 0,8, somente registros altamente semelhantes estão presentes em um mesmo bloco.

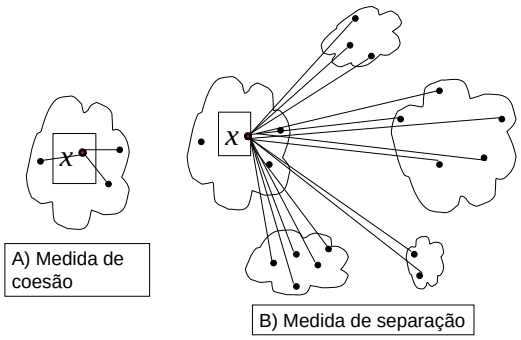


Figura 2.5: Exemplo de como os elementos são selecionados para o cálculos das *medidas de coesão (A) e separação (B)*.

Na segunda etapa, é avaliada a qualidade dos grupos produzidos por cada limiar (criados na primeira etapa) a partir das medidas de *coesão* e *separação*. A *medida de coesão* quantifica o grau de proximidade dos elementos dentro de cada grupo. Já a *medida de separação* quantifica o grau de separação entre elementos de grupos distintos. Por exemplo, um determinado grupo composto por pares com uma baixa similaridade (elementos distantes uns do outros) apresentam uma baixa coesão. Por outro lado, se a distância entre elementos de grupos diferentes for pequena (uma alta similaridade entre elementos de grupos diferentes) então os grupos apresentam uma baixa separação devido a indícios

dos grupos estarem agrupados incorretamente. As medidas de *coesão e separação* são ilustradas na Figura 2.5.

Para quantizar as distâncias entre os pares, é criada uma matriz contendo os valores de similaridade entre todos os registros da base de dados. Por fim, o *coeficiente de silhueta* combina as vantagens das medidas de coesão e separação em uma única métrica. A intuição do método é que o ponto em que o valor do *coeficiente de silhueta* for mais elevado está correlacionado com o valor do limiar ideal.

Os experimentos realizados pelos autores avaliaram o grau de correlação entre o *coeficiente de silhueta* e a métrica F1. Foram utilizadas duas bases de dados reais e quatro bases de dados sintéticas compostas por cerca de 300 a 2.000 registros. SANTOS et al. (2008, 2011) comprovaram experimentalmente que o *coeficiente de silhueta* proporciona uma alta correlação com a métrica F1, na maioria das bases testadas. No entanto, a correlação entre o *coeficiente de silhueta* e a métrica F1 foi baixa em uma das bases de dados devido à base de dados ser composta por poucos pares duplicados, o que resulta em certa instabilidade para o cálculo das medidas de *coesão* e *separação*.

2.3.2 Métodos de deduplicação que exploram a *Blocagem*

Nesta subseção, são apresentados os trabalhos que têm como foco principal a melhora na eficiência ou na redução do esforço manual da deduplicação de dados. A adição da etapa de blocagem é essencial para melhor a eficiência da deduplicação em grandes conjuntos de dados, como será observado no decorrer da subseção.

2.3.2.1 Intervenção do usuário a partir da Definição do Limiar

Vale ressaltar que alguns dos trabalhos descritos a seguir têm como foco principal possibilitar a deduplicação de grandes montantes de dados a partir do mapeamento dos registros para um conjunto de identificadores, chamados assinaturas. Tais trabalhos, coletivamente chamados de *algoritmos baseados em assinaturas*, têm como desafio a redução na geração dos pares candidatos. Nesse contexto, são adicionadas estratégias para a geração e filtragem dos pares que dependem, em algum momento, do ajuste do usuário para o seu correto funcionamento.

SARAWAGI; KIRPAL (2004) - Probe Cluster

Em SARAWAGI; KIRPAL (2004), é proposto um método *baseado em assinaturas* com o objetivo de reduzir o tempo de processamento e a demanda de memória principal para a execução da deduplicação. O método, intitulado Probe Cluster, introduz uma série de melhorias na estrutura básica do índice invertido (comumente utilizado na área de recuperação de informação (MANNING; RAGHAVAN; SCHTZE, 2008)). Uma camada, contendo as funções de similaridade *baseadas em tokens*, é adicionada ao topo da abordagem.

Uma das principais contribuições do Probe Cluster é a redução do número de elementos que devem ser percorridos na estrutura do índice invertido para a geração dos pares candidatos. Basicamente, as entradas (ou listas) criadas pelo índice invertido são ordenadas a partir da ordem crescente do número de registros que compartilham um mesmo termo. A intuição é que listas com poucos registros são criadas pelos termos menos frequentes, enquanto as listas maiores são originadas pelos termos mais frequentes. Processar termos frequentes (longas listas) pode resultar na perda da eficiência devido ao grande número de pares candidatos produzido.

Nesse cenário, é proposta a fragmentação do índice invertido em duas partes: inferior, composta por listas contendo um número reduzido de elementos; e superior, composta por termos mais frequentes (longas listas). Primeiramente, os pares são buscados na parte inferior do índice para que, posteriormente, sejam executas buscas binárias nas listas mais longas para verificar a similaridade de cada par. A intuição é que pares não são exclusivamente compostos por elementos muito frequentes e devem conter termos, em comum, na parte inferior do índice. De fato, a geração de pares candidatos a partir dos termos frequentes resulta no aumento do custo computacional sem uma melhora substancial na qualidade dos pares (ou seja, um aumento superficial na geração dos pares candidatos duplicados). Dessa forma, é reduzido o número de buscas e o tempo de processamento para a construção do conjunto de pares candidatos.

Adicionalmente, foram propostas pequenas otimizações no método, como a remoção de termos frequentes (*stop words*), compactação do índice para execução na memória principal, entre outras. Um dos desafios do método Probe Cluster é identificar o correto valor do limiar que divide o índice invertido em duas partes. Uma definição imprecisa de tal limiar pode inserir termos frequentes na parte inferior do índice, resultando em uma baixa eficiência do método.

Os experimentos foram executados nas bases de dados reais DBLP[10] e CiteSeer[11] com um montante de 250 mil e 500 mil registros. A avaliação experimental demonstrou que as otimizações propostas reduziram o tempo de processamento em até duas ordens de magnitude. A utilização de índices invertidos implica em um modelo de blocagem, permitindo que a experimentação seja conduzida em bases de dados compostas por centenas de milhares de registros.

CHAUDHURI; GANTI; KAUSHIK (2006) - SSJoin

A abordagem proposta por CHAUDHURI; GANTI; KAUSHIK (2006) tem como objetivo explorar a eficiência da deduplicação através da combinação de operadores relacionais primitivos. O método, chamado de SSJoin, estende a deduplicação para o âmbito da interseção de conjuntos, promovendo o casamento exato entre elementos. Diferentemente de Probe Cluster, que adiciona uma camada contendo as funções de similaridade, SSJoin explora as particularidades de cada função de similaridade para maximizar a eficiência.

A principal contribuição da abordagem SSJoin é a formalização do conceito de *filtro de prefixo*. O *filtro de prefixo* tem como objetivo reduzir o número de pares candidatos, evitando executar o cálculo da similaridade em pares com uma baixa probabilidade de representarem uma duplicata. A intuição do *filtro de prefixo* é que se dois registros são similares, então seus prefixos, quando ordenados, devem compartilhar no mínimo um *token* em comum. O *filtro de prefixo* possibilita que apenas uma pequena parte de cada registro (ou seja, o prefixo do registro) seja processada para a construção dos pares candidatos, reduzindo custos de processamento. O *filtro de prefixo* é formalizado a seguir.

Definição 1. *(Filtro de Prefixo): considere um ordenamento global O de um conjunto de tokens U. O filtro de prefixo define que se um par de registros X e Y apresentar um valor de similaridade maior que um limiar α, então a interseção dos (|X ou Y|- α+1) primeiros tokens dos registros deve conter, no mínimo, um token em comum.*

Primeiramente, é computada a frequência global dos *tokens* para a identificação dos

[10]DBLP: http://www.informatik.uni-trier.de/ ley/db/
[11]CiteSeer: http://citeseer.ist.psu.edu/

termos menos frequentes de cada registro, ou seja, o reordenamento dos registros, a partir da frequência global, permite identificar os *tokens* considerados relevantes para a geração dos pares candidatos. Considere, por exemplo, os quatro conjuntos (*x, y, z, w*) ilustrados na Figura 2.6. Cada registro do exemplo está ordenado de acordo com a frequência global dos termos. Considerando que β represente o limiar de similaridade, o tamanho de prefixo de cada registro é formado pelos *(|x|-|x|.β +1)* primeiros elementos. Definindo o valor de β igual a 0,8, no máximo dois termos são utilizados para a geração dos pares candidatos (ou seja, são gerados três pares candidatos [(z,y), (y,x), (w,x)]). Note-se que o produto cartesiano entre os registros resulta em um total de seis pares candidatos (cerca de duas vezes mais pares candidatos em relação ao filtro de prefixo). Por fim, aplicando a função similaridade *Jaccard*[12] com o limiar de similaridade β, somente o par [(z, w)] é considerado uma duplicata (similaridade igual a 0,8 (8/10)).

x=	[C,	D,	F]		
y=	[G,	A,	B	E,	F]
z=	[A,	B,	C,	D	E]
w=	[B,	C,	D,	E,	F]

Figura 2.6: Exemplo de registros, no qual os elementos realçados representam o prefixo de cada registro.

A avaliação experimental foi conduzida em uma base de dados real composta por 25 mil registros. SSJoin foi comparado com a abordagem GRAVANO et al. (2001). Na experimentação, utilizando diversas funções de similaridade (por exemplo, Jaccard e distância de edição) foi constatado que o *filtro de prefixo* foi capaz de melhorar a eficiência da deduplicação, reduzindo o número de pares candidatos em ordens de magnitude em relação ao *baseline*.

ARASU; GANTI; KAUSHIK (2006) - PartEnum

Um novo deduplicador *baseado em assinaturas* é proposto por ARASU; GANTI; KAUSHIK (2006) com objetivo de melhorar a eficiência no casamento exato de pares. O deduplicador, chamado PartEnum, tem como contribuição um novo método para a geração de assinaturas dos pares candidatos.

O PartEnum é dividido em duas etapas: partição e enumeração. Na etapa de partição, o registro é mapeado para um vetor binário e dividido, primeiramente, em n_1 faixas. Em seguida, cada faixa é subdividida em n_2 subfaixas, gerando um total de n_1 x n_2 subfaixas. Na etapa de enumeração, cada subfaixa é combinada com todas as outras subfaixas do registro para a geração das assinaturas. A Figura 2.7 ilustra a geração das assinaturas do vetor binário "010110". Primeiramente, o vetor é dividido pela metade (n_1=2) e cada metade é segmentada em três outras subfaixas (n_2=3). As subfaixas são combinadas para a geração de seis assinaturas. Dessa forma, o algoritmo PartEnum possibilita que pequenos erros de tipografia ou variações não impossibilitem a geração correta dos pares candidatos.

Uma importante observação de ARASU; GANTI; KAUSHIK (2006) é que pares que apresentam uma diferença de tamanho (número total de assinaturas) acima de um va-

[12] A função de similaridade *Jaccard* calcula a similaridade a partir da divisão da interseção pela união dos conjuntos (ELMAGARMID; IPEIROTIS; VERYKIOS, 2007).

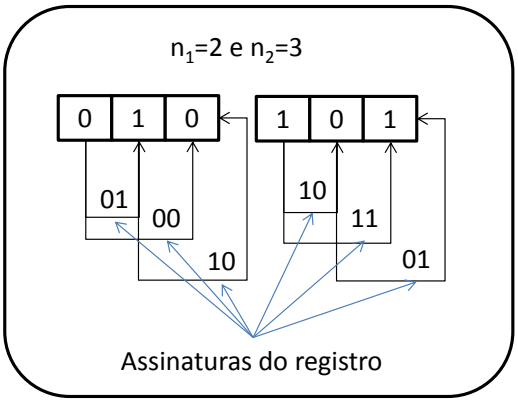


Figura 2.7: Exemplificação das assinaturas geradas por um registro ao definir os limiares com tamanho $n_1 = 2$ e $n_2 = 3$. Os registros que compartilharem assinaturas em comum são agrupados em um mesmo bloco para a geração dos pares.

lor de limiar, possuem uma baixa probabilidade de representarem uma duplicata. Dessa forma, com objetivo de remover pares candidatos dissimilares, foi proposta a Definição 2, chamada de *filtro de tamanho*, formalizada a seguir.

Definição 2. *(Filtro de Tamanho): se a similaridade entre o par X e Y for maior que o valor de limiar γ, segundo a função de similaridade Jaccard, então o par deve possuir uma variação de tamanho entre :* $1/\gamma \geq |X|/|Y| \geq \gamma$ *(ARASU; GANTI; KAUSHIK, 2006).*

Os experimentos foram executados em bases de dados reais e sintéticas, variando de 50 mil até 1 milhão de registros. A abordagem PartEnum foi comparada com a proposta e um conhecido algoritmo para agrupamento que implementa tabelas de dispersões ou *hash tables* (SCHROEDER; GODDARD; RAMAMURTHY, 2000). A experimentação demonstrou que o PartEnum foi capaz de reduzir substancialmente o tempo de processamento (cerca de 10 a 70 vezes mais eficiente). Entretanto, um fator determinante para o bom desempenho da abordagem PartEnum é a especificação manual de bons valores de limiares n_1 e n_2 para o processo de geração das assinaturas.

BAYARDO; MA; SRIKANT (2007) - All-Pairs

BAYARDO; MA; SRIKANT (2007) apresentam uma abordagem, chamada All-Pairs, cujo intuito é maximizar a eficiência e a escalabilidade das abordagens para a deduplicação *baseadas em assinaturas*. All-Pairs propõe que a filtragem de pares candidatos seja realizada utilizando somente a similaridade dos elementos contidos nos prefixos dos pares, semelhante ao *filtro de prefixo*. Os prefixos que apresentarem uma similaridade abaixo do valor de limiar são removidos do conjunto de pares candidatos. Dessa forma, é evitado que parte do registro (sufixo do registro) seja desnecessariamente processada. Além disso, All-Pairs utiliza a *Definição 2*, intitulada *filtro de tamanho*, para remover pares que apresentarem variações no número de elementos acima de um limiar, previamente definido.

All-pairs propõe melhorias ao *filtro de prefixo*, como, por exemplo: implementação autônoma (originalmente é implementado a partir de bancos de dados relacionais); e a utilização de tabelas de dispersões (ou *hash table*) para reduzir o custo de manipulação

dos dados (ao invés de índices invertidos). Apesar das melhorias propostas, conceitualmente não são feitas diferenciações das *Definição 1 e 2* do *filtro de prefixo e tamanho*, previamente propostas.

A experimentação foi conduzida avaliando o desempenho de All-Pairs utilizando como *baseline* as técnicas PartEnum, S&K e LSH (HAGHANI; MICHEL; ABERER, 2009). Uma importante contribuição de All-Pairs é avaliação experimental em volumes substanciais de dados com mais de cinco milhões de registros em três domínios distintos (log de consultas[13], redes sociais[14] e na DBLP[15]). Os resultados obtidos pelo All-Pairs demonstraram ser 5 a 22 vezes mais eficiente que S&K e 10 até 700 vezes mais eficiente que PartEnum. Não foram conduzidos experimentos comparando com a abordagem SSJoin, cujo conceito de *filtro de prefixo* está relacionado.

XIAO et al. (2008, 2011) - PPJoin & PPJoin+

XIAO et al. (2008, 2011), observam empiricamente que nas abordagens All-Pairs e PartEnum o número de pares candidatos produzido cresce quadraticamente em relação ao conjunto de entrada. Nesse contexto, são propostas duas abordagens, chamadas PPJoin e PPJoin+, com intuito de minimizar o número de pares candidatos.

O algoritmo PPJoin combina os *filtros de prefixo e tamanho* com o *filtro de posição*. A intuição do *filtro de posição* é avaliar os pares que sobreviveram ao processo de filtragem, podando os pares falsos positivos (pares não duplicados presentes no conjunto de pares candidatos) remanescentes. A poda é feita levando em consideração a posição dos *tokens* que compõem o prefixo dos conjuntos. Caso a diferença de posição entre os elementos esteja acima de um valor de limiar (definido pelo usuário), então o par candidato é descartado. O *filtro de posição* é definido pela Equação 2.3.

$$Fpos = 1 + min(X_j - |X|, Y_j - |Y|) \tag{2.3}$$

onde:

- |X| e |Y| representam os respectivos tamanhos dos registros;
- X_j representa a posição do *token j* no registros X;
- Y_j representa a posição do *token j* no registros Y;

A Figura 2.8 ilustra um par de registros com uma similaridade relativamente baixa (seis elementos em comum de um total de dez elementos, similaridade de 60%) quando definida uma similaridade mínima de 90%. No entanto, o filtro de prefixo não é efetivo na remoção de tal par devido à presença do elemento "B" em ambos os prefixos. Nesse contexto, o *filtro de posição* complementa o *filtro de prefixo* analisando a posição relativa de cada elemento no prefixo dos registros. Por exemplo, o elemento "B" da mesma figura está representado na primeira posição do registro *X* e na segunda posição do registro *Y*. Ao aplicar a Equação 2.3 (Fpos=1+ min(5-2, 5-4)=4), é obtido um valor abaixo do limiar (definido pela equação $\lceil (Limiar * \lceil |x|y| \rceil) \rceil$), portanto o par deve ser removido do conjunto de pares candidatos.

[13]http://www.google.com
[14]http://www.orkut.com.br
[15]http://www.informatik.uni-trier.de/ ley/db/

x = [**B**, C, D, L, F]

y = [A, **B**, C, D, E]

Figura 2.8: Exemplo de par descartado do conjunto de pares candidatos pelo *filtro de posição*.

O segundo algoritmo proposto, chamado de PPJoin+, combina os *filtros de tamanho, prefixo, posição* e adiciona o *filtro de sufixo*. O *filtro de sufixo* tem como objetivo analisar a posição relativa dos *tokens* nos sufixos de cada conjunto. Tal filtro é capaz de computar a posição relativa dos *tokens*, mesmo que os registros não estejam totalmente indexados (reduzindo atrasos no processamento). A implementação do filtro é feita a partir da estratégia de divisão e conquista, na qual um primeiro *token* *w* é aleatoriamente escolhido para divisão do registro *X* em duas partes (x_r e x_1). Da mesma forma, *w* é buscado no conjunto *Y*, dividindo-o em duas partes (y_r e y_l). A soma das diferenças das posições em (x_r e y_r) e (x_l e y_l) resulta na similaridade de posição do par. A estratégia de divisão e conquista prossegue, de forma recursiva, até atingir uma profundidade máxima ou o par apresentar um valor de similaridade abaixo do limiar definido pelo usuário.

Os algoritmos PPJoin e PPJoin+ foram avaliados em diferentes domínios, como bibliotecas digitais, e-mails, coleções de artigos, sites da Web e comparados com a técnica All-Pairs. Os algoritmos PPJoin e PPJoin+ comprovaram ser até cinco vezes mais rápidos que All-Pairs. Segundo Xiao et al., os bons resultados dos experimentos se devem ao crescimento linear do conjunto de pares candidatos.

AWEKAR; SAMATOVA; BREIMYER (2009)

Em AWEKAR; SAMATOVA; BREIMYER (2009), é proposta uma estratégia incremental para a eliminação na redundância de processamento na deduplicação de dados. É observado que apesar do estado da arte da deduplicação oferecer diversas técnicas explorando a eficiência dos métodos *baseados em assinaturas* (All-Pairs, PPJoin, PartEnum, PPJoin+). No entanto, tais trabalhos estão atrelados a limiares manualmente especificados. A identificação manual de valores de limiares é uma tarefa dispendiosa devido à necessidade de realizar sucessivas execuções em busca da configuração ideal. As execuções resultam na redundância de processamento, visto que o mesmo par pode ser processado inúmeras vezes.

Com objetivo de reduzir a redundância de processamento, é proposto o armazenamento de históricos de execução (*logs*) para evitar o reprocessamento de pares. Como deduplicador, é utilizado o algoritmo baseado em assinaturas All-Pairs. Mais especificamente, o histórico dos pares já processados é armazenado em dois arquivos. O primeiro arquivo armazena os pares duplicados que apresentarem uma similaridade acima do limiar *t*. O segundo arquivo armazena os pares com valor de similaridade abaixo do limiar *t*, ou seja, os possíveis pares duplicados. Um limiar de poda (*simFloor*) é requisitado para definir o valor mínimo de similaridade que os pares devem atingir para serem armazenados no *arquivo de possíveis duplicatas*. Em uma segunda execução do deduplicador, caso o novo valor do limiar *t_new* (definido pelo usuário) for menor que o valor do limiar previamente definido *t_old*, então o arquivo de histórico *t_old* representa um subconjunto de *t_new*. Dessa forma, somente os pares com similaridade de prefixo entre os valores de *t_new* e *t_old* são computados. Esse modelo evita uma redundância de processamento dos

pares já computados nas etapas anteriores. Caso, o limiar *t_new* seja maior que o limiar *t_old*, então somente é necessário recuperar o histórico das execuções anteriores.

Os experimentos foram conduzidos em quatro bases de dados reais variando de 1,5 a 3 milhões de registros. A análise experimental concluiu que a estratégia proposta resultou em uma melhora no tempo de processamento de 2 a 10^5 vezes, se comparado com a abordagem original All-Pairs. Tal melhora é resultado da remoção da redundância entre as sucessivas execuções. Não é discutido abordagens para identificar o valor de limiar ideal, ou seja, a identificação do valor de limiar é deixada a cargo do usuário.

VERNICA; CAREY; LI (2010)

VERNICA; CAREY; LI (2010) propõem a extensão do algoritmo *baseado em assinaturas* PPJoin+ para o processamento em larga escala. O algoritmo utiliza o modelo de programação distribuído *MapReduce* para a paralelização de tarefas (DEAN; GHEMAWAT, 2008). No *MapReduce*, o usuário é responsável pela especificação das funções de mapeamento (*Map*) e redução (*Reduce*), enquanto o modelo de programação se encarrega de paralelizar e distribuir as tarefas sem a intervenção do usuário.

O método proposto é divido em três principais estágios. No primeiro estágio, é criado um ordenamento global dos *tokens* para a definição das assinaturas de cada registro. Cada *token* é substituído por um valor de identificação de acordo com a sua posição no *ranking* de frequência. No segundo estágio, são aplicados os filtros (*prefixo, tamanho, posição e sufixo*) sobre as assinaturas. Dessa forma, são criados conjuntos de pares candidatos. Caso o valor de similaridade do par candidato esteja acima do valor do limiar, previamente definido, então o par é repassado para o estágio seguinte. Por fim, no estágio três, os pares de registros são reconstruídos (os valores numéricos são substituídos pelos respectivos *tokens*) para o salvamento no arquivo de saída.

Os experimentos foram executados na base de dados do DBLP e CITESEERX. Para avaliar a escalabilidade do método, as bases de dados de dados foram replicadas de 5 a 25 vezes do seu tamanho original. A experimentação dos autores demonstrou que o método foi capaz de processar uma base de dados com mais de 30 milhões de registros.

DAL BIANCO; GALANTE; HEUSER (2011) - MD-Aproach

DAL BIANCO; GALANTE; HEUSER (2011) apresentam um deduplicador que combina um eficiente método de blocagem com o modelo de programação *MapReduce*, com intuito de melhorar o desempenho e a qualidade da deduplicação em arquiteturas multiprocessadas. Na abordagem chamada MD-Approach, é proposto um deduplicador que emprega um método de blocagem em duas etapas para evitar o desbalanceamento de carga, ou seja, blocos substancialmente grandes podem consumir recursos (processadores) por um longo período mesmo que os demais blocos tenham sido processados. No MD-Approach, os pares candidatos são criados através da combinação de atributos (ou partes de atributos) para a construção das *chaves de blocagem*[16].

A primeira etapa de blocagem tem como objetivo a criação de blocos genéricos para assegurar que registros duplicados não sejam erroneamente removidos do bloco correspondente. Isso significa que tais blocos utilizam chaves de blocagem genéricas (por exemplo, a primeira letra dos valores de um atributo) e podem conter um número acentuado de registros. Os blocos que contiverem um número de registros acima de um valor de limiar, previamente definido, são enviados para a segunda etapa de blocagem. Na segunda

[16] As *chaves de blocagem* representam um valor de atributo (ou parte de um atributo). Todos os registros que compartilham um mesmo valor de chave são inseridos em um mesmo bloco.

etapa, os blocos são fragmentados em sub-blocos através da definição de chaves de blocagem mais especializadas (por exemplo, três primeiras letras dos valores de um atributo). Dessa forma, a blocagem em duas etapas fragmenta a base de dados em pequenos blocos minimizando o custo de processamento.

A blocagem em duas etapas é combinada com o modelo de programação paralela *MapReduce* para explorar o poder computacional das atuais arquiteturas multiprocessadas. O deduplicador é dividido em *quatro etapas*, como ilustrado na Figura 2.9. Na *primeira etapa* são utilizadas chaves de blocagem genéricas sobre o conjunto de registros da base de dados. Um conjunto de funções de mapeamento (*Map*) é utilizado para paralelizar o processo de geração das chaves. Todos os registros que apresentarem a mesma chave são enviados para um mesmo bloco, utilizando a arquitetura MapReduce para distribuição dos dados. Na *segunda fase*, as funções de redução (*Reduce*) recebem os registros correspondentes de cada bloco. Em seguida, os blocos identificados como balanceados (ou seja, contiverem um número de registros menor que o valor de limiar k) são diretamente processados pela função de similaridade, previamente definida. Os blocos considerados desbalanceados (*unbalanced block*) são enviados para a *terceira fase*. Um novo processo de mapeamento e redução é utilizado para promover uma fragmentação mais especializada desses blocos. Na *quarta fase*, os registros, identificados como duplicados, são agrupados e enviados para o arquivo de saída.

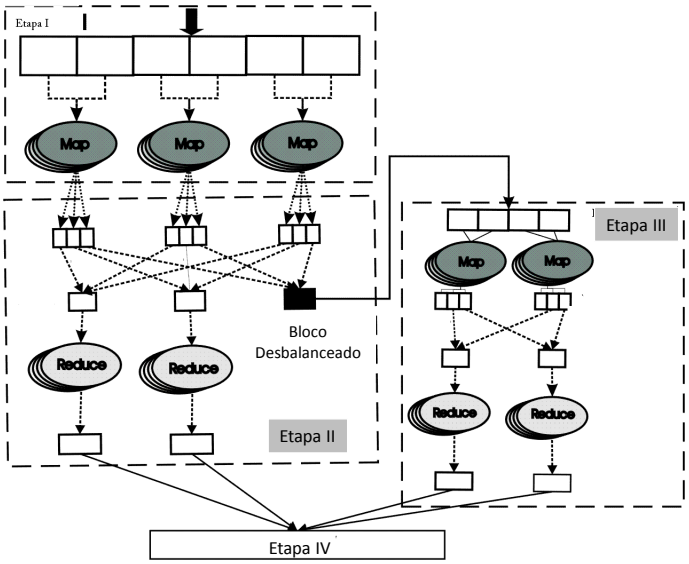


Figura 2.9: Visão geral da abordagem MD-Approach.

Os experimentos tiveram como objetivo avaliar a eficácia e a eficiência da abordagem MD-Approach comparada com VERNICA; CAREY; LI (2010). Os experimentos foram executados com dados sintéticos variando de um a quatro milhões de registros. Os experimentos demostraram que o MD-Approach foi até duas vezes mais eficiente que o *baseline*, com uma melhora de mais de 9% no valor do F1. Entretanto, a eficiência da abordagem MD-Approach depende da definição adequada dos valores de limiares.

WANG; LI; FE (2011) - Fast-Join

Em WANG; LI; FE (2011), é projetado um deduplicador, chamado de Fast-Join, com objetivo de melhorar a eficácia e a eficiência dos métodos *baseados em assinaturas*.

No Fast-Join, é proposta uma nova função de similaridade híbrida, na qual combina vantagens das funções de similaridade baseadas em *tokens* com funções baseadas em *caracteres*. Em uma primeira fase, é utilizada uma função de similaridade *baseada em tokens* com intuito de remover pares com baixa probabilidade de representarem uma duplicata. Na segunda fase, os pares sobreviventes são processados por uma função de similaridade *baseada em caracteres* capaz de identificar pequenas variações.

A função de similaridade híbrida agrega um acentuado custo computacional, visto que o cálculo de similaridade em *nível de caracteres* é executado entre todos os *tokens* dos pares candidatos. É proposto um aperfeiçoamento dos algoritmos baseados em assinaturas para reduzir a geração de pares candidatos. No Fast-Join, as assinaturas são representadas por um *token* juntamente com seu identificador de origem. Esse modelo de assinatura permite identificar e diferenciar quais *tokens* produziram uma determinada assinatura. Tradicionalmente, o *filtro de prefixo* remove as assinaturas mais frequentes, em relação ao conjunto de *tokens*. No entanto, no Fast-Join a remoção das assinaturas é feita em nível de *tokens*, ou seja, são removidos os *tokens* que correspondem às assinaturas mais frequentes. O objetivo é evitar a remoção de *tokens* frequente, mas relevante para a identificação de duplicatas. Dessa forma, evita-se que assinaturas oriundas de termos raros sejam perdidas.

Os experimentos avaliaram a eficácia e a eficiência do método. O método foi avaliado em bases de dados reais oriundas do UOL e da DBLP, contendo 600.000 e um milhão de registros, respectivamente. Na experimentação, foi concluído que o método de geração de assinatura do Fast-Join foi capaz de remover mais pares candidatos que o *filtro de prefixo*, resultando em um tempo de execução três até cinco vezes menor. A função de similaridade proposta no Fast-Join obteve uma melhora na eficácia, se comparado com as funções de similaridade tradicionais (como, por exemplo, a função *Jaccard*). Uma das desvantagens do Fast-Join é o aumento no número de limiares que devem ser definidos pelo usuário para se atingir o desempenho esperado.

2.3.2.2 Intervenção do usuário a partir da definição de limiares e da rotulação de pares

BILENKO; MOONEY (2003) - MARLIN

BILENKO; MOONEY (2003) projetaram um *framework* com intuito de maximizar a eficácia da deduplicação utilizando basicamente funções de similaridade adaptáveis ao domínio. A abordagem, chamada MARLIN (*Multiply Adaptive Record Linkage using INduction*), utiliza de uma amostra, previamente rotulada, para o treinamento em duas camadas: *(i) nível de atributo*; e *(ii) nível de registro*.

Na primeira camada, *nível de atributo*, é proposta a utilização de uma variante da função de distância de edição, na qual são adicionados pesos a substituições, inserções e remoções, utilizando o algoritmo EM (*Expectation- Maximization*)(DEMPSTER; LAIRD; RUBIN, 1977). O intuito é adaptar o cálculo de similaridade de acordo com o ruído presente na base de dados, ou seja, identificar quais são os tipos de erros ou variações responsáveis pela geração das duplicatas para facilitar o cálculo da similaridade. Como ilustrado na Figura 2.10, MARLIN possibilita que o vetor de características seja composto por diferentes combinações de funções de similaridade aplicadas a um determinado atributo. Assim, o vetor de características é capaz de identificar padrões complementares que poderiam ser perdidos caso somente uma função de similaridade fosse aplicada.

Na segunda camada, nível de registro, o algoritmo SVM é treinado com os vetores

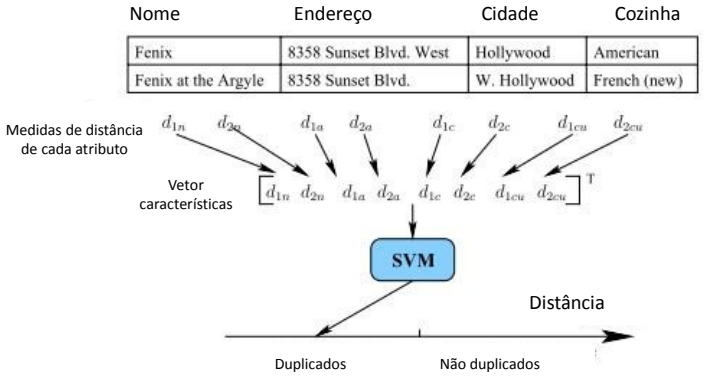


Figura 2.10: Visão geral da abordagem MARLIN (BILENKO; MOONEY, 2003).

de características. Como resultado, é produzido um modelo de treinamento que é responsável pela quantização do grau de confiança do classificador, ou seja, a distância do par à fronteira do *hiperplano*. Por fim, o grau de confiança é avaliado por um classificador binário.

MARLIN emprega o método de clusterização *Canopy* (MCCALLUM; NIGAM; UNGAR, 2000) para promover um processo de blocagem em duas etapas. Na primeira etapa, é utilizada a bem conhecida função de similaridade Jaccard (KOUDAS; SARAWAGI; SRIVASTAVA, 2006), com baixo custo computacional. Na segunda etapa, é utilizada uma estrutura básica de índice invertido (MANNING; RAGHAVAN; SCHTZE, 2008) para o agrupamento de registros.

Os experimentos propostos tiveram como objetivo avaliar a eficácia da abordagem, assim como, a adaptação alcançada pelo processo de aprendizagem. A abordagem não discute a parametrização da blocagem. MARLIN apresentou uma melhora na eficácia comparada aos métodos tradicionais (puros) de treinamento (por exemplo, SVM). A avaliação foi feita em bases de dados reais com cerca de 800 a 1.200 registros. Apesar de sugerir a possibilidade de adoção de algoritmos de *aprendizagem ativa* para a criação da base de treinamento, nenhum experimento foi conduzido com intuito de avaliar o número de pares necessários para a convergência do processo de treinamento.

KöPCKE; RAHM (2010)

KöPCKE; RAHM (2010) projetam um *framework* genérico com objetivo de oferecer uma plataforma para a avaliação de técnicas de deduplicação. A principal contribuição é a proposta de dois métodos para a seleção automática dos pares candidatos. Em tais métodos, o usuário é requisitado para validar um conjunto reduzido de pares que estiverem acima de um valor de limiar de poda, definido pelo usuário.

No primeiro método para seleção dos pares candidatos, são selecionados N pares candidatos que, quando aplicados a função de similaridade *t*, resultam em um grau de similaridade superior ao limiar *m*. O objetivo é descartar os pares pouco relevantes para o ganho de informatividade do conjunto de treinamento. No segundo método, são selecionados N/2 pares candidatos que, quando aplicados à função de similaridade *t*, resultam em um grau de similaridade superior ao limiar *m*. Adicionalmente, são selecionados os N/2 pares que, quando aplicados à função de similaridade *t*, resultam em uma similaridade abaixo do valor de limiar *m*. O principal objetivo do segundo método é produzir uma

amostra balanceada de pares duplicados e não duplicados. Ambos os métodos convergem quando é atingido um tamanho de amostra igual a *N*.

O principal objetivo é de automatizar o processo de seleção de amostras, entretanto, os métodos propostos são dependentes da definição manual das variáveis *m*, *t* e *n*. Na experimentação, é promovida uma avaliação empírica buscando comparar os valores de limiares e diferentes tamanhos de amostras. O método *limiar-igual* demonstrou ser menos dependente do conjunto de treinamento e do valor de limiar. Os experimentos foram executados em quatro bases de dados reais variando de cerca 600 a 66.000 registros. K&R não apresentaram uma análise sobre os parâmetros utilizados para o processo de blocagem e nem para identificar o valor do limiar de poda.

ARASU; GOTZ; KAUSHIK (2010) - AG&K

Segundo ARASU; GOTZ; KAUSHIK (2010), os métodos que utilizam *aprendizagem ativa* (MARLIN e Alias) têm como desvantagem a ausência de controle sobre a qualidade dos pares (ou seja, uma precisão mínima) e, originalmente, são restritos a pequenos conjuntos de dados. Nesse contexto, é proposto um método ativo para deduplicação, chamado aqui de AG&K, no qual o usuário define um valor mínimo de precisão e o método se encarrega de obter a revocação próxima ao ideal.

AG&K propõe o mapeamento de funções de similaridade para um espaço N-dimensional. A intuição do algoritmo é criar uma fronteira no espaço dimensional que seja capaz de separar os pares duplicados dos pares não duplicados utilizando de um processo ativo de seleção e rotulação manual de pares. Observe que na figura, os pares são agrupados em quadrantes com intuito de facilitar a seleção dos pares a serem manualmente rotulados. Mais ainda, é proposta uma busca binária sobre os quadrantes com intuito de reduzir o número de pares rotulados.

A precisão mínima (definida pelo usuário) e a revocação são computadas a partir de um *oráculo* composto por uma conjunção de funções de similaridade e limiares, definidos manualmente. O principal objetivo do *oráculo* é identificar os pares mais ambíguos (pares mais difíceis de serem definidos como duplicata ou não duplicata) para serem rotulados ativamente pelo usuário.

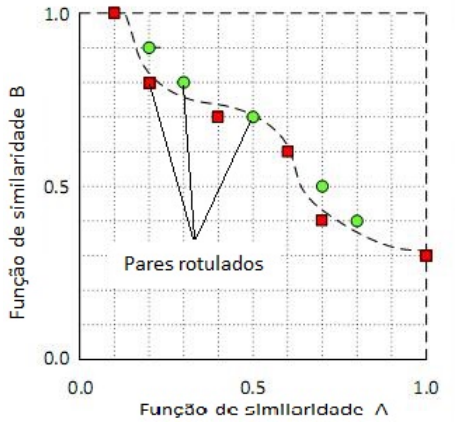


Figura 2.11: Linha de classificação formada pelos pares rotulados pelo usuário no espaço N-dimensional.

Na experimentação, são reimplementadas duas abordagens que exploram a aprendizagem ativa (MARLIN e Alias). Originalmente, MARLIN e Alias não incorporam técnicas

de blocagem, o que restringe sua aplicação a pequenos conjuntos de dados. Assim, na reimplementação de AG&K é adicionado uma etapa blocagem para permitir uma comparação direta com o método proposto. Segundo AG&K, a adição da blocagem resultou em um comportamento inconsistente em relação à qualidade da deduplicação. Mais especificamente, os comitês de classificadores utilizados por MARLIN e Alias não foram capazes de selecionar um conjunto representativo de pares. AG&K contornam tal problema complementando manualmente o conjunto de treinamento.

Os experimentos foram conduzidos em duas bases de dados reais com mais de um milhão de registros. A intervenção do usuário é determinada em dois níveis: (*i*) a partir da definição das conjunções de funções de similaridade e dos valores de limiares; e (*ii*) a partir da rotulação manual de pares. Os autores observam que as abordagens MARLIN e Alias se tornam instáveis à medida que novos pares são adicionados ao conjunto de treinamento. Já o método proposto por AG&K, é capaz de selecionar ativamente pares que acrescentam ganho de informatividade.

BELLARE et al. (2012) - ALD

Em BELLARE et al. (2012), é proposta uma estratégia, chamada de ALD, para o mapeamento de métodos genéricos de aprendizagem ativa para o contexto da deduplicação de dados. É observado que os métodos tradicionais de aprendizagem ativa têm como objetivo melhorar a acurácia[17] da classificação. No entanto, no contexto da deduplicação, a acurácia pode quantificar erroneamente a qualidade da classificação que depende de somente dos pares identificados como duplicatas para a quantização eficácia.

A intuição da abordagem ALD é que, a partir de um conjunto reduzido de pares ativamente rotulados pelo usuário, é possível identificar o classificador que maximize a revocação, respeitando um valor de precisão mínima definido pelo usuário. ALD pressupõe a existência de uma abordagem de aprendizagem ativa para a seleção dos pares mais informativos. Dessa forma, foi adotado o método de seleção ativa chamado IWAL (*Importance Weighted Active Learning*) devido a sua baixa demanda de rotulação manual (detalhes do formalismo podem ser encontradas em BALCAN; BEYGELZIMER; LANGFORD (2009)). No IWAL, os pares são rotulados a partir da divergência entre o classificador com menor "erro" (ou seja, o classificador que minimiza a função de perda (*loss function*)) e um classificador alternativo (novamente com um reduzido erro, mas com uma incorreta predição). O classificador é considerado ótimo quando é alcançada uma taxa de erro mínima, manualmente definida.

A abordagem ALD basicamente invoca o método de aprendizagem ativa IWAL para a seleção dos pares mais informativos para serem rotulados manualmente. Um oráculo, armazenando os pares manualmente rotulados, é utilizado para estimar a qualidade de cada classificador (precisão e revocação). Uma projeção da qualidade dos classificadores é representada no espaço bidimensional para inferir informações e comparar a eficácia de cada classificador. A Figura 2.12 ilustra o espaço bidimensional, no qual os classificadores estão representados. Os pontos na figura representam os classificadores e os eixos X(h) e Y(h) representam funções derivadas das métricas de revocação e precisão. Adicionalmente, o marco "0" no eixo Y(h) representa o limiar de precisão mínimo, ou seja, todos os classificadores, abaixo de tal limiar, são descartados. Para identificar o classificador com uma máxima qualidade, é utilizada uma busca binária sobre o espaço bidimensio-

[17] A acurácia mensura a qualidade da classificação baseada no número total de pares (positivos e negativos) que são corretamente classificados.

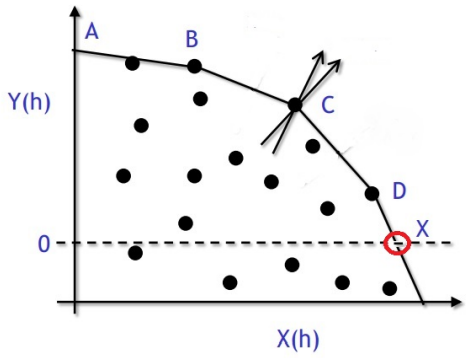


Figura 2.12: Espaço bidimensional utilizado para identificar o classificador ideal.

nal. Observe que na figura, um classificador posicionado no ponto "X" alcança um valor máximo da função X(h), no entanto, nenhum classificador foi gerado em tal ponto. O classificador mais próximo do local ideal é representando pelo ponto "D". Dessa forma, ALD é capaz de identificar o classificador com uma alta qualidade, respeitando a precisão mínima imposta pelo usuário.

Os experimentos realizados tiveram como objetivo avaliar o esforço do usuário, ou seja, quantos pares foram manualmente rotulados para a convergência do método. ALD utiliza bases reais com até 60.000 registros. A avaliação experimental comprovou que ALD foi capaz de reduzir substancialmente o custo da rotulação com uma qualidade superior ao método proposto por AG&K. No entanto, ALD depende da definição do tamanho da amostra de treinamento a ser rotulada, utilizando um limiar definido pelo usuário. Além disso, é utilizada uma conjunção de funções de similaridade e valores de limiares, definidos manualmente, para configurar a etapa de blocagem.

2.3.3 Outras propostas

Como a bibliografia relacionada ao problema da deduplicação é vasta, alguns trabalhos relacionados à tese são sucintamente apresentados a seguir. Levantamentos bibliográficos endereçando as etapas da deduplicação, podem ser encontrados em diversos trabalhos, tais como ELMAGARMID; IPEIROTIS; VERYKIOS (2007); KöPCKE; RAHM (2010); CHRISTEN (2011); DORNELES; GONçALVES; SANTOS MELLO (2011).

- CHAUDHURI et al. (2003) projetam um deduplicador com objetivo de estender funções de similaridade baseada em caracteres para o nível de *tokens*. A ordem inversa da frequência de cada documento (IDF) é computada para avaliar os pesos de cada *token*, similarmente à abordagem *Fast-Join*. No entanto, em CHAUDHURI et al. (2003) é computado a similaridade entre *tokens* buscando o elemento mais próximo, ignorando a posição de cada *token*. O casamento com o *token* mais próximo não garante a propriedade da simetria (ou seja, o valor de similaridade de <a,b> deve ser igual ao valor de similaridade de <b,a>), podendo levar a resultados inconsistentes. Tal problema é abordado na abordagem *Fast-Join*;

- Em BILENKO; KAMATH; MOONEY (2006), a proposta é automatizar especificamente a etapa de blocagem. Para isso, são utilizados métodos de aprendizagem supervisionada para identificar as *funções ou chaves de blocagem*. No entanto, tal método depende de um conjunto de treinamento representativo, produzido pelo

usuário, para a identificação das funções de blocagem ótimas. Nos levantamentos bibliográficos de WINKLER (2005); BAXTER; CHRISTEN; CHURCHES (2003); CHRISTEN (2007) podem ser encontrados estudos de diferentes técnicas de blocagem;

- DORNELES et al. (2007, 2009) propõem um novo escore com objetivo de permitir a configuração da deduplicação a partir da definição de um valor de precisão. O método parte do princípio de que não existe uma função de similaridade globalmente ótima e o usuário raramente é capaz de definir um efetivo valor de limiar. Nesse contexto, é proposto um novo escore, chamado de *escore ajustado*, capaz de construir um mapeamento entre o valor de precisão, definido pelo usuário, para os valores de limiares utilizados internamente pelas funções de similaridade. Para possibilitar tal mapeamento é utilizado um conjunto de treinamento manualmente produzido. Na experimentação foi constatado que é possível identificar o valor de um *escore ajustado* único para cada domínio (por exemplo, um *escore ajustado* único para os atributos contendo nomes de pessoas em diferentes bases de dados). DORNELES et al. (2007, 2009) não discutem estratégias para a blocagem dos pares, restringindo os experimentos a pequenos conjuntos de dados.

- KöPCKE; RAHM (2010) elaboraram os primeiros passos para um estudo comparativo do estado da arte da deduplicação . O intuito do estudo foi avaliar e comparar as características das técnicas de deduplicação PPJoin+, MARLIN e Febrl[18]. Em geral, na experimentação conduzida pelo autor, a técnica PPJoin+ apresentou um tempo de execução representativamente inferior, comparado às abordagens MARLIN e Febrl. Entretanto, a eficácia do PPJoin+ foi inferior nas bases de dados analisadas. Um ponto em aberto no trabalho é ausência de detalhes e justificativas concretas sobre a configuração dos experimentos.

2.4 Análise Comparativa

Esta seção apresenta uma comparação entre os trabalhos relacionados descritos neste capítulo e a classificação apresentada na Seção 2.2. Adicionalmente, utilizando a classificação proposta, é possível realizar uma comparação mais específica buscando salientar os principais desafios de pesquisa encontrados no âmbito da deduplicação de dados de grandes bases de dados. A Figura 2.13 ilustra o posicionamento de cada trabalho dentro da classificação proposta.

Na Seção 2.2, foi adotada a premissa de que a capacidade do deduplicador de processar grandes bases de dados está intimamente relacionada com o emprego de técnicas de blocagem. Tal hipótese pode ser observada analisando o tamanho das bases de dados empregadas na experimentação conduzida pelos respectivos autores. De um modo geral, nos métodos "*Sem Blocagem*" (Figura 2.13), os conjuntos de dados utilizados na experimentação são restritos a alguns milhares de registros. Com exceção de GRAVANO et al. (2001), tais trabalhos têm como objetivo a melhora na eficácia ou a redução do esforço manual da deduplicação, não oferecendo indicativos sobre a capacidade de processar grandes montantes de dados.

As técnicas "sem blocagem" e "automáticas" têm como intuito principal remover por completo a intervenção do usuário. Tais abordagens são restritas a pequenos conjuntos

[18] O *framework* Febrl implementa a formalização de F&S. Adicionalmente, oferece uma plataforma para a inserção de novos módulos para a deduplicação (CHRISTEN; CHURCHES, 2002)

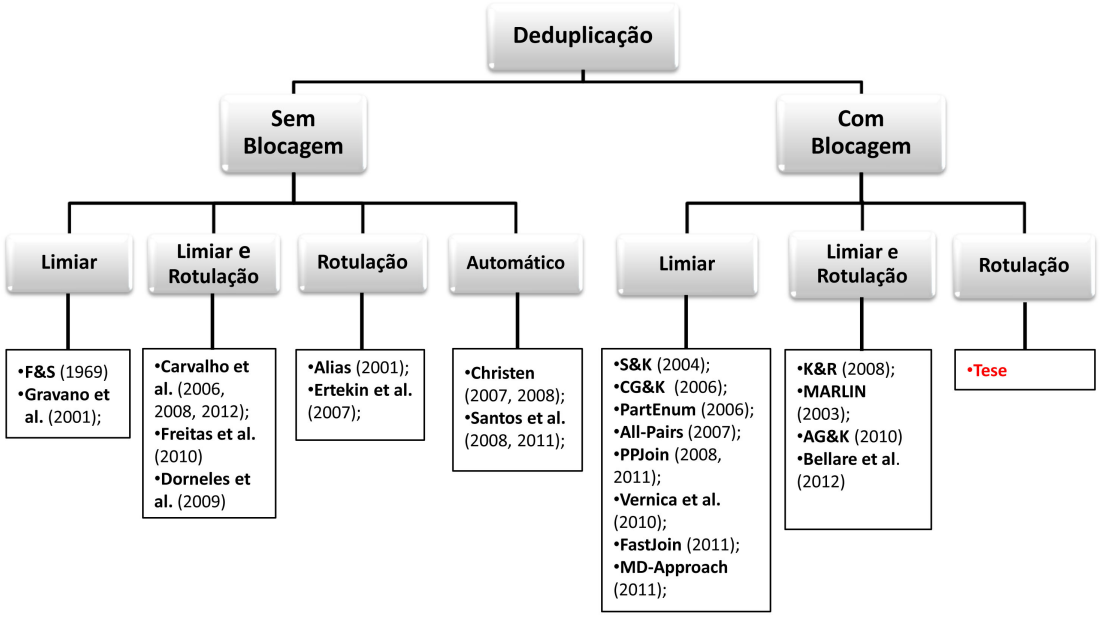


Figura 2.13: Classificação dos trabalhos relacionados de acordo com a utilização ou não da blocagem e do grau intervenção manual.

de dados devido à ausência de uma etapa de blocagem. Em especial, a abordagem SANTOS et al. (2008, 2011) demonstrou ser uma proposta promissora, no entanto, dependente de um alto consumo de recursos computacionais devido ao cálculo da similaridade entre todos os registros da base (custo quadrático de processamento). Uma segunda fraqueza evidenciada pelos autores é a baixa qualidade do método em bases de dados com um baixo nível de ruído. O que demonstra um comportamento instável e dependente do domínio para se obter o resultado esperado. Já em CHRISTEN (2008b), o método automático para rotulação foi incapaz de identificar os padrões mais ambíguos, resultando em uma classificação com uma qualidade restrita. De forma conclusiva, nos trabalhos apresentados nesse capítulo foi evidenciada a necessidade do usuário "ajustar" ou "calibrar" diretamente alguma das etapas internas do processo de deduplicação para alcançar a qualidade esperada.

De um modo geral, as abordagens que promovem o produto cartesiano entre os registros (técnicas "Sem Blocagem") não evoluíram de forma incremental, ou seja, não existe a predominância de um modelo ou abordagem globalmente adotado pelos trabalhos sucessores. Além disso, com algumas exceções, as técnicas não foram comparadas diretamente (empiricamente) com objetivo de salientar as possíveis vantagens de cada proposta. Essa ausência de transparência dificulta a identificação e a quantização das principais vantagens de cada técnica. Em contrapartida, no conjunto de trabalhos "Com blocagem e Limiares" é possível identificar uma série de abordagens que utiliza uma fundamentação semelhante, adicionando melhorias substancias a proposta inicial. Tais trabalhos são chamados de algoritmos *baseados em assinaturas* (*Probe cluster, SSJoin, All-pairs, PPJoin+, FastJoin*, entre outros). Mais especificamente, o primeiro esboço de método *baseado em assinaturas* foi proposto por GRAVANO et al. (2001), no qual pares candidatos são produzidos quadraticamente para a aplicação de uma série de filtros. Em Probe Cluster e SSjoin é inserida uma etapa de indexação/blocagem para minimizar o número de pares candidatos,

utilizando-se do conceito de *filtro de prefixo*. Nos trabalhos posteriores (All-pairs, PP-Join+, FastJoin), os métodos de filtragem foram aprimorados objetivando remover pares candidatos com baixas possibilidades de representarem uma duplicata.

Apesar da alta eficiência oferecida pelos métodos *baseados em assinaturas*, é necessária uma boa escolha dos valores de limiares para alcançar a eficácia esperada. Provavelmente, o usuário deve realizar sucessivas execuções variando a combinação de valores de limiares, o que incrementa substancialmente o tempo de execução. Nesse caminho, AS&B propõem uma abordagem baseadas em históricos (*logs*) para reduzir o custo computacional ao submeter diferentes combinações de valores de limiares, sem descartar o processamento executado nas etapas anteriores. Apesar de AS&B reduzir o processamento redundante, a tarefa de identificar os limiares *ideais* ainda é deixada sobre responsabilidade do usuário.

Nos trabalhos que utilizam "Limiares e a Rotulação", são propostas abordagens que empregam o conceito de *aprendizagem ativa* para reduzir o esforço do usuário no processo de rotulação dos pares (ARASU; GOTZ; KAUSHIK, 2010; BELLARE et al., 2012). O principal objetivo desses trabalhos é evitar que o usuário rotule pares que não são informativos para o treinamento do classificador. Para possibilitar a execução em grandes bases de dados, é inserida uma camada de blocagem manualmente configurada, utilizando funções de similaridade e limiares. Assim, tais abordagens dependem da especificação de diferentes configurações, tais como: (*i*) limiares de blocagem; (*ii*) variáveis para a configuração interna do método (por exemplo, tamanho da base de treinamento, número e tipo de funções de similaridade); e (*iii*) da rotulação de um conjunto de pares. Dessa forma, a intervenção manual é feita em diferentes níveis tornando-se um ponto crítico para a identificação de pares duplicados com uma alta qualidade.

A partir da Figura 2.13, pode-se observar três principais linhas de pesquisa. Em uma primeira linha, é possível evidenciar o esforço concentrado de pesquisa em propor soluções efetivas para melhorar a eficiência da deduplicação (GRAVANO et al., 2001; SARAWAGI; KIRPAL, 2004; CHAUDHURI; GANTI; KAUSHIK, 2006; ARASU; GANTI; KAUSHIK, 2006; BAYARDO; MA; SRIKANT, 2007; XIAO et al., 2011; AWEKAR; SAMATOVA; BREIMYER, 2009; VERNICA; CAREY; LI, 2010; DAL BIANCO; GALANTE; HEUSER, 2011; WANG; LI; FE, 2011). Em uma segunda vertente de pesquisa, concentram-se as abordagens com objetivo de reduzir o esforço do usuário utilizando a aprendizagem ativa (FREITAS et al., 2010; SARAWAGI; BHAMIDIPATY, 2002; ERTEKIN et al., 2007; BELLARE et al., 2012), heurísticas para seleção de pares informativos ((ARASU; GOTZ; KAUSHIK, 2010; KOPCKE; RAHM, 2008)), mapeamento entre limiares (DORNELES et al., 2007, 2009) e a automatização da deduplicação (CHRISTEN, 2008b; SANTOS et al., 2008, 2011). Por fim, em uma terceira direção, o objetivo é melhorar a eficácia da deduplicação a partir da programação genética (CARVALHO et al., 2006, 2008, 2012), combinação de *features* (BILENKO; MOONEY, 2003) e o aprimoramento das funções de similaridade (WANG; LI; FE, 2011). Assim, evidencia-se a quase ausência de abordagens que promovam a junção de algumas das principais características da deduplicação (eficiência, eficácia e a baixa intervenção do usuário) endereçando a deduplicação de uma forma mais ampla.

Por fim, a divisão proposta para apresentar os trabalhos relacionados, nesse capítulo, evidenciou a perspectiva em que esta tese se enquadra. Pode ser claramente observando uma lacuna, entre os trabalhos relacionados, em relação à redução da intervenção do usuário em cenários em que grandes volumes de dados estão presentes. O objetivo desta tese é endereçar diretamente tal lacuna. Mais ainda, este levantamento bibliográfico ilustrou

o potencial para o desenvolvimento de abordagens que incorporem técnicas já fundamentadas na bibliografia em relação à eficiência, como os métodos *baseado em assinaturas*, expandido suas propriedades para outros níveis, como a redução da intervenção do usuário. Cria-se, portanto, a possibilidade de oferecer uma proposta ortogonal ao estado da arte.

3 FS-DEDUP - UMA METODOLOGIA PARA A CONFIGURAÇÃO DA DEDUPLICAÇÃO EM GRANDES BASES DE DADOS

Neste capítulo, é apresentada a metodologia proposta para a identificação da configuração ideal na deduplicação de grandes bases de dados. A metodologia visa remover a demanda de um usuário especialista, capaz de definir valores de limiares ideais, requisitando somente a rotulação manual de um conjunto de pares (como duplicatas e não duplicatas). A intuição é de que a partir de um conjunto de pares rotulados torna-se possível extrair as informações necessárias para configurar idealmente as principais etapas da deduplicação (etapa de blocagem e de classificação) em bases de dados compostas por milhões de registros.

A metodologia proposta pode ser vista como uma camada entre o usuário e o deduplicador, possibilitando o ajuste da deduplicação em cenários compostos por grandes bases de dados. Para tanto, faz-se uso da alta eficiência das abordagens baseadas em assinaturas (descritas na Seção 2.3.2.1) como deduplicador. A partir de métodos de geração e filtragem de pares, os algoritmos *baseados em assinaturas* visam permitir o processamento de grandes montantes de dados, negligenciando a tarefa pouco intuitiva de identificar os valores de limiares. Dessa forma, a metodologia visa adicionar a importante característica da redução da intervenção manual a técnicas de deduplicação já consolidadas na bibliografia (algoritmos *baseados em assinaturas*). É importante salientar que a metodologia não está restrita a um único método *baseado em assinaturas*, devido à fundamentação semelhante adotada por tais algoritmos.

Nas seções a seguir, é apresentado um detalhamento do funcionamento da metodologia, chamada de FS-Dedup (*Framework for Signature based-deduplication* - Uma Metodologia para Deduplicação Baseada em Assinaturas). Para normalizar a nomenclatura, os métodos *baseados em assinaturas*, desta seção em diante, serão chamados de Sig-Dedup (*Signature-based Deduplication*). Mais especificamente, na Seção 3.1, é descrito detalhadamente o funcionamento interno da metodologia FS-Dedup. E, por fim, na Seção 3.2, são apresentadas as considerações finais.

3.1 FS-Dedup - Visão Geral

O principal objetivo da metodologia FS-Dedup é reduzir o esforço do usuário no processo de configuração da deduplicação. Como ilustrado na Figura 3.1, na perspectiva do usuário, a deduplicação deve ser vista como uma única tarefa. Em outras palavras, o usuário é requisitado somente para rotular um conjunto de pares (como duplicatas e não duplicatas) e a metodologia encarrega-se da configuração das principais etapas internas

da deduplicação (etapa de blocagem e de classificação) evitando a intervenção do especialista. Tal isolamento, promovido pela metodologia FS-Dedup, evita que o usuário seja exposto a tarefas complexas e pouco intuitivas, como, por exemplo, a definição de valores de limiares.

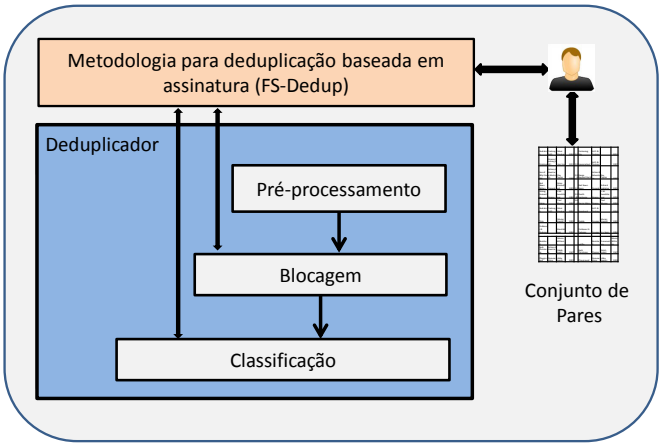


Figura 3.1: Visão geral da metodologia FS-Dedup na perspectiva do usuário.

A Figura 3.2 apresenta uma visão interna das três etapas da metodologia FS-Dedup. A primeira, chamada de Etapa de Ordenamento, tem como objetivo produzir um conjunto controlado de pares candidatos, objetivando maximizar a geração de pares duplicados. A segunda, chamada de Etapa de Seleção visa identificar os pares candidatos mais ambíguos (ou críticos), a partir da rotulação manual de um conjunto reduzido de pares. Note que o usuário somente interage com a metodologia FS-Dedup durante a Etapa de Seleção. Na terceira, chamada de Etapa de Classificação, os pares considerados críticos são avaliados a partir de dois métodos complementares de classificação com intuito de identificar os pares duplicados com uma alta qualidade. Cada uma das etapas é apresentada em detalhes nas seções a seguir.

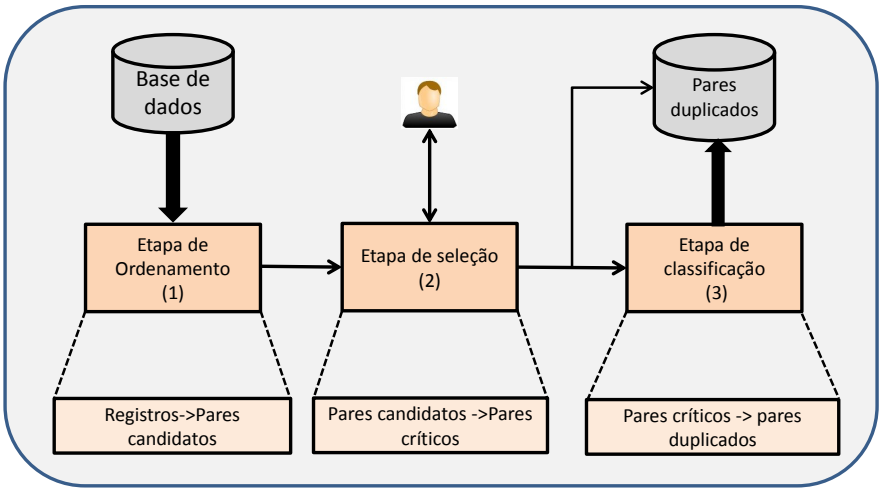


Figura 3.2: Etapas internas da metodologia FS-Dedup.

3.1.1 Etapa de Ordenamento

A Etapa de Ordenamento tem como objetivo configurar o processo de blocagem[1] para a criação de um conjunto de pares candidatos sem a intervenção do usuário. O principal desafio desta etapa é evitar uma excessiva geração de pares candidatos, sem impactar na qualidade dos pares, ou seja, evitar a remoção de pares duplicados. Nesse contexto, é proposta uma heurística para identificar o valor do limiar, chamado de *Limiar Inicial*, capaz de configurar a etapa de blocagem com foco na maximização da geração dos pares duplicados. Para evitar a intervenção do usuário, o *Limiar Inicial* representa um limiar único para todos os blocos.

É importante salientar que a Etapa de Ordenamento não possui o mesmo objetivo de um método automático de blocagem. Os métodos de blocagem visam maximizar o número de pares duplicados (revocação) e minimizar o número de pares não duplicados (precisão), enquanto a Etapa de Ordenamento tem como objetivo maximizar o número de pares duplicados, mantendo um controle do número de pares candidatos gerados. Os pares não duplicados, produzidos por esta etapa, são removidos nas etapas posteriores da metodologia FS-Dedup (Etapas de Seleção e Classificação).

Os pares candidatos selecionados pela Etapa de Ordenamento são efetivamente produzidos pelos filtros dos algoritmos Sig-Dedup (ou seja, os *filtros de prefixo, sufixo, tamanho e de posição*, detalhados na Seção 2.3.2.1) configurados com o valor do *Limiar Inicial*. A correta identificação do *Limiar Inicial* é um fator determinante para a produção de blocos efetivos a partir do algoritmo Sig-Dedup. Antes de detalhar a heurística para a identificação do *Limiar Inicial*, é importante contextualizar o funcionamento interno dos filtros do Sig-Dedup que são descritos a seguir.

O *filtro de prefixo* tem como objetivo produzir pares candidatos a partir da indexação dos termos menos frequentes de cada registro. Primeiramente, a frequência global dos termos é computada para possibilitar o reordenamento de cada registro. A intuição é que termos mais raros são mais informativos para a geração de pares duplicados que termos mais frequentes. Em seguida, o filtro de prefixo seleciona somente os termos posicionados no prefixo dos registros para a indexação (geração dos blocos de registros). Como apenas são indexados os termos menos frequentes de cada registro (prefixo do registro), é garantida a geração de um baixo número de pares candidatos. Por fim, no filtro de prefixo, os pares são gerados quadraticamente dentro de cada bloco.

Os filtros de *tamanho, sufixo e prefixo* são responsáveis pela filtragem de pares candidatos com baixa probabilidade de representarem duplicatas. Por exemplo, tais filtros são capazes de descartar o par candidato que apresenta somente um termo em comum (ou seja, o termo que gerou o par). O *filtro de tamanho* remove pares com uma alta variação no tamanho (número de termos). Já os filtros de posição e sufixo removem pares candidatos cujos termos possuem uma variação no posicionamento, acima de um limiar previamente especificado, segundo o ordenamento global dos registros.

O número de pares candidatos produzidos pelo processo de blocagem do Sig-Dedup depende diretamente do valor de limiar, manualmente definido. A identificação de um valor de limiar ideal depende das características de cada conjunto de dados, como, por exemplo, o nível de ruído, ou seja, bases de dados caracterizadas por um alto grau de ruído, produzem *tokens*[2] extremamente raros. Em outras palavras, a presença de termos

[1] Nesta seção, o termo *processo de blocagem* engloba a geração e filtragem dos pares candidatos como descrito pelos algoritmos *baseados em assinaturas*, como introduzidos na Seção 2.3.2.1.

[2] Como já discutido no Capítulo 2, o termo *token* pode representar um valor de atributo, um termo ou uma *substring*.

com muitas variações ou erros exige que um número mais elevado de *tokens* seja indexado pelo filtro de prefixo, em cada registro, para uma blocagem efetiva. Por outro lado, bases de dados caracterizadas pelo baixo nível de ruído são compostas por pares duplicados com alto grau de similaridade. Assim, os termos possuem poucas variações e, possivelmente, a indexação de um número reduzido de *tokens* em cada registro pelo filtro de prefixo é suficiente para identificar os pares duplicados.

A Figura 3.3 exemplifica a geração de blocos pelo *filtro de prefixo*, utilizando-se de dois valores de limiares em um mesmo conjunto de pares, ordenados pela frequência global dos termos. O primeiro limiar, ilustrado na Figura 3.3-(A), promove a indexação dos "dois" primeiros *tokens* de cada registro. O primeiro limiar produziu somente um par contendo os registros <R2, R3>. Já o segundo limiar, com valor "quatro" (ilustrado na Figura 3.3-(B)), produziu blocos contendo todos os registros da base de dados [<R1,R2>, <R2,R3>, <R3,R4>, <R1,R3>, <R1,R4>, <R2,R3>, <R2,R4>, <R3,R4>]. Em outras palavras, tal valor de limiar indexou os termos mais frequentes da base de dados (por exemplo, o termo "C" que está presente em todos os registros) resultando na impraticável geração quadrática de pares candidatos. Note-se que, no mesmo exemplo, caso o limiar de blocagem seja definido com valor "um" (ou seja, indexando somente o primeiro termo do prefixo) nenhum par candidato será produzido. Assim, bases de dados podem ser compostas por diferentes níveis de ruído, que impactam diretamente no valor do limiar adotado para uma efetiva geração dos pares candidatos. Por fim, após a geração dos pares candidatos, o Sig-Dedup aplica os filtros de tamanho, posição e sufixo para remover pares que não respeitarem o valor de limiar, previamente definido.

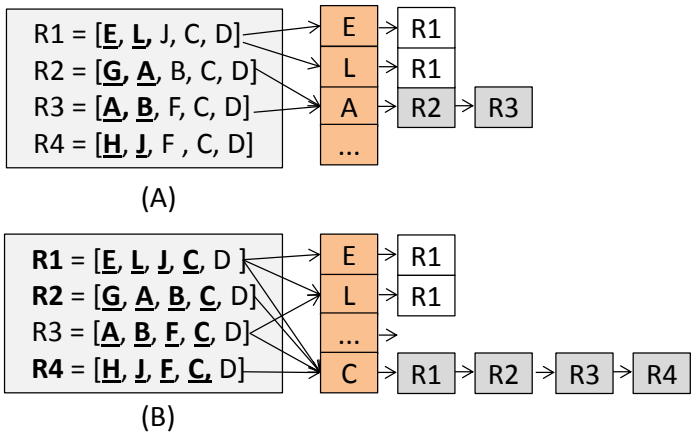


Figura 3.3: Exemplo da geração de pares candidatos, promovida pelo Sig-Dedup a partir de dois valores de limiares ("dois" e "quatro") em uma mesma base de dados.

Em grandes bases de dados, executar os filtros Sig-Dedup com um valor de limiar incapaz de remover os termos frequentes pode resultar em um alto número de pares candidatos e em uma demanda excessiva de recursos computacionais. Dessa forma, é proposta uma heurística para estimar o valor do *Limiar Inicial*. A heurística tem como objeto a identificação do limiar aproximado de blocagem para a criação do conjunto de pares candidatos. A intuição da heurística é avaliar o comportamento de diferentes limiares (a partir do número de pares candidatos) para identificar os valores que produzem amostras controladas de pares candidatos, indexando somente os termos menos frequentes. Para tanto, assume-se que tais termos (menos frequentes) são capazes de produzir efetivamente os pares duplicados (maximizando a revocação).

A Figura 3.4 ilustra os passos internos da heurística proposta para a Etapa de Ordenamento. Primeiramente, é selecionada uma amostra aleatória de registros (passo de "Amostragem") na base de dados. O objetivo é reduzir o consumo de recursos computacionais quando grandes bases de dados são processadas. Em seguida, a amostra é avaliada pelo Sig-Dedup (Pré-processamento, Blocagem e Filtragem) a partir de um valor de limiar (0,2), chamado de *limiar de teste*. O Sig-dedup produz um conjunto de pares, que é retornado para o passo de "Avaliação da Amostra", para que seja avaliado o número de pares que o *limiar de teste* produziu em relação à amostra. Se o valor de *limiar de teste* for insuficiente para remover os termos muito frequentes, então o conjunto de pares será substancialmente grande se comparado ao número de registros da amostra. Assim, o *limiar de teste* é sucessivamente incrementado em um valor fixo, para reduzir o número de *tokens* indexados e, consequentemente, o número de pares candidatos gerados. Note-se que um alto valor de limiar pode resultar na indexação de somente termos raros, reduzindo o número de pares candidatos corretamente recuperados.

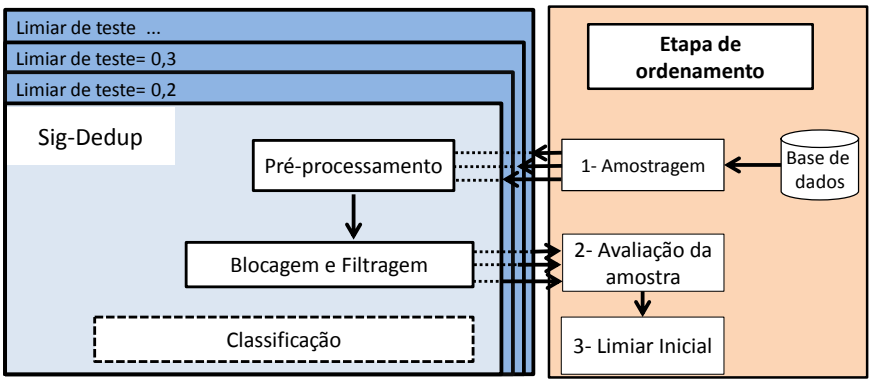


Figura 3.4: Passos internos da *Etapa de Ordenamento*.

Intuitivamente, um critério deve ser adotado para identificar o valor de limiar que evite a indexação de termos muito frequentes e, da mesma forma, a perda de pares candidatos duplicados. Assim, é definida a premissa de que o número de pares candidatos produzidos pelos filtros do Sig-Dedup deve ser menor que o número de registros contido na amostra. Tal definição, chamada de *critério de parada*, é reforçada pelo fato de que a amostragem aleatória naturalmente reduz o número de pares duplicados se comparado ao número total de pares da base de dados. As chances da seleção do registro e das suas respectivas duplicatas são reduzidas devido ao processo de amostragem. Adicionalmente, é assumido que bases de dados, compostas por um grande volume de dados, apresentam um pequeno fragmento de pares duplicados se comparado ao número total de registros.

A partir do *critério de parada*, o *limiar de teste*, que inicialmente recebe o valor 0,2, é incrementalmente ajustado, descartando os valores de limiares indesejáveis para o controle do número de pares candidatos. Por fim, o valor de *limiar de teste* que respeitar o *critério de parada* é definido como o *Limiar Inicial* para o processamento completo da base de dados. Note-se que o *critério de parada* reproduz uma aproximação do cenário ideal, ou seja, não são eliminados todos os pares não duplicados, devido às características peculiares de cada conjunto de dados. O *critério de parada* é formalizado a seguir.

Definição 3. *(Critério de parada) Seja S uma amostra aleatória, criada a partir de uma base de dados U e seja* th_j*=0,2, 0,3, ..., 0,9 um conjunto de limiares de teste com valores fixos. A amostra S é processada com cada limiar de teste* th_j*. O Limiar Inicial é*

definido como primeiro valor do limiar de teste (th_j) que resulte em um número de pares candidatos menor que o número de registros da amostra S.

Vale ressaltar que o *critério de parada* não é efetivo quando todos (ou a grande maioria) dos registros da base de dados apresentam um alto número de duplicatas, produzindo pares candidatos (quase) quadraticamente. Em outras palavras, isso significa que a base de dados é predominantemente composta por informações redundantes, o que raramente acontece em cenários reais.

Como ilustrado na Figura 3.4, o processo de classificação do Sig-Dedup não é acionado na Etapa de Ordenamento. Os pares que sobreviverem ao processo de filtragem do Sig-Dedup, mesmo se apresentarem um baixo valor de similaridade são considerados candidatos a representarem uma duplicata. Apesar de a Etapa de Ordenamento não ser responsável pela classificação dos pares, o Sig-Dedup aplica uma função de similaridade para quantificar o grau de semelhança de cada par. Tal valor de similaridade é utilizado como critério para a produção de um *ranking* de pares candidatos. O *ranking* é fundamental para o ordenamento dos pares como será discutido nas etapas seguintes da metodologia FS-Dedup.

3.1.2 Etapa de Seleção

A Etapa de Seleção tem como objetivo identificar as fronteiras da região crítica[3] do conjunto de pares candidatos, utilizando amostras de pares rotulados pelo usuário. A partir da identificação dos limites da região crítica, é possível dividir o *ranking* de pares candidatos em três conjuntos: (1) não duplicatas, composto por pares com uma baixa similaridade; (2) possíveis duplicatas, os pares são considerados "críticos" por apresentarem indícios insuficientes para defini-los como duplicatas ou para descartá-los; e (3) duplicatas, composto por pares altamente similares. A intuição por trás da Etapa de Seleção é a de que é possível reduzir o esforço do usuário para a criação de um conjunto de treinamento, identificando a região na qual os pares críticos estão presentes. Mais ainda, a identificação das fronteiras da região crítica permite enviar o conjunto de pares duplicados diretamente para a saída do deduplicador e, da mesma forma, descartar os pares considerados não duplicados. A Figura 3.5 apresenta uma visão geral do funcionamento da Etapa de Seleção.

A identificação das fronteiras da região crítica apresenta dois principais desafios: (*i*) produzir uma amostra representativa para ser manualmente rotulada (idealmente, a amostra deve ser representativa o suficiente para identificar a região crítica); e (*ii*) reduzir a rotulação manual, tanto quanto possível, sem perder a representatividade do conjunto de treinamento. A Etapa de Seleção endereça ambos os fatores, como detalham as estratégias apresentadas a seguir.

3.1.2.1 Estratégia para a seleção da amostra

A estratégia para a seleção da amostra objetiva construir um conjunto balanceado de pares candidatos presentes no *ranking* (criado na Etapa de Ordenamento). Idealmente, deve-se capturar somente os pares candidatos mais representativos, ou seja, capazes de prover evidências para a identificação da região crítica. A seleção de tais pares candidatos é uma tarefa complexa, que depende de diferentes variáveis, como o nível de ruído

[3]Como já descrito na Seção 2.3.1.1.1, a região crítica representa o conjunto de pares candidatos mais desafiadores para a tarefa de classificação devido à ausência de evidências representativas para a identificação de pares duplicados e o descarte dos pares não duplicados (FELLEGI; SUNTER, 1969).

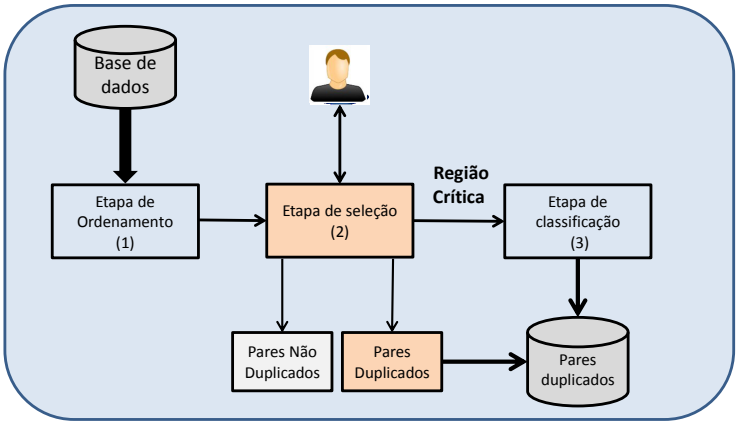


Figura 3.5: Visão geral do funcionamento da *Etapa de Seleção*.

de cada conjunto de dados. Além disso, a proporção entre pares duplicados e não duplicados pode variar acentuadamente, tornando o processo de amostragem extremamente desafiador para se aproximar do comportamento ideal.

Para minimizar o impacto de uma amostragem ineficiente, é proposto o conceito de *faixas*. As *faixas* correspondem a um conjunto de pares candidatos que respeitam determinados limiares de similaridade. Por exemplo, a faixa [0,2;0,3] contém todos os pares candidatos com valor de similaridade entre os valores de limiares 0,2 e 0,3. As *faixas* são constituídas a partir da segmentação do *ranking*, criado na *Etapa de Ordenamento* (Seção 3.1.1). Mais especificamente, o *ranking* dos pares candidatos é dividido em 10 *faixas* de acordo com a similaridade de cada par ([0,0;0,1], [0,1;0,2],...,[0,9;1,0]). O conceito de *faixas* é formalizado a seguir.

Definição 4. *(Faixas) Seja R(q) um ranking de pares ordenados segundo o valor de similaridade. Seja* t_i *e* t_j *limiares que variam em um passo fixo de 0,1 iniciando com os valores de 0,0 e 0,1, respectivamente. Considerando* $t = \{[t_{i1}; t_{j1}], [t_{i2}; t_{j2}], ..., [t_{i9}; t_{j9}]\}$ *um conjunto de limiares. Uma faixa representa um subconjunto de pares candidatos de R(q) posicionados entre dois valores de similaridade* $[t_i; t_j]$.

A seleção dos pares para a rotulação manual é realizada de forma automática, a partir de um processo de amostragem aleatória de pares, dispostos dentro de cada *faixa*. Primeiramente, é definida empiricamente uma constante *N*, que determina o número de pares de cada amostra. Em seguida, é criada uma amostra em cada *faixa*, contendo *N* pares candidatos. Dessa forma, é possível capturar pares contendo os diferentes padrões presentes em cada *faixa*.

É intuitivo que *faixas* com os menores valores de limiares (por exemplo, a faixa [0,0;0,1]) contenham um número substancialmente alto de pares candidatos não duplicados (devido à baixa similaridade dos pares). Tais *faixas* oferecem um alto indício de possuírem somente pares não duplicados que podem ser descartados. Por outro lado, nas *faixas* com altos valores de limiares (por exemplo, a *faixa* [0,9;1,0]) os pares não duplicados são raros devido ao elevado valor de similaridade das *faixas*. Os pares, presentes em tais *faixas*, oferecem altos indícios de conterem somente duplicatas. Já as *faixas* intermediárias (por exemplo, a *faixa* [0,4;0,5]) podem ser compostas por pares insuficientemente informativos e, possivelmente, pertencem à região crítica da base de dados. Assim, o processo de amostragem em *faixas* visa evitar que as regiões mais volumosas dominem a

seleção aleatória dos pares, produzindo amostras pouco informativas do conjunto de pares candidatos.

3.1.2.2 Estratégia de rotulação manual

A rotulação manual é fundamental para identificar a correta posição da região crítica do conjunto de dados. No entanto, a rotulação de pares com baixo nível de informatividade pode resultar na identificação imprecisa da região crítica. Nesse contexto, é proposta uma estratégia para a rotulação dos pares com o objetivo de identificar as fronteiras da região crítica com um esforço reduzido usuário.

As fronteiras da região crítica são identificadas, utilizando pares manualmente rotulados pelo usuário. O usuário é requisitado a rotular amostras aleatórias de *faixas* produzidas pela *Estratégia para a Seleção da Amostra* (descrita anteriormente). Tal processo é chamado de *rotulação de faixas*. A estratégia de rotulação manual obtém proveito do conceito de *faixas* para evitar a rotulação de pares pouco informativos. As *faixas* que compõem a região crítica representam o grupo de pares mais ambíguos e devem ser analisadas especificamente em busca de indícios para a classificação dos pares. Assim, identificar *faixas* que compõem a região crítica assume papel fundamental na seleção dos pares que, de fato, são mais desafiadores para o processo de classificação.

Na estratégia para rotulação, a cada rodada uma determinada amostra de *faixas* é rotulada. A partir das informações presentes na primeira amostra de *faixa* rotulada, são identificadas quais as próximas *faixas* que serão rotuladas. É assumido que *faixas* mais distantes da região crítica não oferecem informações relevantes e não devem ser rotuladas. A seguir são descritas em detalhes as definições propostas para a identificação da região crítica.

Definição 5. *Par Mínimo Verdadeiro (PV) representa o par duplicado que apresenta o menor valor de similaridade entre todos os pares verdadeiros (duplicados) do ranking de pares candidatos.*

Definição 6. *Par Máximo Falso (PF) representa o par não duplicado com o maior valor de similaridade entre os pares falsos (não duplicados) do ranking de pares candidatos.*

Definição 7. *Região crítica representa todos os pares candidatos posicionados entre os limiares de similaridade α e β definidos pelos pares PV e PF, respectivamente.*

A partir do *ranking* de pares candidatos, uma vez identificadas as posições dos pares *PV* e *PF*, têm-se um indício aproximado das fronteiras da região crítica. No entanto, a *rotulação das faixas* pode resultar em pares PV e PF longe das posições ideias. Para minimizar tal problema, é assumido que as *faixas* aos quais os pares PV e PF pertencem definem as fronteiras da região crítica. Por exemplo, se os valores de similaridade dos pares PV e PF são 0,35 e 0,71, respectivamente, então todos os pares com similaridade entre 0,3 e 0,8 serão considerados pertencentes à região crítica. Em outras palavras, os arredores dos pares PV e PF são analisados buscando minimizar erros no processo de seleção dos pares para a rotulação. Os limiares de fronteira da região crítica são chamados de α e β, como introduzidos na Definição 7.

Uma abordagem simplista para identificar os pares *PV* e *PF* é submeter o usuário à rotulação de todas as *faixas* do *ranking*, sucessivamente. No entanto, algumas amostras de *faixas*, quando rotuladas, não oferecem ganhos relevantes de informação por não estarem localizadas próximas à região crítica. Neste contexto, é proposta uma sequência de passos para minimizar a rotulação das amostras de *faixas*. Em um primeiro passo, uma *faixa*

intermediária (FI) é selecionada para a rotulação do usuário (FI=[0,5; 0,6]). A partir das informações presentes na FI são selecionadas as próximas *faixas* para serem manualmente rotuladas. Por exemplo, se a região crítica se delimitar entre as *faixas* 0,4 e 0,6, a estratégia proposta demanda a rotulação das *faixas* [0,3; 0,4], [0,4;0,5], [0,5;0,6] e [0,6;0,7], ou seja, somente quatro de um total de 10 *faixas* são rotuladas, reduzindo o esforço manual do usuário. Em um caso extremo, a rotulação é realizada em todas as 10 *faixas*. No entanto, nos demais casos, é possível reduzir a demanda de rotulação manual.

Ao rotular a *faixa* intermediária (FI), três cenários podem ser encontrados:

1. *a FI contém somente pares não duplicados.* Os pares com valor de similaridade superior a *FI* oferecem um forte indício de pertencerem à região crítica da base de dados. A(s) *faixa*(s) superior(es) deve(m) ser rotulada(s) até identificar a primeira *faixa* contendo pares duplicados e não duplicados. Dentro de tal *faixa*, é possível identificar o par PV e, consequentemente, definir o valor do α. Um novo processo de rotulação de *faixas* é realizado até identificar uma *faixa* que contenha somente pares duplicados. A *faixa* composta somente por duplicatas está fora da região crítica da base de dados e a *faixa* anterior contém o par PF, definindo o valor do β.

2. *a FI contém somente pares duplicados.* A região crítica pertence à(s) *faixa*(s) com valor de similaridade inferior aos pares da *FI*. Primeiramente, as *faixas* que apresentam indícios de conterem a região crítica são rotuladas até alcançarem uma *faixa* composta por pares duplicados e não duplicados. Nesta *faixa*, é encontrado o par PF e definido o valor do β. A rotulação prossegue até atingir uma *faixa* contendo somente pares não duplicados. Novamente, na *faixa* anterior é identificado o par PV, definindo o valor do α.

3. *a FI é composta por pares duplicados e não duplicados.* As *faixas* com valor de similaridade superior ao FI (acima da faixa [0,5;0,6]) são rotuladas até se identificar uma amostra que contenha somente pares duplicados. A *faixa* anterior a esta, contém o par PF e o valor β é definido. *Faixas* contendo pares com valor de similaridade inferior a FI (abaixo da *faixa* [0,5;0,6]) são rotuladas buscando identificar uma *faixa* composta somente por pares não duplicados. A *faixa* anterior à faixa composta por pares não duplicados contém o par PV, portanto, o valor de α é definido.

Uma visão algorítmica da estratégia de rotulação é apresentada no Algoritmo 1. O algoritmo recebe como entrada o conjunto de *faixas* ($F = f_0, f_1, ...f_9$) criadas a partir do *ranking* de pares candidatos. A função *Rotulação_Da_Amostra(Fi)* (Linha 4) seleciona uma amostra aleatória de pares para serem manualmente rotulados, dentro da *faixa i*. As funções *selecionaPV* e *selecionaPF* são responsáveis pela seleção do par verdadeiro com menor valor de similaridade (PV) e o do par falso com o maior valor de similaridade (PF), respeitando as definições 4 e 5, respectivamente. Como ponto de partida do algoritmo, é identificado se a *faixa* intermediária (FI) pertence ao Cenário 1, 2 ou 3 (como descrito anteriormente). Se FI é composta unicamente por pares verdadeiros (Linha 5), então a região crítica está posicionada nas *faixas* inferiores (com valor de similaridade abaixo da FI), como descrito no Cenário 1. Em seguida, as faixas inferiores (i-1) são rotuladas (Linhas 6-11) até atingir a primeira faixa composta por pares verdadeiros e falsos (Linha 6), ou seja, tal *faixa* pertence à região crítica. Tal *faixa* é utilizada para identificar o par PF (Linha 10) a partir da função *SelecionaPF(i)*. A última *faixa* pertencente à região

Algorithm 1 Identificação das fronteiras da *região crítica*.

Require: Conjunto de faixas $F = f_0, f_1, ...f_9$ (por exemplo, f_0 representa a faixa $f_{0,0-0,1}$)

1: $i \leftarrow 5$;
   Definição da faixa intermediária
2: $PF \leftarrow Null$;
3: $PV \leftarrow Null$;
4: $RFi \leftarrow$Rotulação_Da_Amostra(f_5);
   ****Cenário 1****
5: **if** RF_i composto somente por *pares verdadeiros* **then**
6: **while** RF_i não contém somente pares falsos **do**
7: $i \leftarrow (i-1)$;
8: $RF_i \leftarrow$Rotulação_Da_Amostra(f_i);
9: **if** (RFL_i não contém somente pares verdadeiros) and ($PF == Null$) **then**
10: $PF \leftarrow SelecionaPF(F_i)$;
11: **end if**
12: **end while**
13: $PV \leftarrow SelecionaPV(F_{i+1})$;
14: **return** PV, PF and RF_P ;
15: **end if**;
   ****Cenário 2****
16: **if** F_i composto somente por pares falsos **then**
17: **while** F_i não contém pares verdadeiros **do**
18: $i \leftarrow i+1$;
19: $L_{Pi} \leftarrow$Rotulação_Da_Amostra(f_i);
20: **if** (f_i contém somente pares verdadeiros) and ($PV = Null$) **then**
21: $PV \leftarrow SecionaPV(f_i)$;
22: **end if**
23: **end while**
24: $PF \leftarrow SelecionaPF(f_{i-1})$;
25: **return** PV, PF and f_P ;
26: **end if**
   ****Cenário 3****
27: **if** f_i contém pares verdadeiros e falsos **then**
28: **while** f_i não contém somente pares verdadeiros **do**
29: $i \leftarrow i+1$;
30: $f_i \leftarrow$Rotulação_Da_Amostra(f_i);
31: **end while**
32: $PF \leftarrow SelecionaPF(f_{i-1})$;
33: $i \leftarrow 5$;
34: **while** f_i não contém somente pares falsos **do**
35: $i \leftarrow i-1$;
36: $f_i \leftarrow$Rotulação_Da_Amostra(f_i);
37: **end while**
38: $PV \leftarrow SelecionaPV(f_{i+1})$;
39: **return** PV, PF and f_P;
40: **end if**

crítica (composta por pares duplicados e não duplicados) contém o par PV (Linha 13). Identificando a *faixa* composta somente por pares falsos (ou seja, a primeira faixa fora da fronteira da região crítica), é possível selecionar a última *faixa* pertencente à região crítica e identificar o par PV. Caso a FI contenha somente pares duplicados (Linha 16), então a região crítica está posicionada acima da FI (acima do valor de similaridade 0,5). Assim, as

faixas acima da FI contêm os pares PV e PF (Linhas 16-25), como ilustrado no Cenário 2. Por fim, se a FI é composta por pares duplicados e não duplicados, então a região crítica está posicionada nos entorno da FI (Linhas 27-40), como descrito no Cenário 3. Nesse contexto, as faixas superiores a FI são rotuladas buscando identificar o par PF (Linhas 28-33), ou seja, é identificada a *faixa* anterior àquela que contém somente pares verdadeiros. As *faixas* inferiores ao FI são rotuladas buscando identificar a *faixa* anterior àquela que contém somente pares não duplicados (Linha 34-38). O algoritmo retorna os valores dos limiares de fronteira (PV e PF) e as amostras manualmente rotuladas. O valor de similaridade dos pares PV e PF definem, respectivamente, os valores dos limiares α e β.

A Figura 3.6 ilustra um exemplo da região crítica posicionada abaixo da faixa intermediária (Cenário 1). Primeiramente, a FI é manualmente rotulada (Figura 3.6-(A)) e unicamente pares verdadeiros são identificados (representados na figura pelos círculos). Em seguida, a faixa inferior [0,4;0,5] é rotulada, identificando a presença de pares verdadeiros e falsos. O par PF é definido com o valor de similaridade de 0,48 (par não duplicado com valor de similaridade mais elevado) e, consequentemente, o β é definido com o valor de 0,5 (Figura 3.6-(B)). A rotulação de faixas prossegue até se identificar a faixa [0,2;0,3] contendo somente pares não duplicados. O par PV é definido com o valor de similaridade de 0,35 e o α recebe o valor de 0,3 (Figura 3.6-C). Por fim, a região crítica é composta pelos pares com similaridade entre os valores de α e β, como ilustrada na Figura 3.6-C.

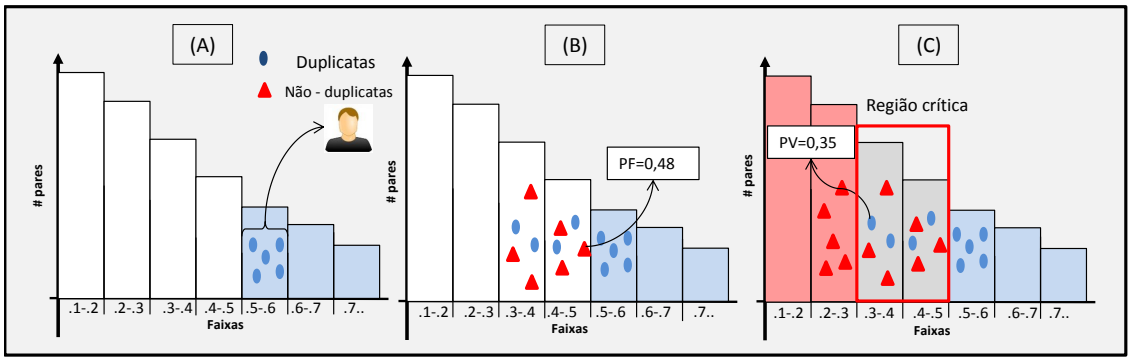


Figura 3.6: Um exemplo de funcionamento da estratégia de identificação da Região Crítica.

A partir da definição dos limites da região crítica, os pares posicionados entre α e β são enviados para a etapa de classificação. E, como já descrito, as *faixas* (ou pares) com similaridade inferior ao valor de α são descartadas. Esse processo reduz substancialmente o número de pares enviados para a Etapa de Classificação, tendo em vista que as *faixas* com menor valor de similaridade contêm um acentuado número de pares candidatos. Os pares com similaridade acima de β são definidos como duplicatas e enviados diretamente para a saída do deduplicador.

3.1.3 Etapa de Classificação

A Etapa de Classificação tem como objetivo identificar os pares duplicados posicionados na região crítica do conjunto de pares candidatos. A Etapa de Classificação recebe como entrada os pares pertencentes à região crítica e o conjunto de pares manualmente rotulados durante a Etapa de Seleção. Como já mencionado, a(s) *faixa*(s) da região crítica

Produto	Marca	Modelo	Descrição
Blu-ray 3D	LG	BP325	Com o Blu-ray BP325 aproveite toda sua galeria de filmes, músicas e fotos diretamente de um HDD externo.
Bluray 3D	LG	BP325	O aparelho é equipado com Dolby True HD e sem dúvidas é a opção ideal para ser o seu primeiro Bluray 3D Player.

Tabela 3.1: Ilustração de alguns de valores de registros de uma base de dados contendo informações sobre produtos.

reúne(m) os pares candidatos mais complexos de serem classificados. Por consequência, métodos específicos devem ser empregados para obter uma deduplicação com uma alta qualidade. Dois métodos complementares são utilizados para a classificação dos pares críticos: FS-Dedup-NGram e FS-Dedup-SVM. O método FS-Dedup-NGram tem como objetivo identificar pequenas variações ou anomalias presentes nos pares, promovendo a fragmentação e o reordenamento dos registros a partir do ordenamento global da frequência dos termos. Já o método FS-Dedup-SVM, utiliza o algoritmo de classificação SVM para construir uma função de classificação, levando em consideração a importância de cada atributo.

Mais especificamente, no método FS-Dedup-NGram, os filtros do Sig-Dedup são refinados para o processamento em nível de NGram, ou seja, as *strings* dos registros são fragmentadas em pequenas unidades (*substrings*) para capturar pequenas variações ou erros. Os *filtros de prefixo, tamanho, sufixo e de posição* (introduzidos na Seção 2.3.2.1) promovem a indexação dos NGram menos frequentes, levando em consideração a frequência global dos *tokens*. O reordenamento dos NGram possibilita que a inversão de valores de campos seja facilmente capturada. Os filtros do Sig-Dedup (configurados em nível de NGram) são aplicados para reduzir o número de pares candidatos e, em seguida, uma função de similaridade é utilizada para quantizar a similaridade de cada par.

Uma primeira desvantagem do método FS-Dedup-NGram é a definição de um peso único para todos os atributos, ou seja, longos atributos, com informações pouco relevantes, podem resultar na baixa similaridade de um par duplicado. A Tabela 3.1 ilustra um pequeno exemplo de dois registros que notavelmente representam um mesmo par. Como o FS-Dedup-NGram não leva em consideração a relevância dos atributos, o atributo "*Descrição*", altamente divergente, resulta na baixa similaridade do par do exemplo, incoerentemente com as informações presentes nos outros atributos ("*Produto*", "*Marca*" e "*Modelo*"). Uma segunda desvantagem do FS-Dedup-NGram é o alto custo computacional quando utilizado em grandes bases de dados, mesmo com a alta eficiência proposta pelo Sig-Dedup. Tal custo é resultado da segmentação das strings em NGram, que impacta diretamente no processo de quantização da similaridade. No entanto, tal custo computacional é reduzido pela metodologia FS-Dedup devido ao processamento de somente pares pertencentes à região crítica da base de dados.

Já o método FS-Dedup-SVM, utiliza o algoritmo SVM para a definição de pesos específicos para cada atributo, utilizando as informações presentes no conjunto de treinamento. Dessa forma, é possível identificar os atributos mais relevantes e reduzir a importância dos atributos menos informativos. No entanto, o método de classificação SVM depende do mapeamento de valores numéricos, produzidos a partir da quantização da similaridade de cada atributo. A quantização da similaridade pode ser promovida a partir de uma variada gama de funções de similaridade, como descrito no decorrer Capítulo 2. Cada função de similaridade apresenta características particulares, indicadas para determinados

tipos de dados, por exemplo, textos, números, strings, nomes, etc. Em outras palavras, não existe uma função de similaridade única global capaz de separar otimamente os pares similares dos não similares. Uma segunda desvantagem do FS-Dedup-SVM é a necessidade do correto posicionamento dos valores dos atributos (segmentação dos atributos), ou seja, a inversão no valor de campos dificulta substancialmente a correta quantização da similaridade de um par.

Ambos os métodos de classificação SVM e NGram possuem vantagens e desvantagens, que são exploradas pela metodologia FS-Dedup. O FS-Dedup-SVM e o FS-Dedup-NGram são treinados, utilizando o conjunto de pares rotulados manualmente durante a Etapa de Seleção (Seção 3.1.2).

O FS-Dedup-NGram depende da identificação de um limiar de classificação para identificar os pares duplicados. Tal limiar, chamado de *limiar NGram*, é identificado utilizando as evidências presentes no conjunto de pares rotulados durante a *Etapa de Seleção*. A partir da tokenização NGram, é possível identificar pequenos erros nos pares e separar mais precisamente os pares pertencentes à região crítica. A intuição para a identificação do *limiar NGram* parte da ideia de que existe um determinado agrupamento de pares duplicados e não duplicados que, quando ordenados em forma de *ranking*, podem ser facilmente separados. A seguir são descritos os passos para identificar o *limiar NGram*, a partir do conjunto de treinamento:

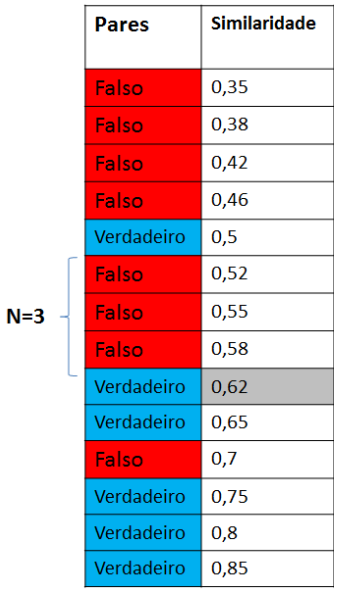
1. a similaridade de cada par pertencente à amostra rotulada é computada, utilizando a tokenização NGram. Na *Etapa de Ordenamento* (Seção 3.1.1), a similaridade dos pares candidatos foi computada a partir da tokenização de termos, ou seja, cada termo é mapeado para um *token*. A tokenização de NGram permite identificar pequenas alterações ou variações nos registros;

2. os pares rotulados são, então, ordenados incrementalmente, de acordo com seus valores de similaridade NGram;

3. uma janela deslizante, com tamanho fixo, é aplicada sobre os pares rotulados. A janela é realocada em uma posição até alcançar a última janela, contendo somente pares não duplicados. O valor de similaridade do primeiro par duplicado, seguinte à última janela, define o valor do *limiar NGram*.

O método de identificação do *limiar Ngram* assume a presença de uma fronteira que separa os pares duplicados dos pares não duplicados, quando aplicado à tokenização NGram. A intuição é semelhante ao conhecido método de deduplicação *Sorted neighborhood Method* (SNM) (HERNáNDEZ; STOLFO, 1995b). A Figura 3.1.3 exemplifica o processo de identificação do *limiar Ngram*. Primeiramente, os pares são organizados em ranking de acordo com o grau de similaridade. Em seguida, uma janela deslizante, com tamanho igual a três, é utilizada para separar os pares duplicados (ou verdadeiros) dos não duplicados (ou falsos). O primeiro par duplicado após a última janela contendo somente pares falsos define o valor do *limiar NGram* com valor 0,62 (valor de similaridade do par). Por fim, o método FS-Dedup-NGram é executado com o valor do *limiar NGram* para a identificação dos pares críticos duplicados.

Somente é possível aplicar o método de identificação do limiar NGram em uma amostra rotulada. O algoritmo Sig-Dedup é aplicado utilizando a tokenização NGram e o *limiar NGram*, a todos os pares presentes na região crítica. Por fim, os pares que atingiram o valor de similaridade definido pelo *limiar NGram*, são enviados para a saída do deduplicador, representando pares duplicados.



Pares	Similaridade
Falso	0,35
Falso	0,38
Falso	0,42
Falso	0,46
Verdadeiro	0,5
Falso	0,52
Falso	0,55
Falso	0,58
Verdadeiro	0,62
Verdadeiro	0,65
Falso	0,7
Verdadeiro	0,75
Verdadeiro	0,8
Verdadeiro	0,85

Figura 3.7: Exemplo da identificação do *limiar Ngram*.

Diferentemente do FS-Dedup-NGram, o método FS-Dedup-SVM demanda somente um conjunto rotulado para promover a classificação dos pares pertencentes à região crítica da base de dados. No entanto, a geração de um modelo de predição efetivo está diretamente relacionada à representatividade do conjunto de treinamento, contendo um número insuficiente de padrões pode impactar na qualidade do processo de deduplicação.

3.2 Considerações finais

Neste capítulo foi apresentada uma nova metodologia para a configuração da deduplicação em grandes bases de dados, a partir de um conjunto de pares manualmente rotulados. Para isso, foi proposta uma metodologia chamada FS-Dedup, que permite remover a demanda de um usuário especialista (capaz de identificar valores de limiares), requisitando somente a rotulação de um conjunto reduzido de pares, como demonstrado nos experimentos do Capítulo 5.

Mais especificamente, o trabalho apresenta as seguintes contribuições: (1) uma heurística para a geração dos pares candidatos, visando maximizar a geração de pares duplicados; (2) a definição de estratégias para reduzir o esforço manual a partir da identificação da região em que os pares mais ambíguos (ou críticos) se encontram; (3) a definição de dois métodos complementares para a classificação dos pares críticos.

A metodologia apresentada neste capítulo foi projetada utilizando, como deduplicador, os algoritmos *baseados em assinaturas*. Como o FS-Dedup promove um mapeamento entre o deduplicador e o usuário, novas otimizações, promovidas nos algoritmos *baseados em assinaturas*, podem ser prontamente adicionados à metodologia proposta. Dessa forma, propicia-se que o FS-Dedup evolua com novas melhorias dos algoritmos baseados em assinaturas, acompanhando novos aprimoramentos da eficiência e da escalabilidade da deduplicação.

Como será demonstrado nos experimentos, o FS-Dedup é capaz de identificar a configuração ideal, tanto nas bases de dados reais como nas sintéticas, com um reduzido esforço do usuário. Será apresentado, ainda, o comportamento dos métodos de classifica-

ção, assim como as principais vantagens e desvantagens de tais métodos, de acordo com as características de cada base de dados.

4 T3S - UMA ABORDAGEM PARA A SELEÇÃO DE PARES INFORMATIVOS EM GRANDES BASES DE DADOS

Neste capítulo, é apresentada uma abordagem chamada de T3S (*Two Stage Sampling Selection- Seleção de pares em dois passos*), com o objetivo de produzir uma amostra reduzida de pares "altamente" informativos[1] no contexto de grandes bases de dados. No capítulo anterior, foi apresentada uma metodologia com objetivo de identificar a configuração ideal, a partir de um conjunto de pares manualmente rotulados. Na metodologia, foi proposta uma estratégia com objetivo de selecionar pequenas amostras aleatórias de pares considerados informativos para a identificação da região crítica (região composta por pares mais desafiadores para a tarefa de classificação). A amostragem aleatória permite que pares com informações redundantes sejam desnecessariamente rotulados, elevando o custo manual de rotulação. Portanto, cria-se a possibilidade de reduzir ainda mais o esforço manual do usuário, removendo pares que, quando rotulados, não acrescentam ganho de informatividade ao conjunto de treinamento. Assim, a abordagem T3S, que complementa a metodologia FS-Dedup, visa à remoção da intervenção especialista nas principais etapas da deduplicação, permitindo reduzir substancialmente o número de pares manualmente rotulados pelo usuário.

O restante do capítulo está organizado da seguinte forma. A Seção 4.1 apresenta um método de aprendizagem ativa, que serviu de embasamento para o desenvolvimento da abordagem T3S. A Seção 4.2 detalha o funcionamento da abordagem T3S. Por fim, a Seção 4.3 apresenta as considerações finais.

4.1 Aprendizagem Ativa Baseada em Regras - SSAR

O cenário da deduplicação de dados é tipicamente caracterizado por um vasto número de pares desprovidos de informações sobre os seus respectivos rótulos. Construir manualmente uma amostra rotulada que contenha uma informatividade semelhante ao conjunto completo de pares é uma tarefa custosa e complexa, que depende da seleção de pares representativos da base de dados. Para aliviar esse problema, métodos de aprendizagem ativa são propostos com objetivo de construir, com auxílio do usuário, uma amostra representativa da base de dados (SETTLES, 2010). O principal desafio, por trás dos métodos de aprendizagem ativa, é evitar que pares com informações irrelevantes ou redundantes sejam desnecessariamente rotulados sem causar perdas de representatividade.

Tradicionalmente, os métodos de aprendizagem ativa utilizam um comitê de classificadores para identificar os pares mais informativos (COHN; ATLAS; LADNER, 1994;

[1] Os pares mais informativos são capazes de caracterizar um amplo número de pares sem perdas de informação.

FREUND et al., 1997; BALCAN; BEYGELZIMER; LANGFORD, 2009). Para o treinamento do comitê, faz-se necessário o ajuste mínimo dos classificadores a partir de um conjunto inicial de treinamento. No entanto, o conjunto inicial de pares rotulados, criado pelo usuário, pode influenciar diretamente a seleção dos pares pelos classificadores. Se os classificadores forem configurados de uma forma incondizente em relação ao conjunto de dados (por exemplo, um treinamento com a quase ausência de uma das classes), o método de aprendizagem ativa pode convergir prematuramente, criando um conjunto de treinamento pouco abrangente.

Nesse contexto, em SILVA; GONCALVES; VELOSO (2011) é proposto um novo método de aprendizagem ativa, chamado de SSAR (*Rule-based Active Sampling- Aprendizagem ativa baseada em regras*), com objetivo de selecionar um conjunto reduzido de pares sem a demanda de um treinamento inicial. A intuição por trás do método SSAR é que os pares com o menor número de atributos em comum em relação aos pares já rotulados sejam os mais informativos e devem ser adicionados ao conjunto de treinamento. Diferentemente dos métodos tradicionais, SSAR converge naturalmente, removendo a necessidade do usuário determinar o tamanho do conjunto de treinamento (a partir de valores de limiares).

Mais especificamente, o SSAR produz projeções de cada par em relação ao atual conjunto de treinamento. Tais projeções têm como objetivo remover os valores de atributos que não estão presentes no atual conjunto de treinamento. Para cada projeção, é quantizado o número de regras de associação (VELOSO et al., 2008), com o objetivo de avaliar o ganho de informatividade de cada par. Em outras palavras, um par cujas projeções produzem um alto número de regras é considerado pouco informativo em relação ao atual conjunto de treinamento. Por outro lado, um par cujas projeções resultam em um baixo número de regras é considerado mais relevante devido ao baixo número de atributos em comum em relação ao atual conjunto de treinamento.

Algorithm 2 Algoritmo de aprendizagem ativa baseado em regras (SILVA; GONCALVES; VELOSO, 2011).

Require: Conjunto de pares T e σ_{min} (≈ 0)
Ensure: Conjunto de treinamento D

while true **do**
  for all $u_i \in T$ **do**
    $P_{ui} \leftarrow D$ projetado de acordo com u_i
    $R_{ui} \leftarrow$ Extração das regras do conjunto P_{ui}
  end for
  if $D = \emptyset$ **then**
    $\lambda_{ui} \leftarrow u_i$, tal que u_i é o par mais representativo de T
  else
    $\lambda_{ui} \leftarrow u_i$ tal que $\forall_{uj} : |R_{ui}| \leq |R_{uj}|$
  end if
  if $\lambda_{ui} \in D$ **then**
    break;
  else
    Rotulação(λ_{ui})
    $D \leftarrow D \cup \{\lambda_{ui}\}$
  end if
end while

Uma visão algorítmica do SSAR é apresentada no Algoritmo 2. Primeiramente, cada par não rotulado u_i é selecionado para produzir uma projeção Pu_i do atual conjunto

de treinamento *D* (Linha 3), ou seja, a projeção Pu_i é criada removendo os valores de atributos do atual conjunto de treinamento que não estão presentes no par u_i. Em seguida, um conjunto de regras de associação R_{ui} é produzido a partir da base de dados projetada P_{ui} (Linha 4). O número de regras representa a informatividade de cada par não rotulado (u_i). O objetivo da projeção P_{ui} é identificar o par mais dissimilar, comparado com o atual conjunto de treinamento. O par u_i que gerar uma projeção com o menor número de regras ($|R_{ui}| \leq |R_{uj}|$) (Linha 9) é rotulado (Linha 14) será inserido no conjunto de treinamento (Linhas 15). Uma nova rodada é executada e, novamente, cada par da base de dados u_i é projetado sobre o atual conjunto de treinamento. Na primeira rodada, o SSAR seleciona o par que produz mais regras (o par mais redundante) para ser rotulado e inserido no conjunto de treinamento. Por fim, SSAR converge quando for selecionado um par que já está presente no conjunto de treinamento (Linhas 11 e 12).

Um exemplo da execução do Algoritmo 2 é ilustrado na Figura 4.1. O exemplo parte do princípio de que uma rodada do SSAR já foi executada e o par Du_1 foi inserido no conjunto D (representando o par mais redundante no conjunto de pares não rotulados). Em seguida, para cada par não rotulado, são criadas projeções da base rotulada D. Para cada projeção P_{ui}, são produzidas as regras de associação (R). Observe que o número de regras geradas depende diretamente do número de atributos das projeções, ou seja, a projeção P_{u1}=<A, B, C, D> produz 15 regras, enquanto a projeção D_{u3}=<A> produz somente uma regra. Nesse cenário, o par P_{u3} é considerado o mais informativo (com o menor número de regras), devendo ser rotulado e adicionado ao conjunto rotulado (D). Uma nova rodada é iniciada e, para cada par não rotulado, são criadas duas projeções em relação aos pares rotulados D_{u1} e D_{u3}. O par u_4 (ou a projeção P_{u4}), que produziu o menor número de regras (7+7=14), é rotulado e inserido no conjunto D. A execução prossegue até que o algoritmo SSAR selecione um par, da base não rotulada, que já está presente na base rotulada.

Como já mencionado, o algoritmo SSAR exige que todos os pares não rotulados sejam reprojetados à medida que novos pares são rotulados e adicionados ao conjunto de treinamento. Tal processo resulta em um custo computacional praticamente inviável em bases de dados compostas por grandes volumes de dados. Um simples experimento foi executado com objetivo de quantizar o tempo de processamento do algoritmo SSAR. Em uma base de dados composta por 105.000 registros, foram necessárias mais de 13 horas de processamento, já em uma segunda base de dados, contendo 150.000 registros, foram necessárias cerca de 31 horas para a execução do algoritmo SSAR[2]. De fato, o experimento ilustrou a inviável demanda de recursos computacionais quando milhões de registros são processados pelo algoritmo SSAR.

Nesse cenário, um dos objetivos da abordagem proposta neste capítulo é possibilitar que o algoritmo SSAR seja capaz de processar grandes conjuntos de dados, mantendo a capacidade de selecionar pares altamente informativos. O objetivo principal da abordagem proposta é selecionar conjuntos reduzidos e balanceados de pares candidatos independente do volume da base de dados.

[2]As bases de dados foram produzidas sinteticamente e serão descritas em detalhes na Seção 5.1. O experimento foi executado em uma processador Core 2 Duo 2,4 GHz com 4 GB de memória RAM.

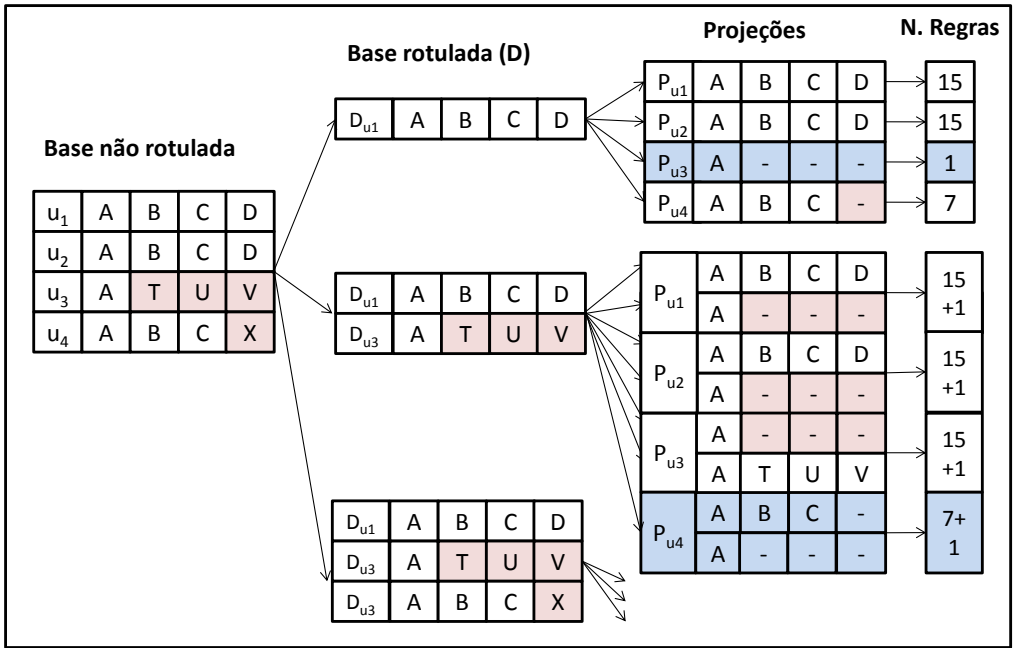


Figura 4.1: Exemplo da execução do algoritmo SSAR.

4.2 T3S - Uma abordagem em dois passos para a construção do conjunto de treinamento

O objetivo da abordagem T3S é reduzir o esforço do usuário na deduplicação de grandes bases de dados, a partir da seleção de um conjunto reduzido de pares em dois passos principais. O primeiro passo visa fragmentar o conjunto de pares não rotulados em amostras reduzidas e balanceadas. Na segunda etapa, é empregado o algoritmo de aprendizagem ativa SSAR (SILVA; GONCALVES; VELOSO, 2011) para selecionar incrementalmente os pares mais informativos nas amostras geradas previamente. A primeira etapa é essencial para permitir o processamento de grandes volumes de dados, visto que o algoritmo SSAR é caracterizado pela alta demanda de recursos computacionais.

A abordagem T3S deve ser combinada com métodos de blocagem e classificação para permitir a execução das principais etapas da deduplicação. Para tanto, os dois passos da abordagem T3S são integrados à metodologia FS-Dedup (apresentada no Capítulo 3), evitando a especificação manual de limiares na configuração das principais etapas da deduplicação.

A Figura 4.2 ilustra graficamente as principais etapas da abordagem T3S, integrada à metodologia FS-Dedup. Primeiramente, é identificada a configuração da etapa de blocagem para a geração dos pares candidatos, utilizando a Etapa de Ordenamento (apresentada na Seção 3.1.1). Em seguida, o primeiro passo da abordagem T3S é invocado para a criação das amostras de pares candidatos. No segundo passo, as amostras são processadas incrementalmente pelo SSAR para a remoção de pares contendo informações redundantes. Dessa forma, é possível identificar, com um esforço reduzido do usuário, a posição da região crítica no conjunto de pares não rotulados. Por fim, os pares pertencentes à região crítica são processados por dois métodos de classificação para a predição dos pares

críticos. O método de classificação T3S-SVM (empregando o classificador SVM) utiliza o algoritmo de aprendizagem de máquina SVM para criar um modelo de classificação dos pares, a partir do conjunto de treinamento criado pela abordagem T3S. Já o método T3S-NGram, empregando o método de classificação NGram, fragmenta os registros em *substrings* para possibilitar a identificação de pequenos erros ou variações. Mais detalhes dos métodos de classificação SVM e NGram foram apresentados na Seção 3.1.3.

A seguir, são descritos os detalhes envolvendo os dois passos da abordagem T3S. É apresentado também como a região crítica é identificada a partir dos pares selecionados pela abordagem T3S.

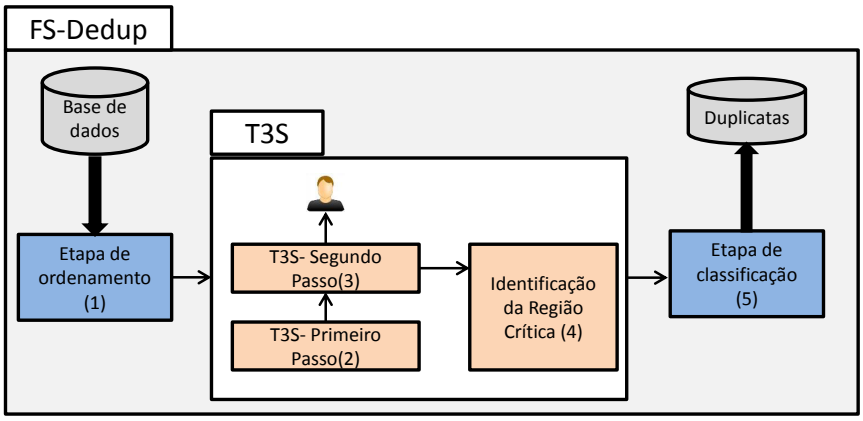


Figura 4.2: Visão geral da abordagem T3S.

4.2.1 T3S-Primeiro passo

O primeiro passo da abordagem T3S tem como objetivo criar um conjunto reduzido e balanceado de pares candidatos. A ideia consiste em fragmentar o *ranking* (criado durante a Etapa de Ordenamento, apresentada na Seção 3.1.1) para permitir a seleção de um conjunto de amostras, visando capturar as características específicas do conjunto de pares candidatos. Como já mencionado, tal passo é fundamental para a seleção de um número relativamente reduzido de pares, viabilizando a execução do segundo passo da abordagem T3S.

Um método simplista para a geração de amostras é coletar pares aleatoriamente no conjunto de pares candidatos. No entanto, tal método produz amostras pouco informativas devido ao conjunto de pares candidatos ser composto predominantemente de pares não duplicados. Como a tarefa da deduplicação visa identificar os pares duplicados e descartar os pares não duplicados, é necessário que o conjunto de treinamento seja formado por um número consistente de pares duplicados. Assim, o segundo passo da abordagem T3S dificilmente é capaz de selecionar um conjunto de treinamento informativo devido à irregularidade ou pouca representatividade dos pares duplicados.

Nesse cenário, o primeiro passo da T3S adota o conceito de *faixas* para permitir que as amostras selecionadas contenham uma diversidade similar ao conjunto de pares não rotulados. Similar ao conceito detalhado na Seção 3.1.1, uma *faixa* representa um conjunto de pares candidatos, delimitados por dois valores de limiares (por exemplo, todos os pares com um valor de similaridade entre 0,0 e 0,1 constituem uma *faixa*). Em cada *faixa*, são coletados aleatoriamente pares para compor as amostras. A aleatoriedade do

processo de seleção em cada *faixa* propicia que diferentes padrões de informatividade sejam incorporados às amostras. Por exemplo, as *faixas* presentes em níveis mais baixos de similaridade naturalmente são compostas por pares não duplicados, enquanto as *faixas* com valores de similaridade mais elevadas são compostos predominantemente por pares duplicados. Dessa forma, é possível produzir um balanceamento entre os pares duplicados e não duplicados, criando o cenário para que o segundo passo da abordagem T3S seja capaz de selecionar um conjunto de treinamento representativo.

4.2.2 T3S- Segundo Passo

O primeiro passo da abordagem T3S é capaz de produzir amostras reduzidas e balanceadas de pares, utilizando um processo de amostragem aleatória. No entanto, tal amostragem aleatória negligencia o fato de muitos pares serem compostos por informações irrelevantes ou redundantes, causando custos adicionais de rotulação. O objetivo do segundo passo da abordagem T3S é reduzir o número de pares a serem manualmente rotulados, executando incrementalmente um processo de seleção ativa.

O processamento das amostras de *faixas* garante que o custo computacional do algoritmo SSAR seja reduzido, mesmo quando o número de pares de cada *faixa* seja substancialmente elevado. Contudo, o processamento de *faixas* acrescenta um novo problema da redundância de informatividade de pares presentes nas bordas (ou fronteiras) das *faixas*. Por exemplo, a Tabela 4.1 ilustra dois pares duplicados, com pequenas variações (por exemplo, “UFRGS”–> “UFRG”, “do”–> “da”). Note que os pares apresentam informações similares e são inseridos em diferentes *faixas* ([0,8;0,9] e [0,9;1,0]). Caso o algoritmo SSAR seja executado isoladamente, pode ocorrer a seleção de ambos os pares por estarem presentes em diferentes amostras de *faixas*. Dessa forma, informações redundantes podem acarretar o aumento do custo manual sem agregar ganho de informatividade.

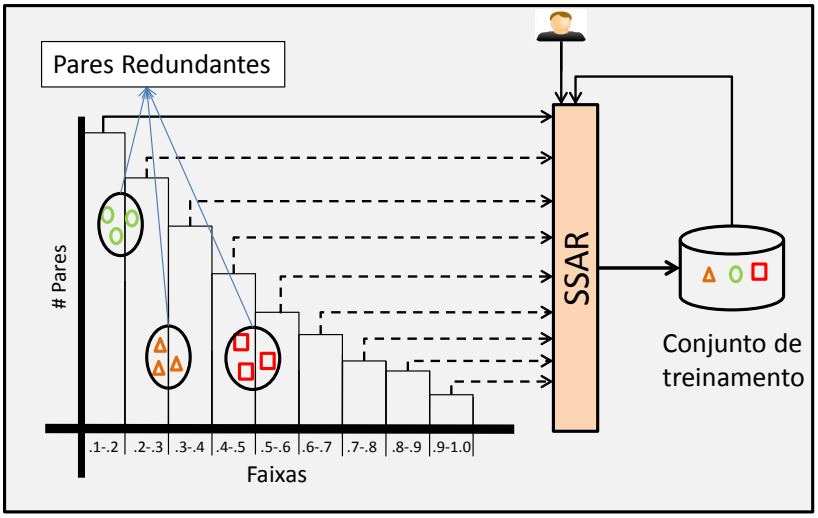


Figura 4.3: Exemplo de pares redundantes, posicionados nas fronteiras das *faixas*, identificados pela abordagem T3S.

Como alternativa para o problema da redundância entre *faixas*, é proposta a execução incremental das amostras pelo algoritmo SSAR, como ilustrada na Figura 4.3. Primeiramente, uma primeira *faixa* ([0,0; 0,1]) é processada pelo SSAR para selecionar somente os pares mais informativos e não redundantes. Tais pares são rotulados e inseridos no conjunto de treinamento (que inicialmente está vazio). Em seguida, as próximas *faixas*

são processadas, incrementalmente pelo SSAR, utilizando o conjunto de treinamento já selecionado pelas etapas anteriores. A intuição é rotular somente os pares que contenham informações dissimilares aos pares já presentes no conjunto de treinamento.

É importante notar que a seleção incremental dos pares dentro das *faixas* produz amostras contendo informações complementares, ou seja, cada amostra de *faixa* processada pelo SSAR depende das informações das *faixas* anteriores para constituir um conjunto de treinamento completo. Dessa forma, as primeiras amostras de *faixas* processadas pelo SSAR resultam em um número mais elevado de pares que as outras amostras de *faixas* devido à abundância de padrões de pares e ao conjunto de treinamento estar inicialmente vazio. À medida que novas *faixas* são processadas, o conjunto de treinamento torna-se mais completo e somente os pares mais raros são selecionados para a rotulação.

Tabela 4.1: Exemplo de pares similares que estão posicionados em diferentes *faixas*.

	Par			Sim	Faixa
Par 1	Universidade federal **do** rio grande do sul (**UFRGS**)	vs.	Universidade federal **da** rio grande do sul (**UFRG**)	0,89	[0,8;0,9]
Par 2	Universidade federal do rio grande do sul (**UFRGS**)	vs.	Universidade federal do rio grande do sul (**UFRG**)	0,91	[0,9;1,0]

4.2.3 Identificação da Região Crítica

Na Seção 3.1.2.2, foi apresentada uma abordagem para a identificação da região crítica com o objetivo de rotular as amostras de *faixas* consideradas relevantes para a identificação dos pares críticos. Diferentemente desse cenário, o conjunto de treinamento selecionado pela abordagem T3S busca identificar incrementalmente, dentro de cada faixa, quais são os pares mais informativos. A seguir, é proposta uma estratégia específica para identificar as fronteiras da região crítica, a partir do conjunto de pares selecionado pela abordagem T3S.

A região crítica é identificada a partir da seleção de pares de fronteira. Para tanto, é necessária a definição do par verdadeiro com menor valor de similaridade (chamado de PV), e o par falso, com maior valor de similaridade (chamado de PF). Os pares PF e PV fornecem um indício do posicionamento da região crítica. No entanto, como os pares PF e PV podem ser imprecisos, é assumido que as faixas que contêm tais pares determinam o posicionamento da região crítica. Por exemplo, se o par PF for encontrado com valor 0,35, toda a faixa [0,3;0,4] será considerada crítica. Um detalhamento mais amplo das definições dos pares PF e PV foi apresentado na Seção 3.1.2.2.

Saliente-se que o par PV representa o par duplicado mais dissimilar em uma faixa predominantemente composta por pares não duplicados, ou seja, a faixa é composta por um acentuado número de pares redundantes não duplicados, que são pouco informativos para a identificação do par PF. O mesmo acontece com as faixas compostas por pares com elevados valores de similaridade, nas quais os pares duplicados são pouco informativos para a identificação do par PV. Nesse cenário, o algoritmo SSAR é fundamental para descartar os pares redundantes (pouco dissimilares), possibilitando a identificação da região crítica com um esforço reduzido do usuário.

O Algoritmo 3 detalha como os pares PV e PF são identificados a partir do conjunto de pares selecionados pela abordagem T3S. Primeiramente, o algoritmo SSAR é invocado, incrementalmente, para a seleção ativa (e a rotulação do usuário) dos pares mais informativos em cada faixa (Linhas 2-5). Em seguida, os pares selecionados em cada *faixa*,

Algorithm 3 Identificação das fronteiras da região crítica a partir dos pares selecionados pela abordagem T3S.

Require: Amostras de faixas $A[] = A_1, A_2, A_3..A_9$
1: PV **and** $PF \leftarrow Null$;
2: $R_A[] \leftarrow NULL$;
3: **for** $i = 0 \rightarrow 10$ **do**
4: $\quad R_A[i] \leftarrow SSAR(A_i, R_A)$
5: **end for**
6: **for** $i = 0$ to 10 **do**
7: $\quad$ **if** ($R_A[i]$ contém pares verdadeiros) **and** ($PV = Null$) **then**
8: $\quad\quad PV \leftarrow SelecionaPV(R_A[i])$;
9: $\quad$ **end if**
10: $\quad$ **if** ($R_A[i]$ contém pares falsos) **and** ($PV! = Null$) **then**
11: $\quad\quad PF \leftarrow SelecionaPF(R_A[i])$;
12: $\quad$ **end if**
13: **end for**
14: **return** PV **and** PF **and** R_A;

iniciando pelo menor valor de similaridade, são avaliados, buscando identificar o par PV. A função *SelecionaPV* é responsável pela seleção do par verdadeiro com menor valor de similaridade (Linha 8). Em seguida, o algoritmo busca o par não duplicado com maior valor de similaridade. A última faixa, composta por pares não duplicados, define o valor do par PF (Linhas 10 e 11). Por fim, os valores de PV e PF são retornados para a definição dos entornos da região crítica. O conjunto rotulado, então, é utilizado para a configuração da Etapa de Classificação.

Todos os pares com valores de similaridade entre as fronteiras da região crítica são enviados para a Etapa de Classificação objetivando a predição dos pares duplicados.

4.3 Considerações finais

A principal contribuição deste capítulo é a proposta de uma abordagem capaz de mitigar o problema da seleção de pares informativos na presença de grandes volumes de pares não rotulados. Mais especificamente, a abordagem proposta, chamada T3S, apresenta dois principais objetivos: (*i*) produzir amostras reduzidas e balanceadas de pares; e (*ii*) remover pares redundantes das amostras previamente selecionadas. O primeiro objetivo é tratado a partir de uma abordagem de seleção em faixas, visando reduzir o custo computacional do processo de seleção de pares em grandes conjuntos de dados e, mais ainda, facilitar a seleção dos pares informativos no cenário altamente desbalanceado da deduplicação de dados. O segundo objetivo é abordado utilizando incrementalmente um algoritmo de aprendizagem ativa nos conjuntos de pares previamente selecionados para a seleção de um conjunto de treinamento altamente informativo.

Além disso, foi ilustrado o potencial de integração da abordagem T3S com algumas das etapas internas da metodologia FS-Dedup, proposta no Capítulo 3. Mais especificamente: a metodologia FS-Dedup, equipada com o algoritmo Sig-Dedup, recebe a tarefa da geração e classificação dos pares, enquanto a abordagem T3S promove a seleção de conjuntos altamente informativos de pares e a identificação da região crítica da base de dados. Desse modo, tal integração visa tirar proveito da capacidade da metodologia FS-Dedup em identificar a configuração ideal (remover a necessidade da definição de valores de limiares) e acrescenta a capacidade de reduzir, mais ainda, o esforço do usuário na

tarefa da calibração da deduplicação.

Como será demonstrado nos experimentos, a abordagem T3S é capaz de reduzir substancialmente a demanda por pares rotulados, mantendo a qualidade da classificação competitiva em relação aos *baselines* analisados. Nos experimentos, é constatado que a abordagem T3S foi capaz de selecionar uma reduzida amostra de treinamento e, até mesmo, aprimorar a eficácia do processo em uma base de dados. Isso demonstra que a seleção de pares altamente informativos é uma estratégia promissora para a minimização da intervenção manual da deduplicação de grandes conjuntos de dados. Detalhes da experimentação são apresentados no Capítulo 5.

5 EXPERIMENTOS

Neste capítulo, são descritos os experimentos realizados com intuito de analisar o comportamento das propostas apresentadas na tese. Os objetivos são listados a seguir:

- demonstrar a efetividade da *Etapa de Ordenamento*, descrita na Seção 3.1.1. A premissa a ser validada é relativa à capacidade de produzir um conjunto controlado de pares candidatos visando maximizar a revocação;
- demonstrar o comportamento dos métodos FS-Dedup-SVM e o FS-Dedup-NGram em relação à eficácia, apresentados na Seção 3.1.3;
- demonstrar como a metodologia FS-Dedup se comporta na seleção dos pares candidatos a serem rotulados em relação à qualidade da deduplicação, validando as ideais apresentas na Seção 3.1.2. A partir desses experimentos, pretende-se ilustrar a capacidade da metodologia FS-Dedup em selecionar uma amostra representativa que possibilite a configuração das principais etapas da deduplicação em grandes bases de dados;
- demonstrar como a abordagem T3S se comporta em relação à redução da amostra rotulada e à qualidade da deduplicação. Pretende-se com esses experimentos ilustrar que é possível reduzir o tamanho da amostra de treinamento demandada pela metodologia FS-Dedup sem perdas na qualidade da deduplicação. Esses experimentos validam as ideias expostas no Capítulo 4.

Este capítulo está organizado da seguinte forma. Na Seção 5.1, são apresentadas as características das bases de dados reais e sintéticas utilizadas na experimentação. As métricas utilizadas na avaliação experimental são descritas na Seção 5.2. Os detalhes sobre a configuração adotada nos experimentos são apresentados na Seção 5.3. Os experimentos realizados e os resultados obtidos analisando as principais etapas da metodologia FS-Dedup são apresentados na Seção 5.4. Na Seção 5.5, são descritos os experimentos com objetivo de validar o comportamento da abordagem T3S. Por fim, a Seção 5.6 apresenta as considerações finais.

5.1 Descrição das bases de dados

Os experimentos desse capítulo foram avaliados utilizando bases de dados sintéticas e reais. Note-se que as bases de dados reais compostas por grandes volumes de dados raramente possuem um gabarito, devido ao alto custo manual de produzi-los. Desse modo, o cenário real foi criado integrando duas bases de dados reais de um mesmo domínio.

Tabela 5.1: Descrição das bases de dados reais e sintéticas.

Base de dados	Número total de registros	Número de duplicatas (Porcentagem)	Número caracteres	Número de termos
DsgenA	105.000	5.000 (5%)	82	235.362
DsgenB	120.000	20.000 (20%)	81	257.083
DsgenC	150.000	50.000 (50%)	82	291.639
DBLP	1.995.539	-	169	2.013.244
CiteSeer	811.408	-	129	644.479
IMDB	1.080.000	-	55	551.583
NetFlix	160.000	-	76	70.654

Adicionalmente, foram utilizadas bases de dados sintéticas para promover uma análise mais detalhada do comportamento das abordagens avaliadas.

Os dados sintéticos foram criados fazendo uso da ferramenta de geração de dados Dsgen (CHRISTEN, 2005). O gerador Dsgen foi escolhido devido ao extenso número de variáveis que podem ser controladas pelo usuário como, por exemplo: o número de registros gerados, o número de duplicatas por registro, o número de alterações (ou erros), o tipo de distribuição, os tipos de variações, etc. Mais especificamente, o gerador Dsgen produz registros baseado em tabelas de frequência de nomes, sobrenomes, endereços, código postais (dos Estados Unidos), número de telefones, etc. As tabelas de frequência são compostas por 656 sobrenomes, 302 nomes e 931 endereços. Em seguida, cada registro original é alterado para a criação da(s) duplicata(s) utilizando a inserção ou remoção de caracteres, junção de palavras, inversão de atributos, remoção de atributos, etc. O comportamento simulado nas bases de dados sintéticas é baseado em padrões identificados nas bases de dados reais (CHRISTEN, 2005).

Como descrito na Tabela 5.1, foram gerados três conjuntos de dados sintéticos (chamados de DsgenA, DsgenB e DsgenC). Cada base de dados é constituída com um determinado número de registros e uma proporção de duplicatas. A configuração da geração das bases de dados sintética visa se aproximar de três cenários: (*i*) baixo nível de ruído (100.000 registros contendo 5.000 duplicatas); (*ii*) nível médio de ruído (100.000 registros contendo 20.000 duplicatas); e (*iii*) um elevado nível de ruído (100.000 registros contendo 50.000 duplicatas). Os registros foram gerados a partir de 10 atributos: “nome”, “sobrenome”, “idade”, “sexo”, “ano de nascimento”, “endereço”, “estado”, “número da rua’, “telefone” e a “data de nascimento”.

Apesar dos dados sintéticos serem importantes para a avaliação experimental, o engessamento a bases de dados oriundas de um único domínio (com características similares) pode ser insuficiente para revelar fraquezas da metodologia proposta. Isso ressalta a necessidade da avaliação em bases de dados reais pertencentes a domínios distintos. No entanto, bases de dados com grandes montantes de informações são, geralmente, de difícil acesso devido a políticas de privacidade.

Foram produzidos dois cenários contendo pares duplicados reais. O primeiro, chamado IMDBxNetFlix, é resultado da integração entre as bases de dados IMDB[1] e NetFlix[2]. Ambas, armazenam informações sobre mídias (por exemplo, filmes, shows, séries, etc.). Os dados foram coletados a partir de algoritmos desenvolvidos a partir de rotinas públicas (APIs). Como detalhado na Tabela 5.1, a base de dados IMDB é composta por mais

[1]IMDB: http://www.imdb.com, último acesso 5/10/2012

[2]NetFlix: http://www.netflix.com, último acesso 3/09/2012

de um milhão de registros e a base de dados NetFlix contém cerca de 160.000 registros. A base de dados NetFlix tem como foco uma parcela específica do mercado de mídias, dessa forma, não pode ser considerada um "subconjunto" da base de dados IMDB (GEMMELL; RUBINSTEIN; CHANDRA, 2011). Em outras palavras, apenas uma fração dos registros entre as duas bases representam pares duplicados. Tais bases de dados foram integradas a partir dos atributos "título da mídia", "diretor" e o "ano de lançamento".

O segundo cenário real foi criado integrando as bases reais oriundas das bibliotecas digitais DBLP[3] e CiteSeer[4]. Ambas armazenam informações relacionadas a publicações científicas (especialmente no domínio da Ciência da Computação). Em mais detalhes, a biblioteca digital CiteSeer promove um mecanismo automático (sem intervenção do usuário) de varredura de documentos dispersos na Web (por exemplo, arquivos no formato PDF, PS, entre outros). Em seguida, as informações relevantes são extraídas dos documentos (por exemplo, título, autor(es), resumo, referências, etc.) para compor a base de dados (GILES; BOLLACKER; LAWRENCE, 1998). Nesse cenário, a base de dados coletada do CiteSeer, composta por 811.000 registros, sofre de diferentes níveis de ruído, por exemplo, registros incompletos, diferentes padrões na nomenclatura, entre outras impurezas. Por outro lado, a biblioteca digital DBLP utiliza-se de um conjunto de *scripts* e ferramentas para a indexação das publicações. Mais ainda, usuários especialistas são responsáveis pela validação dos dados com objetivo de oferecer um serviço com uma alta qualidade dos dados (PETRICEK et al., 2005). O volume de dados coletados para compor a base de dados DBLP representa cerca de dois milhões de registros. A base de dados DBLPxCiteSeer foi criada a partir da integração dos atributos "título da publicação", "autor(es)" e o "ano de publicação".

Os detalhes de cada base de dados podem ser observados na Tabela 5.1. A tabela apresenta o número médio de caracteres de cada registro ("Número de caracteres") e o número total de termos das bases de dados ("Número de termos"). Observe que os registros das bases de dados DBLP e CiteSeer são compostos por aproximadamente duas vezes mais caracteres (170 caracteres) e por um número acentuado de termos comparado aos demais conjuntos de dados (DBLP contém cerca de dois milhões de termos, enquanto a base de dados NetFlix é composta por 70.000 registros). Isso se deve a presença do atributo "título da publicação" que é composto por um vasto vocabulário de termos.

5.2 Métricas utilizadas

A seguir, são apresentadas as métricas utilizadas para a avaliação experimental. A avaliação é baseada em três principais métricas: precisão (*Precision*), revocação (*Recall*) e F1 (*F-measure*) (MANNING; RAGHAVAN; SCHTZE, 2008). As métricas são formalizadas na Tabela 5.2. Mais especificamente, a precisão indica a razão entre o número de pares duplicados recuperados (Verdadeiros Positivos-VP) em relação ao número total de pares duplicados (Verdadeiros Positivos-VP e Falsos Positivos-FP). A medida de revocação indica o percentual de pares duplicados recuperados (VP) sobre o total de pares duplicados existentes na base de dados (Verdadeiros Positivos mais os Falsos Negativos). A medida F1 ou *F-measure* calcula a média harmônica ponderada dos valores de revocação e precisão. A medida F1 quantifica quanto o método maximizou o número de pares VP e minimizou o número de pares FP e FN. Uma classificação perfeita dos pares resulta em um F1 igual ao valor 1,0, ou seja, foram recuperados todos os pares duplicados

[3]DBLP: http://wwww.informatik.uni-trier.de/ley/db, último acesso 22/9/2012
[4]CiteSeer: http://citeseer.ist.psu.edu, último acesso 2/12/2012

(revocação igual ao valor 1,0) e somente pares duplicados (precisão igual ao valor 1,0).

Tabela 5.2: Descrição das métricas utilizadas.

Precisão (P)	$\frac{\|VP\|}{(\|VP\|+\|VF\|)}$
Revocação (R)	$\frac{(\|VP\|)}{(\|VP\|+\|FN\|)}$
F1	$\frac{(2xPxR)}{(P+R)}$

Devido a possíveis ruídos causados pela amostragem aleatória dos pares, cada experimento foi repetido 10 vezes. Juntamente com a média aritmética de cada experimento, é apresentado o desvio padrão com intuito de dar uma ideia da amplitude dos valores encontrados. A significância estatística foi confirmada utilizando o teste estatístico T (*Student's t-test*) (MANNING; RAGHAVAN; SCHTZE, 2008) com um limiar padrão de significância α de 0,05. Quando o valor de *p*, calculado pelo teste T, for menor que α, é ilustrado que existe uma diferença estatística entre as abordagens comparadas, com uma confiança de 95%.

Como as bases de dados reais não apresentam gabarito, a validação foi realizada utilizando cinco usuários (estudantes de graduação do curso de Ciência da Computação da UFRGS) para a rotulação manual de nove subconjuntos (amostras de faixas), cada uma composta por cerca de 100 pares. No total, foram rotulados cerca de 900 pares. Para a criação do conjunto de teste, foram avaliados manualmente todos os pares com o atributo "ano de publicação ou lançamento" com os valores de 1988, 1989 e 1990 (ARASU; RÉ; SUCIU, 2009). A intuição de utilizar os valores de anos é criar uma pequena amostragem sem tendenciar a seleção dos pares. A amostragem resultou em 3.009 e 3.137 pares para a criação do conjunto de teste para as bases de dados DBLPxCiteSeer e IMDBxNetFlix, respectivamente. Os pares que resultaram em certa ambiguidade de rotulação (por exemplo, metades dos usuários associaram o par a classe de duplicata ou não duplicata e o restante dos usuários a classe contrária) foram desambiguados utilizando informações adicionais, por exemplo, cruzando as informações dos respectivos *Web sites*, metadados (quando disponíveis), entre outras informações.

5.3 Configuração dos experimentos

A metodologia FS-Dedup e a abordagem T3S utilizam como deduplicador o algoritmo baseado em assinatura Sig-Dedup proposto por VERNICA; CAREY; LI (2010), como descrito na Seção 2.3.2.1. Na *Etapa de Ordenamento*, o algoritmo Sig-Dedup foi configurado com a tokenização em nível de termos (ou palavras). Já na *Etapa de Classificação* (FS-Dedup-NGram e T3S-NGram), a tokenização foi ajustada em nível de NGram com um custo maior de processamento (que a tokenização em nível de palavras), mas com uma maior capacidade de identificar pequenas variações nos registros. Após o processo de filtragem promovido pelo Sig-Dedup, é empregada a função de similaridade Jaccard (ELMAGARMID; IPEIROTIS; VERYKIOS, 2007) para a quantização da similaridade de par, como originalmente empregado por VERNICA; CAREY; LI (2010).

O tamanho da janela deslizante adotado pelo FS-Dedup-NGram foi experimentalmente identificado com valor igual a dois nos conjuntos de dados sintéticos e cinco nos

conjuntos de dados reais. Tais valores são explicados devido às bases sintéticas serem compostas de um gabarito (toda a base de dados está rotulada) e sem inconsistências (possíveis erros de rotulação). Nos conjuntos reais, o gabarito, manualmente produzido, é suscetível a erros de rotulação dos usuários. Dessa forma, para evitar distorções na identificação do *limiar NGram*, é utilizado um valor igual a cinco nas bases de dados reais. Já nos métodos T3S-(NGram e SVM), como o método de aprendizagem ativa é capaz de remover redundâncias de informações, foram adotados tamanhos da janela deslizante igual a duas posições, novamente identificado experimentalmente.

Tanto o FS-Dedup-SVM quanto o T3S-SVM utilizam o pacote LibSVM (CHANG; LIN, 2011) como implementação do algoritmo SVM. Mais especificamente, o SVM foi configurado com o *kernel* RBF e com os valores dos parâmetros γ e *cost(C)* obtidos a partir da validação cruzada (*cross-validation*) da base de dados de treinamento. O vetor de características (*features*) é computado a partir da função de similaridade Jaccard Ngram, devido à alta eficiência na quantização da similaridade oferecida por tal função de similaridade (ELMAGARMID; IPEIROTIS; VERYKIOS, 2007).

A implementação do método de aprendizagem ativa SSAR foi disponibilizada pelos respectivos autores SILVA; GONCALVES; VELOSO (2011). O algoritmo SSAR tem como requisito que a entrada seja composta por valores de atributos nominais, ou seja, o valor computado pelas funções de similaridade devem ser discretizados em fragmentos nominais. Para isso, foi utilizando o *Tree-based Unsupervised Bin Estimator* (TUBE) proposto em SCHMIDBERGER; FRANK (2005) para a discretização dos valores em 10 fragmentos. Um primeiro problema surge em determinar a função de similaridade, já que a discretização dos valores promovido pelo algoritmo TUBE produz um agrupamento de valores. Caso a função de similaridade não seja capaz de discernir corretamente as similaridades, é possível que padrões dissimilares sejam normalizados em um mesmo grupo causando perdas de informações relevantes. Para contornar esse problema, diferentes funções podem ser concatenadas para a geração de um vetor de características mais informativo para ser utilizado pelo algoritmo SSAR. Foram utilizadas duas configurações de funções de similaridade nos experimentos do T3S. A primeira configuração, chamada T3S-(1SF), utiliza uma função de similaridade para a geração das *features*, resultando em um tamanho do vetor igual ao número de atributos da base de dados. A segunda configuração, chamada T3S-(2SF), utiliza duas funções de similaridade para produzir as *features*, resultando em um tamanho do vetor duas vezes maior que o número de atributos. As funções de similaridade Levenshtein (baseadas em caracteres) e Jaccard NGram (baseada em *tokens*) foram utilizadas para produzir o vetor de características.

Por fim, é empregado método de aprendizagem proposto por BELLARE et al. (2012), chamado de ALD. Uma implementação do método foi disponibilizada pelos autores. A precisão mínima, a ser alcançada pelo método ALD, foi definida com valor igual a 0,85 e o oráculo foi formado com até 100 pares, como sugerido no trabalho original. Os valores de limiares, utilizados para ajustar o número de pares manualmente rotulados, inclui os seguintes valores: $1x10^{-3}$, $1x10^{-4}$,...,$1x10^{-9}$. O conjunto de pares candidatos utilizado pelo ALD foi o mesmo produzido pela *Etapa de Ordenamento* e o vetor de características foi produzido a partir da função de similaridade Jaccard NGram, como nos métodos FS-Dedup-SVM e T3S-SVM.

5.4 Avaliação experimental do FS-Dedup

Esta seção apresenta os experimentos realizados com intuito de validar a metodologia FS-Dedup, proposta no Capítulo 3. Os experimentos foram divididos em três conjuntos. O primeiro conjunto de experimentos visa avaliar o comportamento da estratégia para a geração dos pares candidatos, proposta pela *Etapa de Ordenamento*. O segundo conjunto de experimentos tem como objetivo avaliar o processo de identificação da região crítica e a eficácia obtida pela metodologia FS-Dedup comparado com o Sig-Dedup (manualmente configurado). O algoritmo Sig-Dedup foi utilizado para fins de comparação dos resultados dos experimentos, descrevendo o comportamento obtido quando o algoritmo é idealmente calibrado através da definição manual de valores de limiares. Por fim, o último conjunto de experimentos tem como objetivo avaliar o esforço manual comparando o FS-Dedup com o método de aprendizagem ativa ALD.

5.4.1 Avaliação da Etapa de Ordenamento

O objetivo destes experimentos é avaliar a capacidade das estratégias propostas na *Etapa de Ordenamento* para a geração de pares candidatos. Em outras palavras, a heurística deve, primeiramente, ser capaz de maximizar a geração de pares duplicados e, em seguida, evitar a geração acentuada de pares não duplicados. Para isso, é necessário ajustar o *limiar inicial* para a configuração do processo de filtragem e blocagem do algoritmo Sig-Dedup.

Para a execução destes experimentos, foram utilizadas as bases de dados sintéticas *DsgenA, DsgenB e DsgenC* e as bases reais *DBLPxCiteSeer e IMDBxNetFlix*. Os passos executados para a realização dos principais experimentos (experimentos complementares são apresentados no decorrer desta subseção) podem ser resumidos da seguinte forma:

1. é coletada, de forma arbitrária, uma amostra contendo uma porcentagem dos registros da base de dados, ou seja, 1%, 5% e 10% do total de registros da base de dados;

2. são aplicados os filtros do algoritmo Sig-Dedup (*filtro de prefixo, sufixo, tamanho e posição*) sobre as amostras previamente criadas, variando o valor do *limiar de teste* de 0,2 até 0,9 (com passo fixo de 0,1).

3. após a geração dos pares candidatos por cada *limiar de teste*, é verificado qual é o primeiro valor de limiar que respeita o critério de parada (apresentado na Seção 3.1.1). Tal valor de limiar é definido como o *limiar inicial* e utilizado para a geração dos pares candidatos;

4. após a geração dos pares candidatos, é produzido um ordenamento (*ranking*) utilizando como critério o valor de similaridade de cada par.

Os gráficos da Figura 5.1 ilustram o número de pares gerados pelas bases de dados sintéticas e reais. Nos gráficos, o eixo Y representa o número de pares candidatos criados sobre o número de registros da amostra, chamado de P/T. Tal normalização foi adotada para compactar as informações e melhorar a visualização dos dados. O objetivo a ser alcançado é identificar um valor de P/T abaixo do valor de 1,0, ou seja, o número de pares candidatos deve ser menor que o número de registros da respectiva amostra, como formalizado na Definição 3.

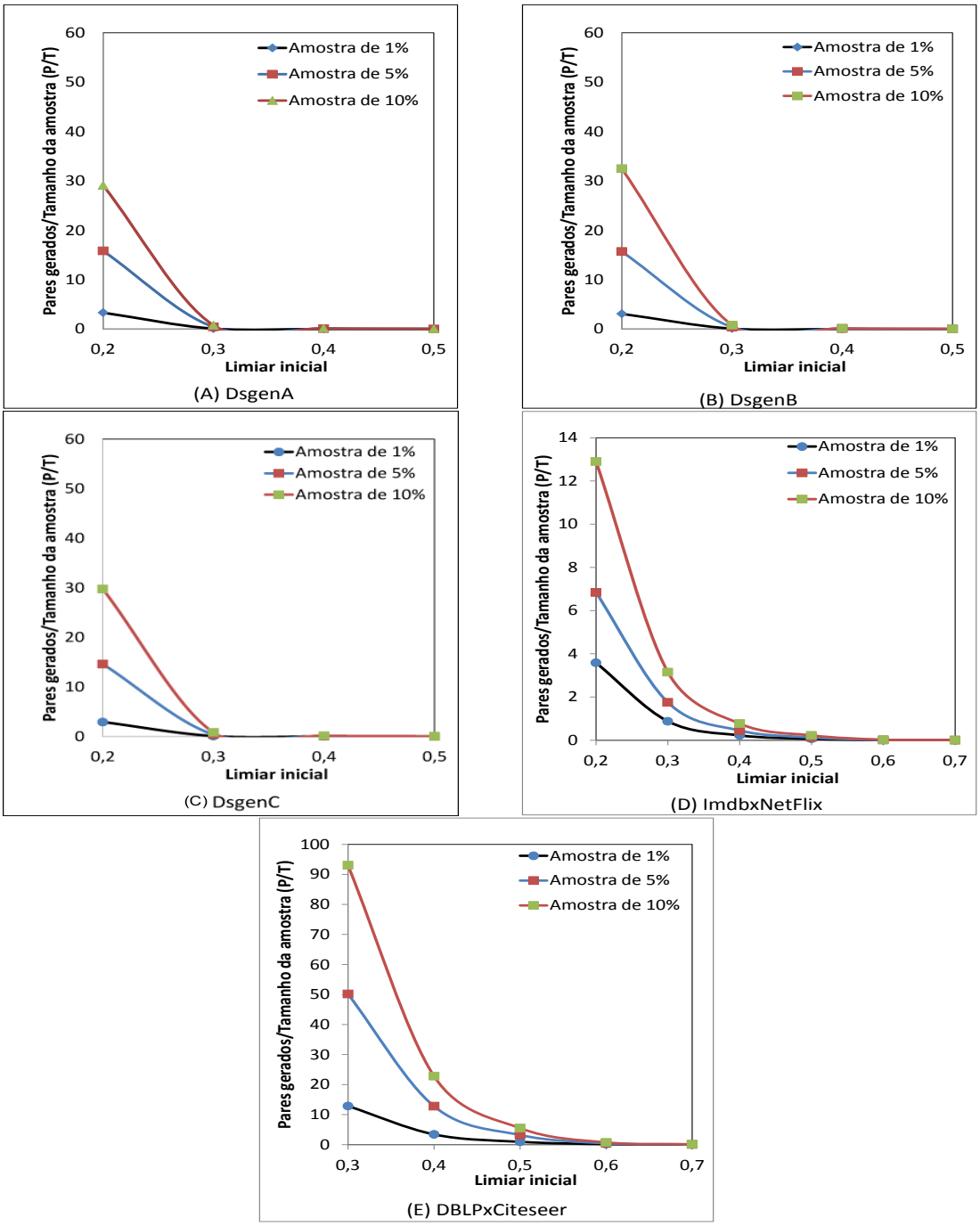


Figura 5.1: Comparação de diferentes valores de *limiar inicial* com as amostras de tamanho 1%, 5% e 10% nas bases de dados sintéticas e reais.

Nos gráficos, os dados sintéticos e reais produziram diferentes curvas dependendo das características dos seus conjuntos de dados. Tal comportamento é esperado, tendo em vista os diferentes padrões de duplicatas e níveis de ruídos de cada base de dados. Por exemplo, as bases DsgenA, DsgenB e DsgenC produziram um acentuado número de pares candidatos utilizando um *limiar de teste* igual ao valor 0,2, ou seja, um número substancial de termos frequentes foi indexado com tal valor de limiar. Na base de dados DsgenC, por exemplo, o *limiar de teste* com um valor de 0,2 e uma amostra de 10% produziram um P/T igual a 30 (cerca de 450.000 pares foram produzidos de uma amostra de 15.000

registros). Por outro lado, o *limiar de teste* 0,3 reduziu substancialmente o número de pares candidatos, produzindo um valor de P/T menor que 1,0 (cerca de 10.000 pares foram gerados de uma amostra de 15.000 registros). Como os gráficos (B) e (C) da Figura 5.1 ilustram, as bases de dados DsgenA e DsgenB apresentaram um comportamento similar à base de dados DsgenC. Dessa forma, o valor 0,3 foi qualificado para ser utilizado como o *limiar inicial* nas bases de dados sintéticas.

Salienta-se que a base de dados IMDBxNetFlix (gráfico da Figura 5.1-(D)) produziu um número reduzido de pares candidatos se comparado as outras bases de dados. Mais especificamente, a base de dados IMDBxNetFlix resultou em um valor de P/T de cerca de duas vezes menor que as bases sintéticas e sete vezes menor que a base DBLPxCiteSeer, em uma amostra de 10% da base de dados. Tal comportamento é explicado pelo fato da base de dados NetFlix ser composta de um número reduzido de registros (160.000 registros) se comparado a base de dados IMDB (cerca de 1 milhão registros), resultando em um baixo número de pares candidatos. No mesmo gráfico, pode-se observar que o *limiar de teste* com valor 0,3 e uma amostra de 1% da base de dados produziram um P/T inferior a 1,0 (cerca de 12.000 pares candidatos foram gerados de um total de 12.400 registros), qualificando tal valor para ser utilizado como o *limiar inicial*. No entanto, o mesmo valor de limiar de teste (0,3) quando aplicado sobre uma amostra de 5% e 10% produz um P/T com os valores de 2,0 e 3,8 (119.000 e 428.735 pares de um total de 68.000 e 136.000 registros, respectivamente). Tal comportamento é explicado pela amostra de 1% não ser suficientemente representativa para selecionar os termos frequentes e os pares duplicados. Assim, foi definido um tamanho de amostra de 5% de cada base de dados como valor padrão de tamanho de amostra para ser processada pelo Sig-Dedup na identificação do *limiar inicial*. Por fim, o limiar com valor 0,4 foi qualificado para ser utilizado como o *limiar inicial* devido ao seu P/T ser inferior ao valor 1,0, na base de dados IMDBxNetFlix.

Como o gráfico da Figura 5.1-(E) ilustra, a base de dados DBLPxCiteSeer resultou em um alto número de pares candidatos comparado as demais bases de dados. O valor de limiar 0,3 e uma amostra de 10% produziram um P/T igual a 90, ou seja, cerca de 26 milhões de pares candidatos foram gerados de uma amostra contendo 280.000 registros. Tal número de pares candidatos é resultado de um alto volume de registros e do acentuado número de caracteres contidos em cada registro (os registros da base de dados DBLPxCiteSeer são compostos de três vezes mais caracteres que os registros da base de dados IMDBxNetFlix). Em tal base de dados, somente o limiar 0,6 foi capaz de produzir um P/T inferior ao valor 1,0. De fato, as características da base de dados DBLPxCiteSeer forçam com que o algoritmo Sig-Dedup seja configurado de uma forma mais agressiva (com um limiar mais elevado) para evitar a indexação dos termos frequentes.

Com objetivo de comparar a qualidade dos pares gerados por cada um dos limiares *iniciais* selecionados, foi projetado um conjunto de experimentos adicional com intuito de avaliar a revocação ao processar todos os registros das bases de dados.

Como apresentado na Tabela 5.3, o limiar com valor 0,2, nas bases sintéticas, obteve praticamente todos os pares duplicados da base de dados (99%), mas com a desvantagem de um alto número de pares candidatos (mais de 40 milhões de pares foram gerados na base de dados DsgenC). Já o *limiar inicial* (com valor 0,3) produziu cerca de 19 vezes menos pares se comparado ao limiar 0,2, com uma revocação que se aproximou dos 98%. Os limiares com valores de 0,4 e 0,5 reduziram substancialmente o número de pares candidatos, mas com uma redução de 7% e 20% na revocação. Os limiares 0,4 e 0,5 impactaram diretamente na qualidade dos pares gerados removendo um alto número de pares duplicados. De fato, o valor do *limiar inicial* (0,3) foi capaz de reduzir o número de

Tabela 5.3: Revocação e número de pares candidatos produzidos nas bases de dados sintéticas por diferentes limiares.

Base de Dados	Limiar	Revocação	#Pares
(A)DsgenA	0,2	99,08	35.695.217
	0,3	**97,94**	**1.238.129**
	0,4	92,22	331.938
	0,5	80,26	65.158
(B)DsgenB	0,2	99,00	38.833.591
	0,3	**97,6**	**2.066.238**
	0,4	92,59	393.812
	0,5	80,95	100.758
(C) DsgenC	0,2	0,990	43.629.641
	0,3	**0,979**	**2.776.461**
	0,4	0,935	263.872
	0,5	0,833	90.292

Tabela 5.4: Revocação e número de pares candidatos produzidos nas bases de dados reais (IMDBxNetFlix e DBLPxCiteSeer) por diferentes limiares.

Base de Dados	Limiar	Revocação	#Pares
(A) IMDBxNetFlix	0,3	1,00	33.730.220
	0,4	**1,00**	**8.381.906**
	0,5	0,99	2.340.065
	0,6	0,94	243.109
(B) DBLPxCiteseer	0,5	1,00	69.313.296
	0,6	**0,99**	**13.180.090**
	0,7	0,9	2.122.995
	0,8	0,7	317.818

pares candidatos sem depreciar substancialmente a qualidade dos pares candidatos.

A Tabela 5.4-(A) apresenta o valor de revocação e o número de pares candidatos produzidos pelos *limiares de teste* na base de dados IMDBxNetflix. O valor de limiar 0,3, considerado ideal nas bases de dados sintéticas, obteve todos os pares duplicados (100%), mas com a desvantagem de produzir mais de 33 milhões de pares candidatos. O valor definido como *limiar inicial* (0,4) obteve todos os pares duplicados com um número de pares candidatos de cerca de oito milhões (24 milhões a menos que o limiar 0,3). Em outras palavras, os 24 milhões de pares produzidos a mais pelo limiar 0,3 acrescentaram um custo substancial de processamento sem produzir melhoras na qualidade da deduplicação. Já o limiar 0,5 foi capaz de reduzir em 3,6 vezes o número de pares candidatos em relação ao *limiar inicial* (0,4) sem perdas substanciais no número de pares duplicados. Note-se que os limiares 0,4 e 0,5 foram capazes de maximizar o número de pares duplicados sem uma geração excessiva de pares duplicados candidatos, assim ambos estão aptos para serem usados como valores do *limiar inicial*. Para evitar que o *limiar inicial* descarte pares duplicados, é selecionado o primeiro limiar capaz de promover o descarte dos termos mais frequentes, nesse caso, o limiar 0,4.

Na base de dados DBLPxCiteSeer, como ilustrado na Tabela 5.4-(B), o *limiar de teste* com valor 0,5 produziu cerca de 70 milhões de pares candidatos com uma revocação de 100%. Já o valor de *limiar inicial* (0,6) reduziu em mais de cinco vezes o número de pares (13 milhões de pares) sem causar impactos significativos no número dos pares duplicados (uma revocação de 99%). Já o limiar 0,7 reduziu em cerca de dois milhões o número de pares duplicados com uma perda de cerca de 10% no número de pares duplicados, inviabilizando sua utilização devido à reduzida qualidade na geração dos pares candidatos.

Por fim, pode-se concluir que a estratégia proposta para a geração dos pares candidatos

é capaz de identificar o valor do *limiar inicial* com uma revocação de mais 98% (até 100% em uma das bases de dados) com uma geração controlada de pares candidatos, sem a necessidade da intervenção do usuário. Por exemplo, o produto cartesiano entre as bases de dados DBLP e CiteSeer resulta em um total de $1,6 * 10^{12}$ pares candidatos, enquanto o número de pares candidatos produzido pela Etapa de Ordenamento foi de cerca de $1,3 * 10^{6}$ com uma perda de somente 1% dos pares duplicados. De fato, é possível evitar uma geração excessiva no número de pares candidatos, sem causar perdas na qualidade da deduplicação.

5.4.2 Avaliação da eficácia do FS-Dedup

Neste conjunto de experimentos é avaliada a eficácia do FS-Dedup-NGram e do FS-Dedup-SVM na identificação da região crítica, bem como, a demanda de rotulação manual. Nesta seção, é adotado o algoritmo Sig-Dedup configurado com um valor de limiar ótimo (manualmente identificado) como *baseline*. Primeiramente, é discutido um conjunto de experimentos nas bases de dados sintéticas e, em seguida, são apresentados os experimentos nas bases de dados reais.

Os passos executados para a realização dos experimentos podem ser resumidos da seguinte forma:

1. o *ranking* de pares candidatos, criado na etapa anterior (Etapa de Ordenamento), é fragmentado em 10 *faixas*. Dentro de cada *faixa*, é selecionado uma amostra aleatória contendo 10, 50, 100, 500 ou 1000 pares;

2. amostras que foram consideradas relevantes são rotuladas para a identificação da região crítica. Nesses experimentos, foram utilizados pares previamente rotulados para simular a rotulação do usuário;

3. os pares, considerados críticos, são classificados pelos métodos FS-Dedup-NGram e FS-Dedup-SVM.

Para simplificar, o número de pares selecionados aleatoriamente dentro de cada *faixa* é chamado de *tamanho da faixa*. Nas Tabelas 5.6 e 5.9, são descritos detalhes sobre a eficácia da deduplicação nos métodos FS-Dedup-NGram e FS-Dedup-SVM nas bases de dados sintéticas e nas bases de dados reais, respectivamente. A primeira coluna das tabelas define o *tamanho de faixa*. As médias aritméticas e os desvios padrão da *precisão*, *revocação* e *F1* são descritos nas próximas colunas para cada um dos métodos. Por fim, junto à última coluna, é apresentado o teste T para comparar estatisticamente os resultados encontrados. O teste T é ilustrado pelos símbolos $\uparrow$, $\downarrow$ e o que representam, respectivamente, um ganho estatisticamente significativo do FS-Dedup-NGram sobre o FS-Dedup-SVM, um ganho estatisticamente significativo do FS-Dedup-SVM sobre o FS-Dedup-NGram e um empate estatístico entre os métodos.

Com objetivo de complementar as informações presentes nas tabelas anteriores (Tabelas 5.6 e 5.9) são apresentadas as Tabelas 5.7 e 5.10 para a avaliação do esforço manual do usuário. Nessas tabelas, na primeira coluna é novamente apresentado o *tamanho das faixas*. Em seguida, nas colunas α e β, são detalhadas as médias aritméticas e os respectivos desvios padrão dos limites de fronteira da região crítica. Por fim, é apresentado o número médio de pares manualmente rotulados em cada *tamanho de faixa*.

Tabela 5.5: F1 obtido quando é variado o valor do limiar nas bases de dados sintéticas.

Base de dados \ Limiar	0,3	0,4	0,5	0,6
DsgenA	0,92	**0,98**	0,86	0,69
DsgenB	0,89	**0,976**	0,87	0,66
DsgenC	0,92	**0,975**	0,86	0,694

5.4.2.1 Experimentos nas bases de dados sintéticas

A seguir, são apresentados os experimentos para avaliar a eficácia e o esforço manual da metodologia FS-Dedup nas bases de dados sintéticas. O objetivo da metodologia FS-Dedup é obter um valor de eficácia competitivo em relação ao Sig-Dedup (otimamente configurado) requisitando somente a intervenção a partir de pares manualmente rotulados.

Para fins de comparação, primeiramente é necessário identificar a configuração ideal do algoritmo Sig-Dedup (ou seja, o valor de limiar ideal) que resulte na maximização da eficácia da deduplicação. Para isso, cada uma das bases de dados foi avaliada com diferentes valores de limiares buscando maximizar a métrica F1, como reportado na Tabela 5.5. Pode-se observar que foi obtida uma eficácia praticamente idêntica nas bases analisadas devido ao semelhante nível de ruído nos dados. O limiar ideal do Sig-Dedup foi identificado com valor igual a 0,4 com um valor de F1 em torno de 98%. Observe que o Sig-Dedup reduz a eficácia em mais de 11% quando utilizado um valor de limiar de limiar 0,5, ilustrando a importante tarefa de se identificar o valor ideal de limiar.

Nos gráficos da Figura 5.2 é ilustrada uma comparação da eficácia dos métodos FS-Dedup-SVM, FS-Dedup-NGram e do algoritmo Sig-Dedup (calibrado com o valor de limiar ideal) com diferentes tamanhos de *faixas* (10, 50, 100, 500 e 1.000 pares). O objetivo desse experimento é avaliar a capacidade dos métodos FS-Dedup-SVM e FS-Dedup-NGram em obter a máxima eficácia oferecida pelo algoritmo Sig-Dedup.

Nos gráficos da Figura 5.2, o FS-Dedup-NGram foi capaz de obter a máxima eficácia em relação ao Sig-Dedup nas três bases de dados analisadas. Observe que para o FS-Dedup-NGram atingir o valor máximo de F1 da base DsgenA, foi necessário rotular um número maior de pares, se comparado as bases DsgenB e DsgenC. As bases de dados DsgenA e DsgenB dependem de um *tamanho de faixa* de cerca de 500 pares para obter o valor de F1 idealmente alcançado pelo Sig-Dedup. Já a base de dados DsgenC alcança a máxima eficácia com um *tamanho de faixa* de somente 100 pares manualmente rotulados. Tal comportamento pode ser explicado pela presença de um número superior de pares duplicados nas bases de dados DsgenB e DsgenC, facilitando a tarefa de criação de uma amostra balanceada de pares duplicados e não duplicados.

O método FS-Dedup-SVM produz um valor de F1 pouco competitivo em relação ao Sig-Dedup. O FS-Dedup-SVM somente produz um valor de F1 competitivo em relação ao FS-Dedup-NGram, com um *tamanho da faixa* reduzido (cerca de 10 pares). Quando o *tamanho de faixa* é ampliado para 500 pares, o FS-Dedup-NGram atinge o valor máximo de F1 alcançado pelo Sig-Dedup, enquanto o FS-Dedup-SVM apresenta uma baixa melhora, claramente insuficiente para alcançar o valor ideal do Sig-Dedup. A baixa qualidade do FS-Dedup-SVM é explicada a partir do padrão de ruído especificamente encontrado nas bases de dados sintéticas. Isto significa que anomalias nos dados, facilmente capturadas pelo FS-Dedup-NGram, demonstram ser altamente desafiadoras para o método de classificação FS-Dedup-SVM, tais como: a inversão de valores de campos; ausência de valores e pequenos erros em caracteres.

Em mais detalhes, na Tabela 5.6 pode ser observado que o FS-Dedup-NGram é mais

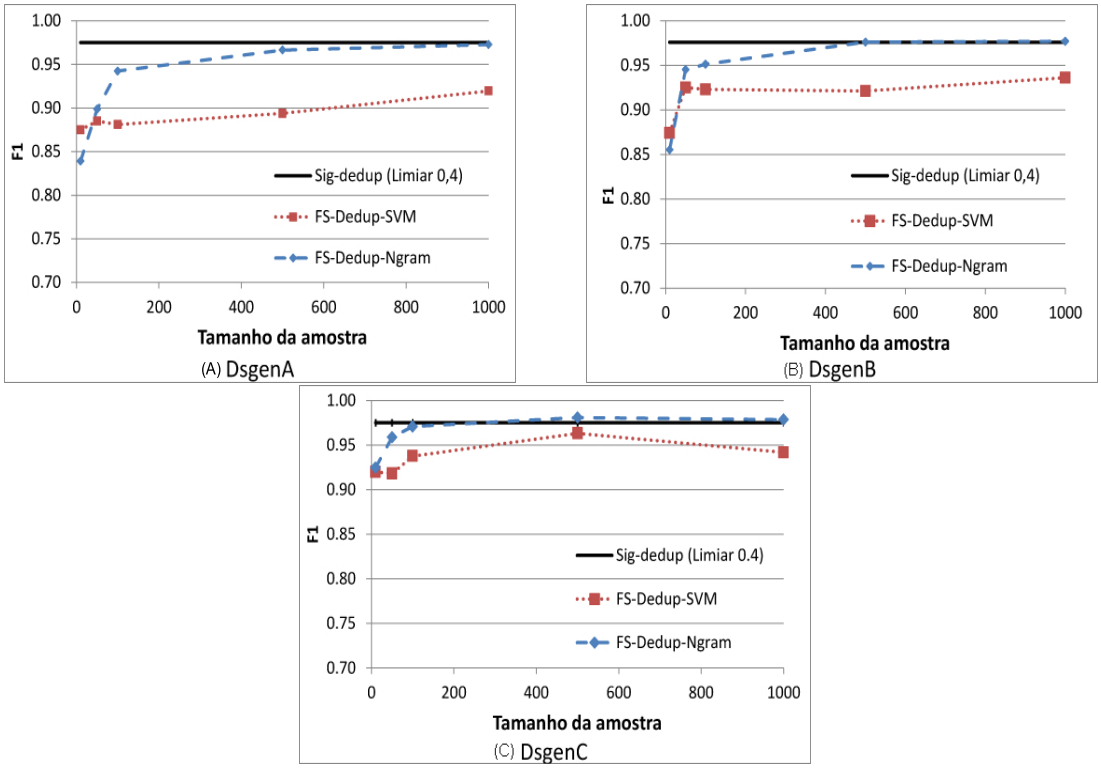


Figura 5.2: Comparação dos métodos FS-Dedup-NGram, FS-Dedup-SVM e o Sig-Dedup (manualmente configurado com o limiar ideal) nas bases sintéticas DsgenA, DsgenB e DsgenC.

eficaz que o FS-Dedup-SVM, nas bases de dados sintéticas, em basicamente todos os tamanhos de *faixas* (com exceção das *faixas* compostas por 10 pares). Considerando as 15 combinações de experimentos executadas (com três bases de dados e cinco variações de *tamanho de faixas*) e os valores de F1, o FS-Dedup-NGram alcança melhorias nos resultados em 11 experimentos, empate estatístico em três experimentos e perde em somente um caso. Os resultados também ilustram que o FS-Dedup-NGram é capaz de alcançar um valor de F1 igual ou superior a 94% em todos os conjuntos de dados sintéticos com um *tamanho de faixa* de somente 100 pares. Quando o *tamanho de faixa* é ampliado para 1.000 pares, o FS-Dedup-NGram aprimora a revocação (cerca de 5%, 4%, 2% nas bases de dados DsgenA, DsgenB e DsgenC, respectivamente) mantendo a precisão próxima ao valor máximo (100%). A baixa qualidade do FS-Dedup-SVM é explicada pela queda substancial no valor da precisão ao se ampliar o *tamanho da faixa*, ou seja, o aumento no número de pares rotulados não é suficiente para manter os valores de precisão encontrados nos tamanhos de *faixas* iniciais.

A Tabela 5.7 detalha os valores de α e β e o esforço manual de cada *tamanho de faixa* nas bases de dados sintéticas. Pode-se observar que o aumento no número de pares rotulados causa uma redução do valor do α, resultando em uma ampliação do número de pares candidatos pertencentes à região crítica. Tal ampliação é importante para melhorar a revocação, mas exige um modelo de classificação mais específico para a correta identificação das duplicatas. Como descrito anteriormente, a precisão do FS-Dedup-SVM apresenta uma expressiva redução (por exemplo, cerca de 10% na base de dados DsgenA com um *tamanho de faixa* de 1000 pares) quando mais pares candidatos são adicionados à região

Tabela 5.6: Comparação da eficácia dos métodos FS-Dedup-SVM e o FS-Dedup-NGram nas bases de dados sintéticas.

Base de dados	T	FS-Dedup-SVM			FS-Dedup-NGram		
		Precisão (σ)	Revocação(σ)	F1(σ)	Precisão(σ)	Revocação(σ)	F1(σ)
DsgenA	10	0,99±(0,01)	0,79±(0,10)	**0,87±(0,07)**	0,96±(0,11)	0,77±(0,12)	0,84±(0,09)↓
	50	0,98±(0,03)	0,82±(0,11)	0,89±(0,07)	100,0±(0,00)	0,82±(0,11)	0,90±(0,07) *o*
	100	0,90±(0,15)	0,89±(0,04)	0,88±(0,09)	0,99±(0,01)	0,90±(0,04)	**0,94±(0,03)**↑
	500	0,87±(0,09)	0,92±(0,04)	0,89±(0,04)	0,99±(0,00)	0,94±(0,04)	**0,97±(0,02)**↑
	1000	0,90±(0,07)	0,95±(0,03)	0,92±(0,04)	0,99±(0,00)	0,95±(0,03)	**0,97±(0,02)**↑
DsgenB	10	0,98±(0,01)	0,79±(0,10)	0,87±(0,06)	0,95±(0,05)	0,79±(0,14)	0,86±(0,08) *o*
	50	0,97±(0,05)	0,88±(0,05)	0,92±(0,03)	1,00±(0,00)	0,90±(0,05)	**0,95±(0,03)**↑
	100	0,94±(0,07)	0,91±(0,05)	0,92±(0,03)	0,99±(0,02)	0,92±(0,05)	**0,95±(0,03)**↑
	500	0,90±(0,06)	0,95±(0,01)	0,92±(0,03)	0,99±(0,01)	0,96±(0,00)	**0,98±(0,00)**↑
	1000	0,92±(0,04)	0,95±(0,00)	0,94±(0,02)	100,0±(0,00)	0,96±(0,00)	**0,98±(0,00)**↑
DsgenC	10	0,99±(0,11)	0,86±(0,04)	0,92±(0,06)	0,98±(0,02)	0,87±(0,04)	0,92±(0,02)*o*
	50	0,93±(0,12)	0,92±(0,05)	0,92±(0,07)	0,99±(0,02)	0,93±(0,05)	**0,96±(0,03)**↑
	100	0,94±(0,07)	0,94±(0,03)	0,94±(0,04)	0,99±(0,01)	0,95±(0,03)	**0,97±(0,02)**↑
	500	0,97±(0,03)	0,96±(0,01)	0,96±(0,02)	0,99±(0,01)	0,97±(0,00)	**0,98±(0,00)**↑
	1000	0,92±(0,04)	0,96±(0,00)	0,94±(0,02)	0,99±(0,01)	0,97±(0,00)	**0,98±(0,01)**↑

crítica. Isso demonstra que o método FS-Dedup-SVM é pouco efetivo na classificação dos pares presentes nas *faixas* compostas por pares com baixos valores de similaridade devido à diversidade de padrões de pares candidatos. Por exemplo, o único caso em que o FS-Dedup-SVM supera o FS-Dedup-NGram pode ser observado quando a região crítica é composta por um pequeno número de pares candidatos (α igual a 0,34, β igual a 0,5 e com um *tamanho de faixa* de 10 pares).

Observe que na Tabela 5.7 os valores de α são reduzidos quando um número maior de pares é rotulado, enquanto os valores de β se mantêm praticamente constantes. Tal comportamento é explicado pelo fato de que as *faixas* mais baixas (10, 50 e 100 pares) são compostas por um acentuado número de pares não duplicados, dificultando a identificação precisa do par mínimo verdadeiro (PV) (responsável pela definição do α). Com o aumento no tamanho das *faixas* para 500 ou 1000 pares, a região crítica pode ser identificada mais precisamente devido ao maior volume de pares manualmente rotulados. Dessa forma, surge a lacuna de como identificar os pares candidatos mais relevantes e não redundantes para reduzir o esforço manual. Tal lacuna é diretamente endereçada pelo Capítulo 4 que visa selecionar para a rotulação os pares que acrescentem ganho de informatividade para o conjunto de treinamento.

Em geral, nas bases sintéticas somente as amostras entre as *faixas* [0,1;0,2] até [0,5;0,6] foram manualmente rotuladas pelo usuário e todos os pares com similaridade acima de 0,5 são considerados duplicatas. Em outras palavras, somente cinco amostras de *faixas* foram rotuladas pelo usuário, evitando que pares que não acrescentam ganho de informação ao processo de identificação da região crítica sejam manualmente rotulados.

5.4.2.2 Experimentos nas bases de dados reais

A seguir, é avaliada a eficácia e o esforço manual da metodologia FS-Dedup nas bases de dados reais IMDBxNetFlix e DBLPxCiteSeer. Primeiramente, é reportado um experimento com objetivo de identificar o limiar ideal no algoritmo Sig-Dedup. Nas bases de dados reais, o limiar resultou em um comportamento específico para cada conjunto de dados, como apresentado na Tabela 5.8. O limiar ideal foi mais elevado na base de dados

Tabela 5.7: Comparação do esforço manual do usuário nas bases de dados sintéticas.

Base de dados	T	$\alpha(\sigma)$	$\beta(\sigma)$	Número de pares rotulados
DsgenA	10	0,34±(0,05)	0,50±(0,00)	36
	50	0,31±(0,05)	0,50±(0,00)	194
	100	0,27±(0,04)	0,50±(0,00)	430
	500	0,20±(0,06)	0,50±(0,00)	2500
	1000	0,18±(0,06)	0,50±(0,00)	5222
DsgenB	10	0,32±(0,06)	0,43±(0,05)	32
	50	0,26±(0,05)	0,50±(0,00)	220
	100	0,24±(0,06)	0,50±(0,00)	460
	500	0,16±(0,05)	0,51±(0,03)	2750
	1000	0,20±(0,00)	0,50±(0,00)	5000
DsgenC	10	0,29±(0,04)	0,42±(0,05)	33
	50	0,22±(0,06)	0,49±(0,03)	235
	100	0,20±(0,05)	0,50±(0,00)	500
	500	0,15±(0,05)	0,50±(0,00)	2750
	1000	0,11±(0,03)	0,50±(0,00)	5900

Tabela 5.8: Eficácia dos diferentes valores de limiares definidos manualmente nas bases de dados reais.

Base de dados \ Limiar	0,4	0,5	0,6	0,7	0,8
IMDBxNetflix	0,694	0,83	0,914	**0,923**	0,896
DBLPxCiteseer	0,795	0,902	**0,921**	0,876	0,790

IMDBxNetflix (com valor 0,7) se comparado à base de dados DBLPxCiteSeer (com valor 0,6) com um valor semelhante da medida F1 de 92%. Tal valor de limiar relativamente alto é explicado pelo nível de ruído da base de dados IMDBxNetflix ser inferior aos demais conjuntos de dados. Os registros duplicados na base de dados IMDBxNetflix são mais similares que os pares duplicados presentes nas outras bases de dados. O objetivo da metodologia FS-Dedup é alcançar o valor de F1 idealmente identificado pelo Sig-Dedup, requisitando somente pares manualmente rotulados.

Nos gráficos da Figura 5.3, é comparada a eficácia do Sig-Dedup (calibrado com a configuração ideal) em relação ao FS-Dedup-SVM e FS-Dedup-NGram, variando o *tamanho de faixa* em 10, 50 e 100 pares. Na base de dados DBLPxCiteSeer, os métodos

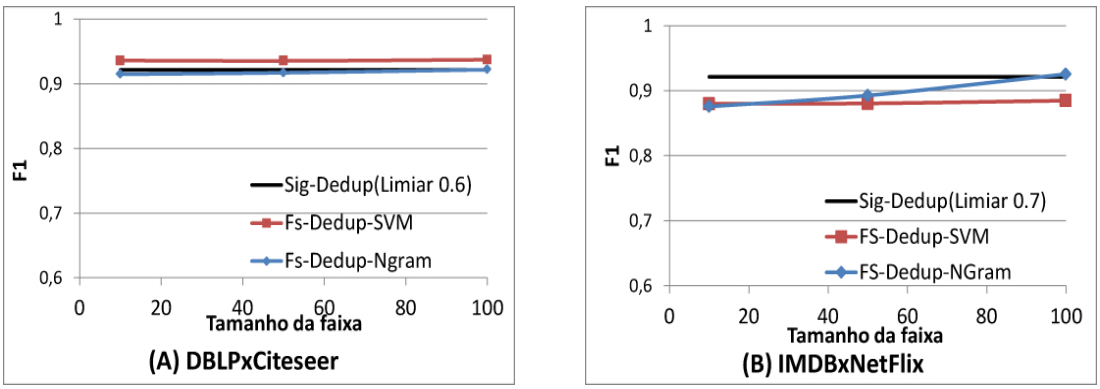


Figura 5.3: Comparação dos métodos FS-Dedup-NGram, FS-Dedup-SVM e o Sig-Dedup (manualmente configurado com o limiar ideal) nas bases reais.

FS-Dedup-NGram e FS-Dedup-SVM demostraram alcançar o valor máximo de F1 (ou próximo ao máximo) com *tamanho de faixa* de 10 pares. Diferente das bases de dados sintéticas, na DBLPxCiteSeer, o método FS-Dedup-SVM superou estatisticamente o Sig-Dedup e o FS-Dedup-NGram em todos os *tamanhos de faixas* em cerca de 2% em relação ao F1. Tal comportamento é explicado pelo padrão específico de duplicatas presentes em tal conjunto de dados. A base de dados CiteSeer é composta por dados automaticamente extraídos da Web, resultando em registros com informações fragmentadas ou incompletas. Além disso, o classificador SVM é capaz de identificar, a partir de um pequeno conjunto de treinamento, quais atributos apresentam informações mais relevantes para o processo de classificação.

Na base de dados IMDBxNetFlix, ilustrada na Figura 5.3-(B), pode ser observado que o FS-Dedup-SVM obteve um valor de F1 constante em todos os *tamanhos de faixas*. Nessa base de dados, o FS-Dedup-NGram obteve uma melhora no F1 quando o *tamanho de faixa* é estendido para 100 pares equiparando ao valor ideal de F1 do Sig-Dedup.

Tabela 5.9: Detalhamento da eficácia dos métodos FS-Dedup-SVM e FS-Dedup-NGram nas bases de dados reais.

Base de dados	t	FS-Dedup-SVM			FS-Dedup-NGram		
		Precisão (σ)	Revocação(σ)	F1(σ)	Precisão(σ)	Revocação(σ)	F1(σ)
IMDBx NetFlix	10	0,81±(0,07)	0,97±(.01)	0,88±(0,04)	0,79±(0,08)	0,99±(0,02)	0,88±(0,04) o
	50	0,80±(0,03)	0,98±(0,01)	0,88±(0,01)	0,81±(0,06)	0,99±(0,00)	**0,89±(0,04)**↑
	100	0,80±(0,00)	0,99±(0,00)	0,88±(0,00)	0,98±(0,00)	0,93±(0,00)	**0,92±(0,00)**↑
DBLPx CiteSeer	10	0,91±(0,01)	0,96 ±(0,01)	**0,94±(0,01)**	0,86±(0,02)	0,97±(0,01)	0,91±(0,01)↓
	50	0,91±(0,01)	0,96±(0,01)	**0,94±(0,01)**	0,88±(0,03)	0,95±(0,02)	0,92±(0,01)↓
	100	0,92±(0,00)	0,95±(0,00)	**0,94±(0,00)**	0,90±(0,00)	0,94±(0,00)	0,92±(0,00)↓

Tabela 5.10: Detalhamento do esforço manual do usuário nas bases de dados reais.

Base de dados	T	$\alpha(\sigma)$	$\beta(\sigma)$	Número de pares rotulados
IMDBx NetFlix	10	0,42±(0,08)	0,65±(0,00)	47
	50	0,32±(0,08)	0,76 ±(0,00)	330
	100	0,30±(0,00)	0,90±(0,00)	796
DBLPx CiteSeer	10	0,47±(0,06)	0,72±(0,00)	45
	50	0,43±(0,00)	0,80±(0,00)	285
	100	0,40±(0,00)	0,80±(0,00)	581

Um detalhamento dos resultados descritos anteriormente é apresentado nas Tabelas 5.9 e 5.10. Novamente, na base DBLPxCiteSeer pode-se observar que o aumento do número de pares rotulados não reflete em ganhos em termos da eficácia em ambos os métodos. A região crítica é identificada entre os limiares 0,4 a 0,8 e menos que 50 pares manualmente rotulados são suficientes para configurar idealmente o FS-Dedup. Na base de dados IMDBxNetFlix, os métodos FS-Dedup-(SVM e NGram) obtiveram um valor de F1 de 0,88, requisitando somente a rotulação de cerca de 47 pares. Com o aumento no *tamanho da faixa* para 100 pares, o método FS-Dedup-NGram obtém o valor máximo de F1 em relação ao algoritmo Sig-Dedup (92%), similar ao comportamento nas bases de dados sintéticas. Já o método FS-Dedup-SVM não obteve nenhum ganho em termos de F1 com o aumento no número de pares rotulados, ou seja, o método FS-Dedup-SVM alcançou um F1 constante de 0,88 em todos os *tamanhos de faixas*. Os resultados superiores alcançados pelo FS-Dedup-NGram podem ser explicados por uma possível ausência

de representatividade no conjunto de treinamento para o efetivo treinamento do algoritmo SVM.

Como um resumo dos experimentos apresentados nesta seção, pode-se concluir que a metodologia FS-Dedup foi capaz de identificar a configuração ótima em relação ao algoritmo Sig-Dedup sem a necessidade de limiares manualmente configurados. Os experimentos demostraram que o FS-Dedup-NGram atingiu a configuração ideal em todas as bases de dados analisadas comparado ao algoritmo Sig-Dedup, manualmente configurado. Já o FS-Dedup-SVM apresentou um comportamento instável, especialmente nas bases de dados sintéticas. A exceção foi a base de dados DBLPxCiteSeer, na qual o FS-Dedup-SVM obteve resultados superiores devido à presença de atributos com diferentes pesos ou importâncias no processo de classificação. Assim, é possível concluir que os métodos FS-Dedup-SVM e FS-Dedup-NGram são complementares e dependentes das características de cada conjunto de dados para a obtenção de uma alta qualidade da deduplicação.

5.4.3 Comparação do esforço manual

A seguir, é reportado um conjunto de experimentos com objetivo de comparar o esforço do usuário e a eficácia da metodologia FS-Dedup com o método proposto por BELLARE et al. (2012), chamado de ALD, representando o estado da arte da aprendizagem ativa para deduplicação (como apresentado na Seção 2.3.2.2). Para permitir a comparação e a repetição dos experimentos, são reportados somente experimentos utilizando bases de dados sintéticas nesta seção. Nas bases de dados reais, os conjuntos de pares manualmente rotulados utilizam o conceito de *faixas* para a seleção de amostras balanceadas. Dessa forma, utilizar tal conjunto rotulado para avaliar o *baseline* promoveria uma injusta vantagem, visto que a tarefa mais desafiadora do *baseline* é selecionar pares informativos a partir do conjunto completo de pares candidatos. Além disso, devido à necessidade de repetição dos experimentos, torna-se inviável rotular manualmente cada amostra de treinamento selecionada pelo ALD.

Nos gráficos da Figura 5.4 são apresentados os resultados comparando o ALD com o FS-Dedup-(SVM e NGram) utilizando-se da métrica F1 e do número de pares rotulados. Nos gráficos, as amostras iniciais (com um número de pares 36, 32 e 33 para as bases DsgenA, DsgenB e DsgenC, respectivamente) selecionadas pelos métodos FS-Dedup-(SVM e NGram) produziram bons valores de eficácia se comparado ao *baseline* em todos os conjuntos de dados. O FS-Dedup-NGram apresentou melhorias substanciais quando o número de pares rotulados é ampliado até atingir o ponto de estabilização, em torno dos 500 pares. O FS-Dedup-SVM não apresentou melhorias significativas com o aumento no número de pares manualmente rotulados, demonstrando ser menos eficaz comparado ao FS-Dedup-NGram em relação a métrica F1.

O *baseline* ALD, que sempre inicia com uma amostra manualmente rotulada contendo até 100 pares para configurar o "oráculo" e o classificador, alcança uma eficácia instável nas bases de dados DsgenA e DsgenB. Nas mesmas bases de dados, ALD somente se aproxima do FS-Dedup-NGram após o conjunto de treinamento ser composto por cerca de 2.000 e 700 pares, respectivamente. Na base de dados DsgenC, os métodos alcançam basicamente a mesma eficácia, ou seja, a abundância de pares favorece a seleção de pares altamente informativos em todos os métodos avaliados. De fato, o conjunto de treinamento inicial selecionado pelos métodos FS-Dedup-NGram e FS-Dedup-SVM é *3,8, 4,7 e 7,5* vezes menor que o conjunto de treinamento utilizado pelo *baseline* sem uma variação estatisticamente significava nas bases de dados DsgenA, DsgenB e DsgenC,

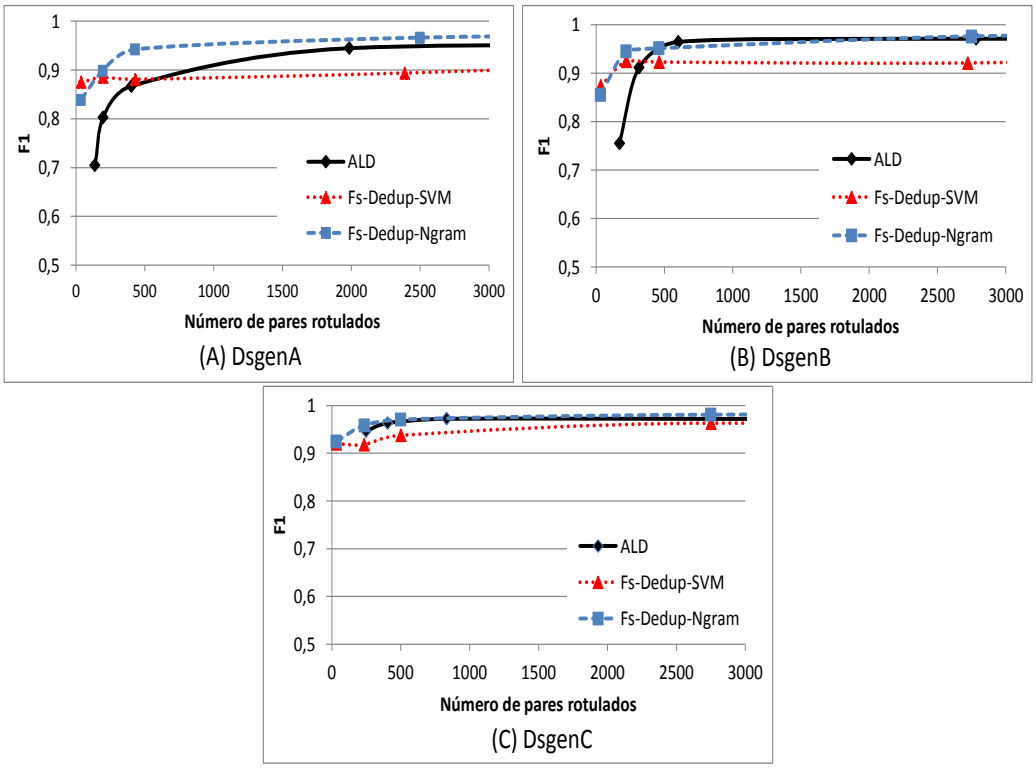


Figura 5.4: Comparação da eficiência e do esforço manual da metodologia FS-Dedup-(NGram e SVM) em relação ao *baseline* ALD.

respectivamente.

Por fim, esse conjunto de experimentos demostrou que a metodologia FS-Dedup é capaz de reduzir substancialmente o esforço manual do processo de deduplicação com uma qualidade competitiva quando comparado a um recente método de aprendizagem ativa para deduplicação. O resultado do FS-Dedup é explicado devido à combinação da estratégia de seleção das amostras de *faixas* relevantes para a correta identificação das fronteiras da região crítica.

5.5 Avaliação experimental da abordagem T3S

Nesta seção, é avaliada a capacidade da abordagem T3S de reduzir o número de pares manualmente rotulados sem perdas na qualidade da deduplicação. Os experimentos foram divididos em três conjuntos: (*i*) avaliação do primeiro passo da abordagem T3S, visando comparar a seleção de amostras utilizando o conceito de *faixas* com uma abordagem aleatória de coleta de pares; (*ii*) comparação do esforço manual e da eficácia da abordagem T3S em relação ao FS-Dedup; e (*iii*) avaliação da redução do esforço manual do usuário em relação a um recente método de aprendizagem ativa.

5.5.1 Avaliação do primeiro passo da abordagem T3S

Os experimentos, descritos a seguir, têm como objetivo comparar a estratégia de seleção em *faixas* (primeiro passo da abordagem T3S) com a estratégia de seleção aleatória de pares no conjunto de pares candidatos (chamada de T3S-Aleatória). A intuição é avaliar a importância da amostragem em *faixas* para a criação de um conjunto de pares a serem

processados pelo segundo passo da abordagem T3S.

As duas estratégias (primeiro passo do T3S e a T3S-Aleatória) foram comparadas mantendo em comum as Etapa de ordenamento e a Etapa de Classificação para possibilitar uma comparação direta do processo de seleção dos pares. É importante salientar que somente são reportados experimentos utilizando o classificador SVM (T3S-SVM) devido à estratégia T3S-Aleatória ser incapaz de selecionar pares em número suficiente para identificar o *limiar NGram*.

Como discutido na Seção 4.1, o algoritmo de aprendizagem ativa SSAR demanda de um inviável consumo computacional para a execução em grandes conjuntos de pares. Assim, para a viabilidade desta experimentação, foram produzidas amostras aleatórias do conjunto de pares não rotulados. Em mais detalhes, os passos da execução dos experimentos da abordagem T3S-Aleatória são descritos a seguir:

1. pares candidatos são gerados utilizando a Etapa de Ordenamento (como apresentado na Seção 3.1.1);

2. na T3S-Aleatória, são coletadas aleatoriamente 10 amostras cada uma com 10, 50, 100, 500 ou 1000 pares no conjunto de pares não rotulados. Já o T3S promove a seleção dos pares a partir da fragmentação do *ranking* em *faixas*, com intuito de evitar o desbalanceamento;

3. o conjunto de pares selecionado pela abordagem T3S-Aleatória é utilizado para identificar a região crítica e configurar o classificador SVM para a identificação dos pares duplicados.

Os gráficos da Figura 5.5 apresentam a comparação da abordagem T3S-SVM (um detalhamento dos passos do experimento do método T3S-SVM são apresentados a seguir, na Subseção 5.5.2) com a abordagem T3S-Aleatória nas bases de dados sintéticas. Pode-se observar uma baixa eficácia da abordagem T3S-Aleatória em todos os conjuntos de dados sintéticos em relação ao T3S-SVM. Tal comportamento é explicado pela ineficiência na seleção de um conjunto de treinamento representativo. Por exemplo, a amostra de pares candidatos produzida pela abordagem aleatória é composta, na maioria dos casos, por um baixo número de pares duplicados (menos de cinco pares duplicados) e um acentuado número de pares não duplicados. A (quase) ausência de pares duplicados resulta em uma configuração inapropriada do método de classificação, que depende de um conjunto de treinamento representativo.

Mais especificamente, o T3S-SVM obteve um valor médio de F1 de 82%, 84% e 90% utilizando a amostra inicial de pares (66, 64, 62 pares rotulados), nas bases DsgenA, DsgenB e DsgenC, respectivamente. À medida que pares são adicionados ao conjunto de treinamento (331, 336, 345 pares rotulados), a eficácia converge para valores similares aos encontrados na amostragem inicial (84%, 84% e 88%). Já a abordagem T3S-Aleatória, inicialmente obteve um valor médio de F1 de 29%, 54% e 64% a partir de uma amostra de 79, 76, 74 pares manualmente rotulados. Quando mais pares são rotulados, pode-se observar uma redução substancial no valor médio de F1 em alguns pontos. Por exemplo, com o maior conjunto de treinamento (197, 216 e 226 pares rotulados) a eficácia do T3S-Aleatória foi de 34%, 40% e 52%. Tal redução na eficácia da abordagem T3S-Aleatória é explicada pelo acréscimo no número de pares identificados como críticos. Como o conjunto de treinamento é pouco informativo (baixo número de pares duplicados), o classificador SVM produz uma predição com uma baixa precisão.

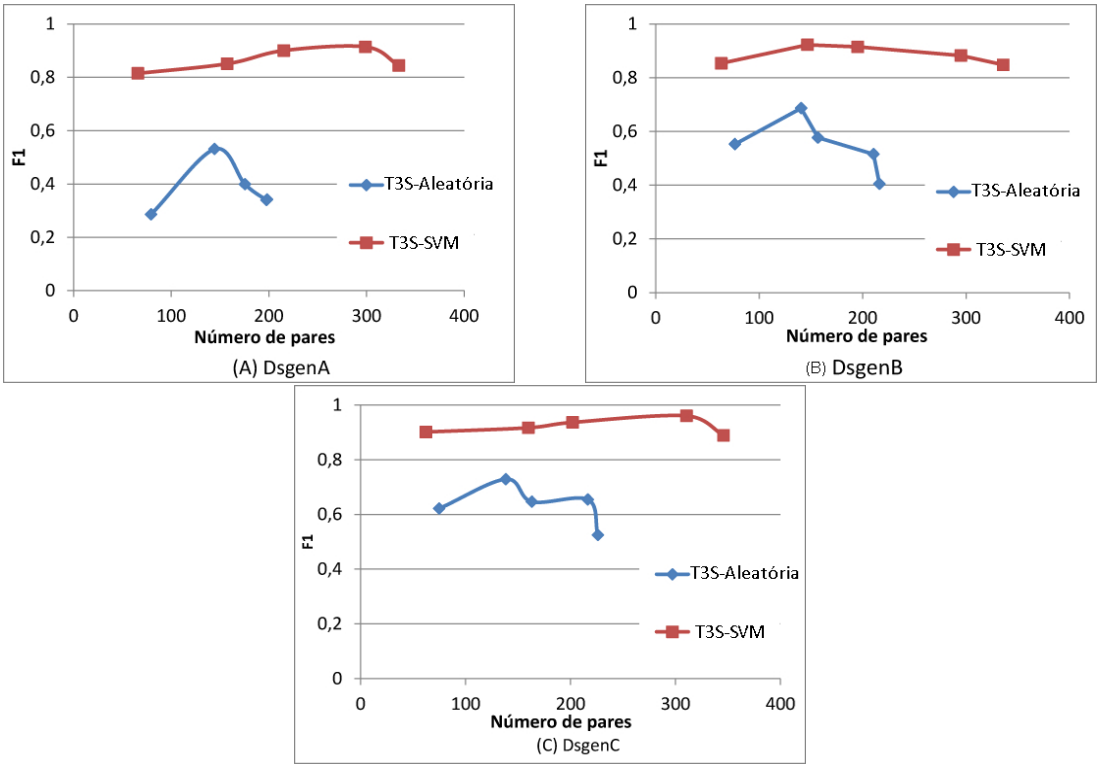


Figura 5.5: Comparação do T3S-SVM com a abordagem T3S-Aleatória (sem a utilização das *faixas* para a seleção de amostras) nas bases sintéticas DsgenA, DsgenB e DsgenC.

Salienta-se que a estratégia T3S-aleatória produz uma amostra rotulada menor que a amostra selecionada pela estratégia T3S-SVM (em alguns casos cerca de 50% menos pares manualmente rotulados). Esse fato é explicado devido à (quase) ausência de pares duplicados nos conjuntos selecionados pela estratégia T3S-Aleatória, reduzindo substancialmente a informatividade e os padrões presentes no conjunto de treinamento.

Por fim, pode-se concluir que a abordagem T3S-Aleatória não foi capaz de produzir amostras balanceadas devido à presença insuficiente de pares duplicados. A experimentação comprovou que a informatividade das amostras produzidas pela abordagem T3S-Aleatória foi insuficiente para configurar o classificador SVM. Já a abordagem T3S-SVM, munida com o conceito de *faixas*, selecionou conjuntos balanceados de pares para a configuração do classificador.

5.5.2 Comparação do T3S com a metodologia FS-Dedup

Os experimentos, apresentados a seguir, têm como objetivo demonstrar que a partir do processo de seleção de pares em duas etapas, promovido pela abordagem T3S, é possível reduzir o esforço do usuário (número de pares manualmente rotulados) sem a depreciação da qualidade da deduplicação. Naturalmente, como *baseline* na experimentação desta seção é adotada a metodologia FS-Dedup, tendo em vista os objetivos semelhantes em relação à redução do esforço do usuário.

Os experimentos consistem em avaliar a demanda de pares manualmente rotulados e medir a qualidade da deduplicação (F1) em cada conjunto de dados. Para cada base de dados, os seguintes passos foram executados:

1. pares candidatos foram gerados utilizando os conceitos propostos pela Etapa de

Ordenamento (apresentados na Seção 3.1.1). Os pares candidatos gerados são organizados em um *ranking* a partir da similaridade de cada par;

2. o *ranking* é fragmentado em 10 *faixas*, de acordo com a Definição 4;

3. a primeira etapa da abordagem T3S é responsável pela coleta de amostras aleatórias contendo 10, 50, 100, 500 ou 1.000 pares, utilizando o conceito de *faixas*. Incrementalmente, cada amostra de faixa é processada pelo segundo passo do T3S para a construção de um conjunto rotulado. Novamente, a rotulação do usuário é simulada utilizando pares previamente classificados;

4. o conjunto de pares selecionado pelo T3S é utilizado para identificar as fronteiras da região crítica. O mesmo conjunto de pares, rotulado previamente, é utilizado para a configuração dos métodos de classificação NGram e SVM (apresentados na Seção 3.1.3) para a predição dos pares duplicados.

As Tabelas 5.11, 5.12, 5.13 e 5.14 detalham o esforço manual e a eficácia da deduplicação, comparando as abordagens T3S-(NGram e SVM) com a metodologia FS-Dedup-(NGram e SVM) nas bases de dados sintéticas e reais.

As abordagens T3S-(SVM e NGram) foram equipadas com uma ou duas funções de similaridade, chamadas T3S-(SVM ou NGram)(1SF) e T3S-(SVM ou NGram)(2SF), com objetivo de incrementar a informatividade do conjunto de treinamento. Em tais tabelas, a coluna "Tam Faixa" especifica o tamanho das amostras de cada faixa (10, 50, 100, 500 ou 1.000 pares). Em seguida, a coluna "F1(σ)" representa os valores médios de eficácia (F1) seguido pelos desvios padrão (σ). O número médio de pares rotulados é descrito na coluna "#Pares". Finalmente, a coluna "%sel" representa a redução no tamanho do conjunto de treinamento comparado com o esforço do usuário da abordagem T3S em relação à metodologia FS-Dedup. O T3S-SVM e o T3S-NGram utilizaram o mesmo conjunto de treinamento para possibilitar a comparação direta da eficácia dos método. Para facilitar a visualização, os valores das colunas "#Pares" e "%Sel" foram replicados nas tabelas referentes ao T3S-NGram (Tabelas 5.11 e 5.13) e ao T3S-SVM (Tabelas 5.12 e 5.14).

Tabela 5.11: Comparação do esforço manual e da eficácia do T3S-NGram com o FS-Dedup-NGram nas bases de dados sintéticas.

Base de dados	Tam Faixa	FS-Dedup-NGram F1(σ)	FS-Dedup-NGram #Pares	T3S-NGram(1FS) F1(σ)	T3S-NGram(1FS) #Pares	T3S-NGram(1FS) % sel	T3S-NGram(2FS) F1(σ)	T3S-NGram(2FS) #Pares	T3S-NGram(2FS) % sel
DsgenA	10	0,84±(0,09)	40	0,83±(0,08)	66	167	0,83±(0,14)	65	166
	50	0,90±(0,08)	196	**0,88±(0,08)**	**157**	80	0,90±(0,06)	171	87
	100	0,94±(0,03)	432	**0,93±(0,02)**	**215**	50	0,91±(0,03)	246	57
	500	0,97±(0,02)	2391	**0,96±(0,03)**	**299**	13	0,96±(0,02)	380	16
	1000	0,97±(0,02)	5127	**0,95±(0,05)**	**333**	6	0,97±(0,02)	469	9
DsgenB	10	0,86±(0,09)	35	0,88±(0,11)	63,5	181	0,90±(0,05)	66	189
	50	0,95±(0,03)	222	**0,94±(0,03)**	**147**	66	0,94±(0,03)	160	72
	100	0,95±(0,03)	462	**0,95±(0,03)**	**195**	42	0,95±(0,03)	213	46
	500	0,98±(0,00)	2724	**0,98±(0,00)**	**295**	11	0,98±(0,01)	349	13
	1000	0,98±(0,00)	5002	**0,97±(0,02)**	**336**	7	0,97±(0,01)	419	8
DsgenC	10	0,92±(0,02)	35	0,92±(0,06)	62	178	0,92±(0,01)	63	180
	50	0,96±(0,03)	237	**0,93±(0,02)**	**160**	67	0,96±(0,02)	155	65
	100	0,97±(0,02)	502	**0,96±(0,04)**	**202**	40	0,97±(0,03)	215	43
	500	0,98±(0,00)	2752	**0,94±(0,06)**	**311**	11	0,96±(0,05)	372	14
	1000	0,98±(0,01)	5902	**0,96±(0,05)**	**346**	6	0,98±(0,02)	497	8

A Tabela 5.11 apresenta uma comparação da métrica F1(σ) do T3S-NGram (equipado com uma (1SF) ou duas função de similaridade(2SF)) com o FS-Dedup-NGram nas bases de dados sintéticas. Pode-se observar que o conjunto de treinamento criado pelo T3S-NGram(1SF) demonstrou ser suficientemente informativo para configurar o método T3S-NGram se comparado ao FS-Dedup-NGram. Com um *tamanho de faixa* de 500 e 1.000 pares, a redução no conjunto de treinamento é mais expressiva que os outros tamanhos de faixas (10, 50 e 100 pares) devido à alta redundância de pares em tais *faixas*. Em outras palavras, quando amostras de *faixas* contendo 1.000 pares são produzidas, o T3S-NGram é capaz de reduzir em mais de 16 vezes a demanda de pares manualmente rotulados sem perdas estatisticamente significativas na eficácia (T3S-NGram demanda de 333, 331 e 346 pares rotulados comparados com os 5,127 e 5,002 e 5,902 pares rotulados pelo FS-Dedup-NGram, na base de dados DsgenA, DsgenB e DsgenC, respectivamente). Naturalmente, com um *tamanho de faixa* menor (100 pares), o T3S-NGram apresentou uma redução menor, mas ainda substancial (cerca de duas vezes menos pares rotulados que o FS-Dedup-NGram), sem uma diferença estatisticamente significativa da eficácia. Claramente, um acentuado número de pares selecionados pelo FS-Dedup-NGram é composto por informações redundantes que podem ser descartadas sem impactar na eficácia da deduplicação nas bases de dados sintéticas.

Ao utilizar duas funções de similaridade, o T3S-NGram não apresentou melhorias significativas na eficácia nas bases de dados sintéticas. De fato, o conjunto de treinamento criado pelo T3S-NGram(1SF) demonstrou ser suficientemente informativo para identificar corretamente o valor do limiar a ser usado pelo método T3S-NGram.

Tabela 5.12: Comparação do esforço manual e da eficácia do T3S-SVM com o FS-Dedup-SVM nas bases de dados sintéticas.

Base de dados	Tam Faixa	FS-Dedup-SVM F1(σ)	FS-Dedup-SVM #Pares	T3S-SVM(1FS) F1(σ)	T3S-SVM(1FS) #Pares	T3S-SVM(1FS) % sel	T3S-SVM(2FS) F1(σ)	T3S-SVM(2FS) #Pares	T3S-SVM(2FS) % sel
DsgenA	10	0,87±(0,07)	40	0,81±(0,07)	66	167	0,81±(0,13)	65	166
	50	0,89±(0,07)	196	**0,85±(0,07)**	**157**	80	0,87±(0,05)	171	87
	100	0,88±(0,10)	432	**0,90±(0,01)**	**215**	50	0,89±(0,03)	246	57
	500	0,89±(0,04)	2391	**0,91±(0,06)**	**299**	13	0,90±(0,07)	380	16
	1000	0,92±(0,04)	5127	0,84±(0,24)	333	6	**0,94±(0,02)**	**469**	9
DsgenB	10	0,87±(0,07)	35	0,85±(0,11)	64	181	0,85±(0,08)	66	189
	50	0,92±(0,03)	222	**0,92±(0,03)**	**147**	66	0,90±(0,06)	160	72
	100	0,92±(0,03)	462	**0,91±(0,05)**	**195**	42	0,93±(0,03)	213	46
	500	0,92±(0,03)	2724	**0,88±(0,08)**	**295**	11	0,95±(0,01)	349	13
	1000	0,94±(0,02)	5002	0,85±(0,26)	336	7	**0,93±(0,05)**	**419**	8
DsgenC	10	0,92±(0,02)	35	0,90±(0,05)	62	178	0,84±(0,19)	63	180
	50	0,92±(0,07)	237	**0,92±(0,03)**	**160**	67	0,92±(0,11)	155	65
	100	0,94±(0,04)	502	**0,94±(0,04)**	**202**	40	0,93±(0,07)	215	43
	500	0,96±(0,01)	2752	**0,96±(0,01)**	**311**	11	0,95±(0,01)	372	14
	1000	0,94±(0,02)	5902	0,89±(0,18)	346	6	**0,96±(0,00)**	**497**	8

Na tabela 5.12 são apresentados detalhes do número de pares manualmente rotulados e do valor de F1 comparando a abordagem T3S-SVM (equipada por uma (1SF) ou duas (2SF) funções de similaridade) em relação ao FS-Dedup-SVM, nos conjuntos de dados sintéticos. Pode-se observar que o T3S-SVM(1SF) alcançou uma equivalência estatística da métrica F1 em todos os *tamanhos de faixas* em relação ao FS-Dedup-SVM, similar ao comportamento do T3S-NGram. No entanto, com um *tamanho de faixa* de 1.000 pares, o T3S-SVM(1FS) resultou em um valor instável de F1 devido à região crítica ser composta por um volume maior de pares candidatos se comparado aos outros *tamanhos*

de faixas. Um volume maior de pares candidatos demanda de um conjunto de treinamento mais informativo para a criação de um modelo de classificação mais efetivo. Mais especificamente, utilizando a base de dados DsgenA e uma *faixa* de 1.000 pares, o T3S-SVM(1SF) resultou em uma redução média no valor de F1 de cerca de 10%, em relação ao FS-Dedup-SVM. Tais resultados podem ser explicados devido aos dados sintéticos serem compostos por um acentuado número de padrões de duplicatas (inversão de valores de atributos, remoção/alteração de caracteres, remoção de atributos entre outros) que são facilmente identificados pelo T3S-NGram (ou seja, o ordenamento global promovido pelo Sig-Dedup é capaz de identificar facilmente a ausência ou inversão de atributos).

Para contornar o problema da baixa informatividade do conjunto de treinamento, a abordagem T3S-SVM foi equipada com duas funções de similaridade (T3S-SVM(2SF)) nas bases de dados sintéticas, como reportado na Tabela 5.12. Como pode ser observado, o T3S-SVM(2SF) com um *tamanho de faixa* de 1.000 pares resultou em valores estáveis de F1 com a penalidade de um aumento no número de pares manualmente rotulados de cerca de 40%, 25% e 42% comparado ao T3S-SVM(1SF) nas bases de dados DsgenA, DsgenB e DsgenC, respectivamente. Mesmo assim, em relação ao FS-Dedup, o T3S-SVM(2SF) foi capaz de reduzir o número de pares rotulados (em mais de 90% na *faixa* de 1.000 pares) sem variações substanciais no valor médio do F1 nos conjuntos de dados sintéticos.

O T3S-NGram e o T3S-SVM apresentaram um acréscimo no número de pares manualmente rotulados com um *tamanho de faixa* de 10 pares em cerca de 60% em relação ao FS-Dedup. Tal comportamento é explicado pelo baixo número de pares presentes em tal *tamanho de faixa*, o que reduz substancialmente a presença de pares com informações redundantes. Com tal tamanho de faixa praticamente todos os pares selecionados pelo primeiro passo da abordagem T3S são informativos para complementar o conjunto de treinamento, enquanto o modelo adotado pela metodologia FS-Dedup, na qual são rotuladas somente amostras de *faixas* consideradas relevantes para a identificação da região crítica, permite selecionar *faixas* que produzem algum ganho de informatividade em relação ao conjunto de treinamento. Dessa forma, a metodologia FS-Dedup é capaz de produzir um conjunto inicial informativo e com baixo número de pares rotulados, o que não ocorre quando utilizados *tamanhos de faixas* mais elevados (acima de 100 pares).

Como detalhado nas Tabelas 5.13 e 5.14, nos conjuntos de dados reais (IMDBxNetFlix e DBLPxCiteSeer), ambos os métodos T3S-NGram e T3S-SVM alcançam um valor de F1 competitivo com uma redução substancial no número de pares manualmente rotulados em relação à metodologia FS-Dedup. Por exemplo, na base de dados IMDBxNetFlix, o conjunto de treinamento criado pelas abordagens T3S-(SVM-NGram) representaram somente 10% do total de pares selecionados pela metodologia FS-Dedup, com um *tamanho de faixa* de 100 pares (73 pares compõem o conjunto de treinamento da abordagem T3S comparado aos 696 da metodologia FS-Dedup) sem uma diferença estatisticamente significativa da métrica F1. A utilização de duas funções de similaridade, com objetivo de aumentar a informatividade do conjunto de treinamento, não produziu melhorias na qualidade do processo.

Excepcionalmente, o método T3S-SVM obteve uma melhoria significativa da eficácia na base de dados IMDBxNetFlix com uma redução substancial no número de pares rotulados em relação ao FS-Dedup-SVM. Mais especificamente, com um *tamanho de faixa* de 100 pares foi possível melhorar significativamente em 5% a eficácia com uma redução de mais de 9 vezes no número de pares rotulados em relação ao FS-Dedup-SVM.

Na base de dados DBLPxCiteSeer, os métodos T3S-(NGram e SVM) alcançaram um

valor de F1 sem variações estatisticamente significativas em relação ao FS-Dedup, a partir de somente um *tamanho de faixa* de 10 pares. Na base de dados DBLPxCiteSeer, foi necessário rotular somente 27 pares (redução de 40% em relação ao FS-Dedup) para atingir uma eficácia de mais de 92%. A variação marginal do valor de F1 na base de dados DBLPxCiteSeer, em diferentes *tamanhos de faixas*, é explicada pelo fato da base de dados ser composta por dados automaticamente recuperados da Web, produzindo duplicatas com informação incompletas que facilmente são classificadas pelos métodos propostos.

Note-se que nas bases de dados reais, a abordagem T3S não apresentou um aumento no número de pares rotulados em relação à metodologia FS-Dedup nos *tamanhos de faixas* iniciais (10 pares), diferentemente dos resultados reportados nas bases de dados sintéticas. Isso é explicado pelo baixo número de atributos presentes nas bases reais (três atributos) que resultam em poucos padrões de duplicatas se comparado às bases sintéticas (compostas por 10 atributos). Como o algoritmo de aprendizagem ativa SSAR utiliza como critério para a seleção dos pares o número de atributos em comum, é esperado que em bases de dados reais, com menos atributos, ocorra a convergência com menos pares rotulados do que nas bases de dados sintéticas.

Tabela 5.13: Comparação do esforço manual e da eficácia do T3S-NGram com o FS-Dedup-NGram nas bases de dados reais.

Base de	Tam	FS-Dedup-NGram		T3S-NGram(1FS)			T3S-NGram(2FS)		
dados	Faixa	F1	#Pares	F1	#Pares	% sel	F1	#Pares	% sel
IMDBx	10	0,88±(0,04)	47	**0,81±(0,06)**	**28**	60	0,89±(0,06)	34	73
NetFlix	50	0,89±(0,04)	330	**0,89±(0,06)**	**54**	16	0,91±(0,03)	75	23
	100	0,92±(0,00)	696	**0,93±(0,00)**	**73**	10	0,93±(0,00)	103	15
DBLPx	10	0,91±(0,01)	45	**0,91±(0,02)**	**27**	60	0,91±(0,02)	31	70
CiteSeer	50	0,92±(0,01)	285	0,92±(0,00)	53	18	0,92±(0,00)	79	28
	100	0,92±(0,00)	581	0,92±(0,00)	75	13	0,92±(0,00)	107	18

Tabela 5.14: Comparação do esforço manual e da eficácia do T3S-SVM com o FS-Dedup-SVM nas bases de dados reais.

Base de	Tam	FS-Dedup-SVM		T3S-SVM(1FS)			T3S-SVM(2FS)		
dados	Faixa	F1	#Pares	F1	#Pares	% sel	F1	#Pares	% sel
IMDBx	10	0,86±(0,01)	47	0,80±(0,03)	28	60	**0,90±(0,05)**	**34**	73
NetFlix	50	0,88±(0,01)	330	**0,88±(0,06)**	**54**	16	0,91±(0,04)	75	23
	100	0,88±(0,00)	696	**0,93±(0,01)**	**73**	10	0,94±(0,01)	103	15
DBLPx	10	0,94±(0,01)	45	**0,92±(0,02)**	**27**	60	0,93±(0,00)	31	70
CiteSeer	50	0,94±(0,01)	285	0,92±(0,01)	53	18	0,93±(0,01)	79	28
	100	0,94±(0,00)	581	0,94±(0,00)	75	13	0,93±(0,01)	107	18

Em suma, os experimentos descritos nessa seção demostraram que o T3S-NGram foi capaz de reduzir substancialmente a demanda por pares manualmente rotulados sem perdas na qualidade dos pares em relação à metodologia FS-Dedup. Já o T3S-SVM, nas bases de dados sintéticas, depende de duas funções de similaridade para obter um valor competitivo de F1, mesmo assim, resultou em uma redução substancial no número de pares rotulados em relação ao FS-Dedup. Nas bases de dados reais (IMDBxNetFlix e DBLPxCiteSeer), ambos os métodos T3S-SVM e T3S-NGram, com uma função de similaridade (1SF) e um *tamanho de faixa* de 100 pares, alcançaram um valor competitivo de F1 utilizando menos de 13% do total de pares rotulados demandado pela metodologia FS-Dedup. De fato, para configurar o T3S-NGram com uma eficácia de 92% na base de

dados DBLPxCiteSeer (com cerca 2,8 milhões de registros) foram necessários somente 27 pares manualmente rotulados. Um ponto interessante identificado nos experimentos dessa seção foi a melhora da qualidade da deduplicação do método T3S-SVM mesmo que com uma redução substancial no número de pares candidatos, demonstrando a capacidade da abordagem T3S em selecionar um conjunto de treinamento altamente informativo.

5.5.3 Comparação do esforço do usuário

A seguir, são reportados experimentos comparando o esforço do usuário e a eficácia do T3S-(NGram e SVM) em relação ao método ALD, proposto por BELLARE et al. (2012). ALD representa um recente método de aprendizagem ativa com objetivo de reduzir o esforço do usuário na deduplicação. Como já mencionado anteriormente (Seção 5.1.3), somente serão reportados experimentos utilizando bases de dados sintéticas devido aos dados reais terem sido rotulados utilizando o conceito de *faixas*, proposto pela metodologia FS-Dedup. Dessa forma, utilizar os dados rotulados a partir da metodologia FS-Dedup resultaria em uma injusta vantagem à abordagem ALD.

Nos gráficos da Figura 5.6 são ilustrados os resultados da comparação do ALD em relação ao T3S-(NGram e SVM) utilizando uma ou duas funções de similaridade, a partir da eficácia e do número de pares rotulados. Pode-se observar que em todos os conjuntos de dados, o T3S-NGram(1SF) converge para um alto valor de eficácia com um reduzido número de pares rotulados (cerca de 300 pares). A utilização de duas funções de similaridade (T3S-NGram(2SF)) não reflete em melhorias na qualidade da abordagem proposta. Já o método T3S-SVM(1SF), com um *tamanho de faixa* de 1.000 pares, apresentou valores médios de eficácia instáveis, criando a necessidade de equipar o método T3S-SVM com duas funções de similaridade para melhorar sua eficácia.

O método ALD, que sempre depende inicialmente de uma amostra de até 100 pares rotulados para a configuração do "oráculo", inicia com valores de F1 instáveis nas bases de dados DsgenA e DsgenB, como ilustrado nos gráficos da Figura 5.6. Nas bases de dados DsgenA e DsgenB, a abordagem ALD demanda de 1,3 e 2,0 vezes mais pares manualmente rotulados com perdas significativas na eficácia (uma redução no valor de F1 de 10% e 2,1%, respectivamente) em relação ao T3S-NGram. Devido ao reduzido número de pares duplicados, tais bases de dados se tornam mais desafiadoras para a classificação dos pares. ALD apenas alcança uma eficácia estatisticamente competitiva com cerca de 6,300 e 8,200 pares manualmente rotulados, ou seja, cerca de 20 vezes mais pares em relação ao T3S-NGram, nas bases de dados DsgenA e DsgenB, respectivamente. Já a base de dados DsgenC, que representa o conjunto de dados menos desafiador para a classificação devido à abundância de pares duplicados, todos os métodos (T3S-SVM, T3S-NGram e ALD) convergem para uma eficácia de 97% sem uma diferença estatisticamente significante, mas o ALD apresentou demanda mais elevada de pares rotulados. T3S-SVM(2FS), que demanda mais pares em relação ao T3S-NGram(1FS), alcança um valor de F1 competitivo com a redução de 7%, 58% e 56% no número de pares rotulados comparado ao ALD.

Por fim, pode-se concluir que o T3S foi capaz de reduzir substancialmente o conjunto de treinamento se comparado ao ALD nas bases de dados sintéticas, sem variações significativas no F1. Em suma, esse experimento demonstrou que o T3S reduz substancialmente o esforço manual e produz um competitivo valor de eficácia em termos de F1 se comparado ao método de aprendizagem ativa ALD.

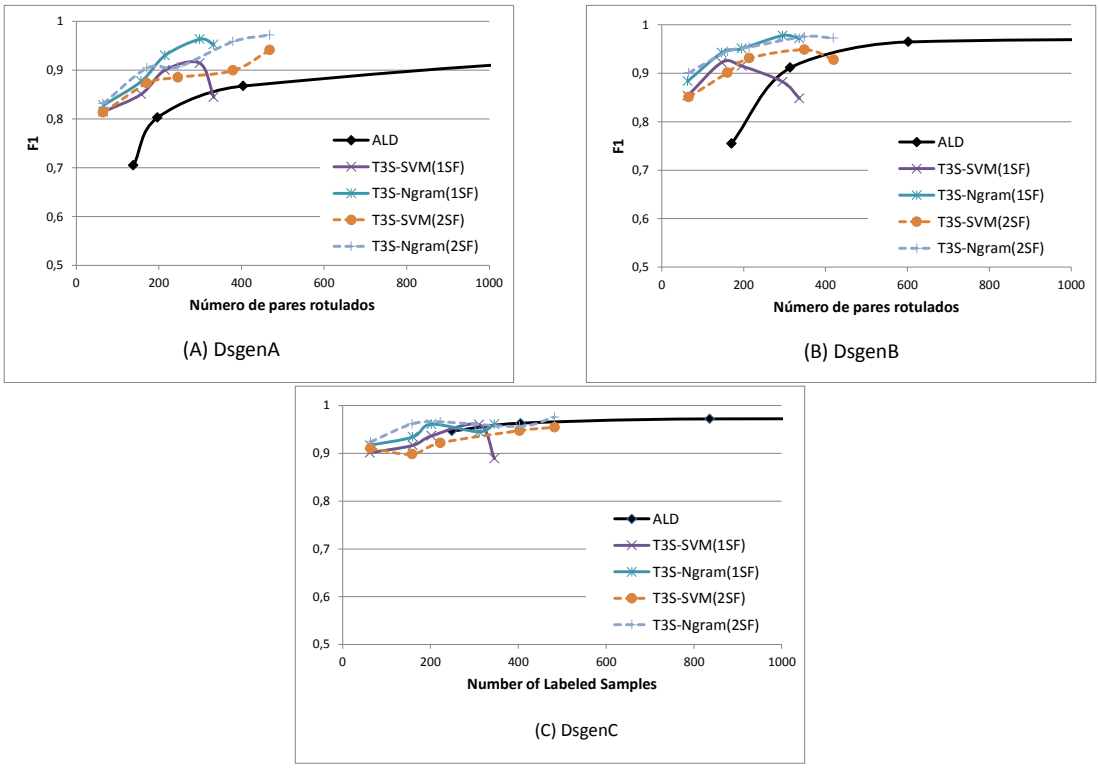


Figura 5.6: Comparação do T3S-SVM e T3S-NGram com uma ou duas funções de similaridades com a abordagem ALD nas bases sintéticas DsgenA, DsgenB e DsgenC.

5.6 Considerações finais

A partir da descrição experimental apresentada nesse capítulo, pode-se afirmar que é possível remover a necessidade de intervenção do especialista na deduplicação de grandes conjuntos de dados. Além disso, os experimentos comprovaram que é plausível calibrar as etapas internas da deduplicação requisitando que o usuário somente rotule um conjunto reduzido de pares. Esse cenário é promissor, visto que o usuário foi removido completamente da tarefa pouco intuitiva de definir manualmente valores de limiares.

Nos experimentos, a metodologia proposta foi capaz de obter uma eficácia similar a métodos configurados otimamente pelo usuário e, até mesmo, obter uma melhora na qualidade da deduplicação. Para obter tais resultados, foram utilizadas estratégias para geração de conjuntos controlados de pares candidatos, identificação das fronteiras da região crítica, criação de um conjunto de treinamento altamente informativo e identificação do limiar necessário para a classificação dos pares. Tais estratégias visam minimizar a necessidade de intervenção do usuário a partir de uma série de definições que podem, em algum momento, falhar ou ser pouco efetivas. No entanto, apesar dos diversos fatores que dificultam a deduplicação de grandes conjuntos de dados (por exemplo, a geração de pares candidatos, qualidade do processo, intervenção do usuário), nessa experimentação inicial a metodologia demostrou resultados promissores em relação à redução do esforço do usuário em todos os conjuntos de dados analisados.

Identificar quais pares contêm informações relevantes para a configuração da deduplicação demonstrou-se ser uma tarefa complexa agravada pelo custo computacional de se processar grandes montantes de dados. Assim, foi proposta uma abordagem, em dois passos, para contornar o problema da seleção de pares informativos em grandes conjun-

tos de dados, sem depreciar a qualidade do processo de deduplicação. A partir desses experimentos iniciais é possível concluir que a abordagem para a seleção do conjunto de treinamento foi capaz de reduzir substancialmente o esforço manual. Mais ainda, foi capaz de melhorar a qualidade da deduplicação com uma redução substancial no número de pares rotulados.

6 CONCLUSÕES

Neste capítulo, é apresentada uma visão geral das contribuições da tese. Dentre elas, são listadas as publicações resultantes dos trabalhos realizados no decorrer do doutorado. Por fim, são apresentadas algumas direções de trabalhos interessantes, que podem ser desenvolvidos a partir da presente tese.

6.1 Contribuições

De um modo geral, o objetivo final para a tarefa da deduplicação de dados é promover métodos eficientes, eficazes e capazes de operar sem a supervisão de um usuário. No entanto, apesar do atual estado da arte da deduplicação de grandes volumes de dados oferecer soluções escaláveis e eficientes, ainda é exigido do usuário a tarefa de intervir diretamente, a partir da definição de limiares, nas principais etapas do processo de identificação de duplicatas.

Nessa perspectiva, a presente tese teve como foco principal o desenvolvimento de propostas para a redução da intervenção manual no contexto da deduplicação de grandes conjuntos de dados. Mais especificamente, a tese tratou a hipótese de que é possível remover a necessidade de uma intervenção especialista para a tarefa de definir valores de limiares para a configuração da deduplicação. Foi mostrado que é possível reduzir a dispendiosa tarefa de rotular manualmente pares para a calibração das principais etapas da deduplicação sem perdas na eficácia.

Inicialmente, foi apresentado um estudo detalhado sobre os problemas envolvendo a deduplicação, especialmente no contexto de grandes volumes de dados. A partir desse estudo, foi detectada a convergência na fundamentação de diversos trabalhos para o tratamento do problema da eficiência da deduplicação. Além disso, foi identificada a demanda da configuração especialista nas principais etapas internas da deduplicação. Desse modo, foi proposta uma nova metodologia, chamada de FS-Dedup, capaz de promover um mapeamento entre as informações fornecidas pelo usuário (pares manualmente rotulados) e as informações requisitadas pelo deduplicador (valores de limiares), a fim de possibilitar a calibração da deduplicação. A metodologia proposta é responsável pela tradução das informações requisitadas pelo deduplicador, evitando que o usuário seja responsável por tarefas pouco intuitivas, que exigem um conhecimento prévio do contexto em questão.

Foi proposta ainda uma estratégia visando controlar a geração de pares candidatos com foco na maximização do número de pares duplicados. O objetivo foi evitar um inviável custo quadrático na geração dos pares sem depreciar o número de pares duplicados (maximizar a revocação). Em seguida, estratégias foram empregadas para selecionar pequenas amostras, para a rotulação manual, visando identificar as regiões nas quais os pares mais ambíguos estão concentrados (chamada de região crítica da base de dados). A

identificação das fronteiras da região crítica permite definir a posição em que se encontram os pares mais desafiadores de serem classificados. Constata-se que a identificação da região crítica permite remover um volume substancial de pares candidatos, reduzindo substancialmente os custos de processamento da Etapa de Classificação.

Os pares considerados críticos foram classificados utilizando dois métodos de classificação. O primeiro visava identificar variações ou erros nos valores de atributos e a inversão de valores de campos, a partir do cálculo da similaridade entre os pares. No entanto, um problema surgiu em como definir o valor do limiar capaz de capturar os pares verdadeiros e descartar os pares não duplicados. Para isso, foi adotada uma estratégia de janela deslizante, aplicada sobre o conjunto de pares previamente rotulados, para identificar o limiar capaz de separar os grupos dos pares duplicados dos pares não duplicados. O segundo método de classificação utilizou o algoritmo de aprendizagem de máquina SVM para a construção de um modelo de classificação, a partir das informações presentes no conjunto de treinamento. O modelo de classificação é utilizado para a predição dos pares não rotulados.

Com o objetivo de validar a metodologia FS-Dedup, foram produzidos cenários reais com volumes substanciais de dados (até três milhões de registros) a partir da integração de bases de dados reais. Amostras das bases de dados foram rotuladas manualmente (um total de cerca de 8.000 pares) para permitir a avaliação da metodologia em cenários reais. Destaca-se que, até então, as bases de dados reais com grande volume de dados não eram abertamente disponíveis. Dessa forma, a disponibilização de tais bases de dados reais possibilita que novas técnicas sejam avaliadas experimentalmente, evitando o custoso trabalho de coleta e rotulação de pares. Mais ainda, a padronização das bases de dados possibilita comparar e quantificar diferentes avanços no estado da arte da deduplicação de dados.

Na experimentação promovida com objetivo de avaliar a metodologia FS-Dedup, foram identificados resultados promissores em relação à capacidade da metodologia em identificar a configuração ideal em cada conjunto de dados. A metodologia foi capaz de obter uma eficácia competitiva, se comparada ao algoritmo Sig-Dedup (calibrado manualmente com valores de limiares). Foi constatado que a estratégia de geração de pares candidatos permitiu reduzir o número de pares rotulados sem perdas substanciais na qualidade dos pares. Ainda assim, a capacidade de identificar a configuração ideal está relacionada a um conjunto de treinamento representativo.

Para mitigar o problema do custo da rotulação manual, na metodologia FS-Dedup, foi proposta uma nova abordagem para a seleção de um reduzido e informativo conjunto de pares candidatos a compor o conjunto de treinamento. A abordagem, chamada T3S, visa identificar os pares informativos em dois passos. No primeiro passo, são produzidas amostras reduzidas e balanceadas de pares. Em seguida, no segundo passo, cada amostra é processada incrementalmente por um método de aprendizagem ativa para a seleção dos pares altamente dissimilares para serem submetidos ao processo de rotulação manual. O primeiro passo é essencial para reduzir o custo computacional do método de aprendizagem ativa. A abordagem T3S é integrada à metodologia FS-Dedup, com objetivo de reduzir a intervenção nas principais etapas da deduplicação.

Na experimentação, a abordagem T3S foi capaz de reduzir substancialmente a demanda de pares manualmente rotulados (em até 16 vezes em relação à metodologia FS-Dedup), sem reduzir a eficácia do processo. Se comparada a um recente método de aprendizagem ativa para deduplicação, a abordagem T3S demonstrou ser substancialmente superior em relação à redução do custo manual. A partir da experimentação inicial, é possí-

vel concluir que a integração da abordagem T3S com a metodologia FS-Dedup permitiu alcançar notáveis avanços na redução do esforço do usuário em relação ao estado na arte na deduplicação de grandes bases de dados.

A seguir, são apresentadas as produções científicas e os trabalhos desenvolvidos relacionados à tese:

- Guilherme Dal Bianco, Renata Galante, Marcos André Gonçalves, and Carlos A. Heuser. 2014. Two stage sampling selection. In IEEE Transactions on Knowledge and Data Engineering (TKDE), (Submetido para avaliação em dezembro 2013, Qualis-CC A1).

- Guilherme Dal Bianco, Renata Galante, Marcos André Gonçalves, and Carlos A. Heuser. 2013. Tuning large scale deduplication with reduced effort. In Proceedings of the 25th International Conference on Scientific and Statistical Database Management (SSDBM). ACM, New York, NY, USA, 12 pages. (Qualis-CC A2)

- Guilherme Dal Bianco, Renata Galante, and Carlos A. Heuser. 2012. FS-Dedup - A Framework for Signature-based Deduplication in large datasets. Simpósio Brasileiro de Bancos de Dados - SBBD 2012, Workshop de Teses e Dissertações, São Paulo, SP.

- Guilherme Dal Bianco, Renata Galante, and Carlos A. Heuser. 2011. A fast approach for parallel deduplication on multicore processors. In Proceedings of the 2011 ACM Symposium on Applied Computing (SAC '11). ACM, New York, NY, USA, 1027-1032. (Qualis-CC A1)

Além dos artigos citados anteriormente, outros trabalhos foram desenvolvidos explorando assuntos indiretamente relacionados a esta tese:

- Sérgio Canuto, Guilherme Dal Bianco, Marcos Gonçalves, Thierson Couto and Jussara Almeida. 2013. UDRB: Uma Nova Heurística Eficaz para Deduplicação de Referências Bibliográficas. Simpósio brasileiro de banco de dados (SBBD), Recife, PE.(Resumo)

- Guilherme Dal Bianco, Renata Galante, Mirella M. Moro. 2009. P-Canopy, uma Proposta para Blocagem Paralela. Escola regional de alto desempenho (Erad), Caxias do Sul, RS. (Resumo)

Além disso, está sendo implementado por um aluno de iniciação científica uma extensão da presente tese como objetivo de endereçar o problema da deduplicação online, como será descrito na Seção 6.2.

Por fim, a experiência adquirida durante a tese permitiu contribuir diretamente com o desenvolvimento do trabalho de conclusão do curso de graduação descrito a seguir:

- Tiago Gomes Santos. Uma Ferramenta para inclusão de Web Forms à base WFSim. 2013. Trabalho de Conclusão do Curso. (Graduação em Ciências da Computação)-Universidade Federal do Rio Grande do Sul. Orientadora: Prof[a] Renata Galante, Co-orientação: Guilherme Dal Bianco.

6.2 Trabalhos futuros

A experiência de pesquisa, adquirida durante o desenvolvimento da tese, permitiu identificar alguns problemas que não foram abordados nesse trabalho. É apresentada, então, a seguir, uma visão geral desses problemas, junto ao esboço das possíveis soluções.

6.2.1 Avaliar a eficiência da metodologia FS-Dedup

A metodologia FS-Dedup tem como objetivo configurar a deduplicação em grandes montantes de dados, utilizando os algoritmos Sig-Dedup. Mais especificamente, a implementação do deduplicador Sig-Dedup foi construída utilizando a plataforma MapReduce (DEAN; GHEMAWAT, 2008), para permitir o processamento de grandes montantes de dados (VERNICA; CAREY; LI, 2010). Até onde sabemos, nenhuma abordagem, presente na bibliografia, promoveu a redução do esforço do usuário no contexto da deduplicação em arquiteturas distribuídas (LEE et al., 2012). Dessa forma, cria-se o cenário em que a metodologia FS-Dedup seja experimentalmente avaliada em relação à eficiência e à escalabilidade. Um primeiro desafio para alcançar tal objetivo é obter bases de dados, compostas por dezenas de milhões de registros, justificando a utilização em uma arquitetura distribuída. Em VERNICA; CAREY; LI (2010), é proposta a extensão do algoritmo Sig-Dedup para a plataforma MapReduce, em que bases de dados reais foram acrescidas de pares sinteticamente criados para se obter um volume apropriado de dados. No entanto, tais registros sintéticos podem conter padrões de duplicatas distantes dos cenários reais, tendenciando a avaliação da qualidade dos pares. Como o objetivo de VERNICA; CAREY; LI (2010) é avaliar somente a eficiência de sua abordagem, não foram feitos experimentos em relação à eficácia. Como um dos objetivos da metodologia FS-Dedup é obter uma alta qualidade dos pares, mais estudos devem ser feitos para identificar estratégias factíveis para a geração dos dados, que permitam uma ampla avaliação em cenários reais.

6.2.2 Promover a combinação de funções de similaridade para melhorar a qualidade da classificação da metodologia FS-Dedup

A metodologia FS-Dedup mostrou ser promissora na tarefa da calibração da deduplicação. No entanto, não foram realizados esforços concentrados para aprimorar a qualidade dos métodos de classificação, tal como, a ausência de estudos específicos com objetivo de identificar a função de similaridade ideal para cada atributo, ou seja, a função de similaridade capaz de separar otimamente os pares duplicados dos não duplicados.

Nesse contexto, a Programação Genética (PG) tem se mostrado bastante promissora no intuito de identificar, a partir de um conjunto de treinamento, a melhor combinação de atributos, funções de similaridade e os respectivos pesos (CARVALHO et al., 2008, 2012; KARTHIGHA; ANAND, 2013; ISELE; BIZER, 2013; GODOI et al., 2013). Como resultado da PG, é produzida uma função de classificação com um desenho semelhante a uma árvore, na qual operadores matemáticos básicos são utilizados para determinar a melhor combinação dos atributos.

Incorporando as características da PG ao FS-Dedup, pretende-se explorar diferentes combinações de funções de similaridade, para aprimorar a qualidade dos resultados alcançados. Atualmente o método FS-Dedup-NGram é equipado com somente uma função de similaridade (Jaccard NGram), que pode ser pouco efetiva em alguns cenários. Assim, a combinação de diferentes funções de similaridade pode facilitar a identificação de pares duplicados e, até mesmo, reduzir o número de pares manualmente rotulados. Assim, faz-

se necessário avaliar em maior profundidade o uso da PG para verificar sua viabilidade.

6.2.3 Identificar o melhor método de classificação

Os métodos de deduplicação SVM e NGram (T3S e FS-Dedup) obtiveram resultados complementares na experimentação em relação à eficácia. O método NGram apresentou uma alta eficácia quando o conjunto de dados é composto por pequenas variações em caracteres e pela inversões de campos, o que ocorreu nas bases de dados sintéticas. Já o método SVM foi mais efetivo em cenários compostos por atributos com diferentes relevâncias, como demonstrado na base de dados DBLPxCiteSeer. Tal constatação deixa em aberto como identificar o melhor método de classificação, sem um conhecimento prévio do nível de ruído da base de dados. Mais estudos devem ser realizados para identificar uma possível estratégia para tal problema.

6.2.4 Estender os métodos baseados em assinaturas para o contexto da deduplicação online

Tipicamente, a deduplicação é executada em cenários estáticos, nos quais todos os registros são analisados em busca de duplicatas. Quando a base de dados sofre constantes alterações (por exemplo, *streaming* de dados), processar todos os registros em busca de duplicatas torna-se uma tarefa impraticável. A deduplicação online (ou incremental) visa preservar a qualidade dos dados, evitando que os dados ruidosos sejam indevidamente armazenados na base de dados (CHRISTEN; GAYLER; HAWKING, 2009; HASSANZADEH et al., 2011). Serviços de coleta de dados sofrem diretamente com o problema da deduplicação online. Por exemplo, o Twitter recebe em média mais de 500 milhões de mensagens (tweets) por dia (www.twitter.com), sendo que muitas apresentam informações duplicadas. Os serviços de monitoramento de redes sociais, por exemplo, devem ser capazes de processar tal volume de dados para a extração das informações consideradas relevantes.

O principal problema da deduplicação online é a demanda de processamento de grandes volumes de dados em um intervalo reduzido de tempo (ou em tempo real). Além disso, a base de dados pode sofrer alterações nos padrões de duplicatas, devido às constantes atualizações/alterações nos dados, forçando o deduplicador online a adaptar-se a diferentes níveis de ruídos. Assim, cria-se a necessidade de soluções para a deduplicação online que promovam a eficiência, eficácia e a baixa intervenção do usuário na presença de grandes volumes de dados. Tipicamente, a bibliografia explora soluções para a deduplicação estática (por exemplo, os métodos *baseados em assinaturas*, apresentados na Seção 2.3.2.1), criando uma lacuna de soluções específicas para o problema da deduplicação online.

Nesse cenário, surge a possibilidade de expandir a estrutura dos *filtros de prefixo, sufixo, posição e tamanho* (detalhados previamente na Seção 2.3.2.1) para o contexto da deduplicação online. A Figura 6.1 ilustra um esboço da ideia inicial. O *filtro de prefixo* parte da intuição de que termos menos frequentes são mais relevantes para a identificação de duplicatas e, a partir do reordenamento de cada registro utilizando a frequência global dos termos, é possível agrupar com uma alta eficiência os pares duplicados. Mais ainda, o filtro de prefixo garante que somente termos menos frequentes sejam indexados durante a etapa de blocagem, gerando blocos com um pequeno número de registros. Os demais filtros são encarregados da tarefa de remover pares candidatos que não respeitam determinadas condições, tais como uma variação máxima do tamanho dos registros, alteração no posicionamento (respeitando um valor de limiar) dos termos no prefixo e no sufixo

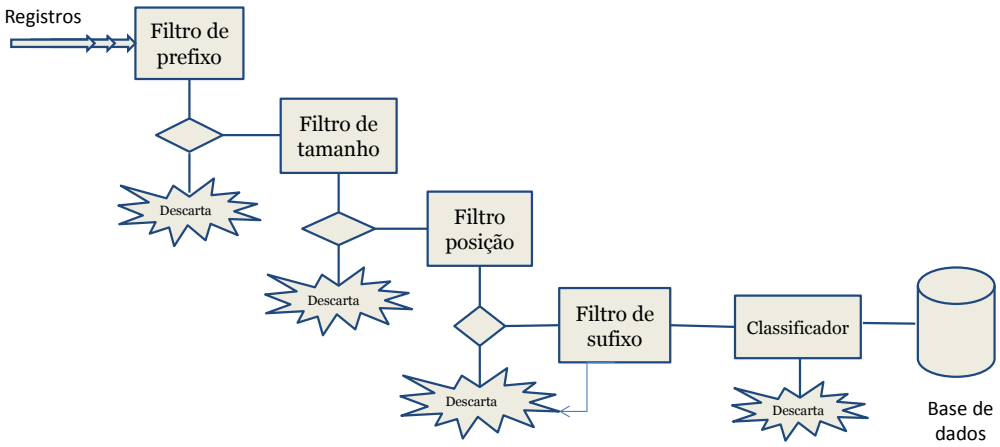


Figura 6.1: Visão geral da abordagem para a deduplicação online que está em fase de desenvolvimento

dos registros. No contexto da deduplicação online, os filtros podem ser encadeados para permitir o descarte de pares com baixas chances de representarem duplicatas, com uma promissora eficiência.

O principal desafio da deduplicação online, utilizando o processo de filtragem, é manter atualizada a estrutura que contabiliza a frequência global dos termos em memória principal. À medida que novos registros são inseridos na base de dados, termos que eram considerados relevantes podem se tornar frequentes e pouco representativos. A ordem dos registros deve ser alterada para acomodar as possíveis variações na frequência dos termos. Dessa forma, o método deve ser capaz de atualizar constantemente os termos presentes no contador de frequência global.

Um primeiro esboço para tratar do problema da deduplicação online está sendo desenvolvido a partir de um *framework* para computação distribuída em tempo real, chamado Storm (ANDERSON, 2013). Semelhante ao modelo MapReduce, o *framework* Storm propõe um modelo abstrato para a gerência de processos, capaz de atribuir tarefas e gerenciar a comunicação entre os processos. Assim, cada filtro está sendo modelado como uma função independente, compartilhando a estrutura de frequência dos termos. Uma vez detectado um gargalo em alguma das funções ou etapas do processos, o *framework* permite disparar novos processos a fim de evitarem atrasos, tendo em vista a necessidade de respostas em tempo real.

6.2.5 Reduzir o esforço manual da deduplicação online

Como descrito anteriormente, a deduplicação online demanda que a deduplicação seja monitorada periodicamente para evitar que alterações nos padrões de duplicatas não impactem na qualidade da deduplicação. À medida que a base de dados é atualizada a configuração do deduplicador pode não ser suficientes para manter a qualidade esperada na identificação dos pares duplicados. Nesse cenário, surge a lacuna de como manter atualizada a configuração ideal sem que o usuário seja responsável pelo controle manual da qualidade da deduplicação.

Uma primeira alternativa para a configuração da deduplicação online é adaptar algumas das soluções propostas nos Capítulos 3 e 4 para o contexto da deduplicação online. Inicialmente, o esboço inicial visa adaptar as propostas para a seleção de um conjunto re-

duzido de pares (abordagem T3S) para identificar novos pares para serem rotulados pelo usuário. A abordagem T3S pode ser fundamental para identificar pares que ofereçam informações complementares ao atual conjunto de treinamento. Para isto, faz-se necessário promover uma refatoração da abordagem T3S para possibilitar o processamento online dos pares. A partir de um conjunto de treinamento atualizado pode ser possível manter atualizada a calibração dos métodos de classificação, evitando a intervenção do usuário.

REFERÊNCIAS

ANDERSON, Q. **Storm Real-Time Processing Cookbook**. [S.l.]: Packt Publishing Ltd, 2013.

ARASU, A.; GANTI, V.; KAUSHIK, R. Efficient exact set-similarity joins. In: VERY LARGE DATA BASES, 32. **Proceedings...** VLDB Endowment, 2006. p.918–929. (VLDB '06).

ARASU, A.; GOTZ, M.; KAUSHIK, R. On active learning of record matching packages. In: MANAGEMENT OF DATA, 2010., New York, NY, USA. **Proceedings...** ACM, 2010. p.783–794. (SIGMOD '10).

ARASU, A.; RÉ, C.; SUCIU, D. Large-Scale Deduplication with Constraints Using Dedupalog. In: IEEE INTERNATIONAL CONFERENCE ON DATA ENGINEERING, 2009., Washington, DC, USA. **Proceedings...** IEEE Computer Society, 2009. p.952–963. (ICDE '09).

AWEKAR, A.; SAMATOVA, N. F.; BREIMYER, P. Incremental all pairs similarity search for varying similarity thresholds. In: WORKSHOP ON SOCIAL NETWORK MINING AND ANALYSIS, 3., New York, NY, USA. **Proceedings...** ACM, 2009. p.1:1–1:10. (SNA-KDD '09).

BALCAN, M.-F.; BEYGELZIMER, A.; LANGFORD, J. Agnostic active learning. **Journal of Computer and System Sciences**, [S.l.], v.75, n.1, p.78–89, 2009.

BAXTER, R.; CHRISTEN, P.; CHURCHES, T. A Comparison of Fast Blocking Methods for Record Linkage. In: KDD 2003 WORKSHOPS. **Anais...** [S.l.: s.n.], 2003. p.25–27.

BAYARDO, R. J.; MA, Y.; SRIKANT, R. Scaling up all pairs similarity search. In: WORLD WIDE WEB, 16., New York, NY, USA. **Proceedings...** ACM, 2007. p.131–140. (WWW '07).

BELLARE, K. et al. Active sampling for entity matching. In: KDD, New York, NY, USA. **Anais...** ACM, 2012. p.1131–1139. (KDD '12).

BILENKO, M.; KAMATH, B.; MOONEY, R. Adaptive Blocking: learning to scale up record linkage. In: DATA MINING, 2006. ICDM '06. SIXTH INTERNATIONAL CONFERENCE ON. **Anais...** [S.l.: s.n.], 2006. p.87 –96.

BILENKO, M.; MOONEY, R. J. Adaptive duplicate detection using learnable string similarity measures. In: ACM SIGKDD INTERNATIONAL CONFERENCE ON KNOWLEDGE DISCOVERY AND DATA MINING, New York, NY, USA. **Proceedings...** ACM, 2003. p.39–48. (KDD '03).

CARVALHO, M. G. de et al. Learning to deduplicate. In: ACM/IEEE-CS JOINT CONFERENCE ON DIGITAL LIBRARIES, 6., New York, NY, USA. **Proceedings...** ACM, 2006. p.41–50. (JCDL '06).

CARVALHO, M. G. de et al. A Genetic Programming Approach to Record Deduplication. **IEEE Transactions on Knowledge and Data Engineering**, Los Alamitos, CA, USA, v.24, p.399–412, 2012.

CARVALHO, M. G. et al. Replica identification using genetic programming. In: ACM SYMPOSIUM ON APPLIED COMPUTING, 2008., New York, NY, USA. **Proceedings...** ACM, 2008. p.1801–1806. (SAC '08).

CAVNAR, W. B.; TRENKLE, J. M. N-gram-Based text categorization. In: IN PROC. OF SDAIR-94, 3RD ANNUAL SYMPOSIUM ON DOCUMENT ANALYSIS AND INFORMATION RETRIEVAL. **Anais...** [S.l.: s.n.], 1994. p.161–175.

CHANG, C.-C.; LIN, C.-J. LIBSVM: a library for support vector machines. **ACM Trans. Intell. Syst. Technol.**, New York, NY, USA, v.2, n.3, p.27:1–27:27, May 2011.

CHAUDHURI, S. et al. Robust and efficient fuzzy match for online data cleaning. In: ACM SIGMOD INTERNATIONAL CONFERENCE ON MANAGEMENT OF DATA, 2003., New York, NY, USA. **Proceedings...** ACM, 2003. p.313–324. (SIGMOD '03).

CHAUDHURI, S.; GANTI, V.; KAUSHIK, R. A Primitive Operator for Similarity Joins in Data Cleaning. In: DATA ENGINEERING, 2006. ICDE '06. PROCEEDINGS OF THE 22ND INTERNATIONAL CONFERENCE ON. **Anais...** [S.l.: s.n.], 2006. p.5.

CHRISTEN, P. Probabilistic Data Generation for Deduplication and Data Linkage. In: GALLAGHER, M.; HOGAN, J.; MAIRE, F. (Ed.). **Intelligent Data Engineering and Automated Learning - IDEAL 2005**. [S.l.]: Springer Berlin / Heidelberg, 2005. p.101–107. (Lecture Notes in Computer Science, v.3578). 10.1007/11508069_15.

CHRISTEN, P. Performance and scalability of fast blocking techniques for deduplication and data linkage. **Proc. VLDB Endow.**, Vienna, Austria, v.1, n.2, p.1253–1264, 2007.

CHRISTEN, P. Automatic record linkage using seeded nearest neighbour and support vector machine classification. In: ACM SIGKDD INTERNATIONAL CONFERENCE ON KNOWLEDGE DISCOVERY AND DATA MINING, 14., New York, NY, USA. **Proceedings...** ACM, 2008. p.151–159. (KDD '08).

CHRISTEN, P. Febrl: a freely available record linkage system with a graphical user interface. In: HDKM '08: PROCEEDINGS OF THE SECOND AUSTRALASIAN WORKSHOP ON HEALTH DATA AND KNOWLEDGE MANAGEMENT, Darlinghurst, Australia, Australia. **Anais...** Australian Computer Society: Inc., 2008. p.17–25.

CHRISTEN, P. A Survey of Indexing Techniques for Scalable Record Linkage and Deduplication. **IEEE Transactions on Knowledge and Data Engineering**, Los Alamitos, CA, USA, v.99, n.PrePrints, 2011.

CHRISTEN, P. A Survey of Indexing Techniques for Scalable Record Linkage and Deduplication. **IEEE Trans. on Knowl. and Data Eng.**, Piscataway, NJ, USA, v.24, n.9, p.1537–1555, Sept. 2012.

CHRISTEN, P.; CHURCHES, T. **Febrl - Freely extensible biomedical record linkage**. [S.l.]: DSpace at The Australian National University [http://dspace.anu.edu.au/dspace-oai/request] (Australia), 2002.

CHRISTEN, P.; CHURCHES, T.; HEGLAND, M. Febrl - A Parallel Open Source Data Linkage System. In: PAKDD. **Anais...** [S.l.: s.n.], 2004. p.638–647.

CHRISTEN, P.; GAYLER, R.; HAWKING, D. Similarity-aware indexing for real-time entity resolution. In: ACM CONFERENCE ON INFORMATION AND KNOWLEDGE MANAGEMENT, 18. **Proceedings...** [S.l.: s.n.], 2009. p.1565–1568.

CHRISTEN, P.; GOISER, K. Quality and Complexity Measures for Data Linkage and Deduplication. In: QUALITY MEASURES IN DATA MINING. **Anais...** Springer Berlin / Heidelberg, 2007. p.127–151. (Studies in Computational Intelligence, v.43).

COHEN, W. W.; RAVIKUMAR, P.; FIENBERG, S. E. A Comparison of String Metrics for Matching Names and Records. In: KDD WORKSHOP ON DATA CLEANING AND OBJECT CONSOLIDATION. **Anais...** [S.l.: s.n.], 2003.

COHN, D.; ATLAS, L.; LADNER, R. Improving generalization with active learning. **Machine Learning**, [S.l.], v.15, n.2, p.201–221, 1994.

COSTA, G. et al. Data De-duplication: a review. In: BIBA, M.; XHAFA, F. (Ed.). **Learning Structure and Schemas from Documents**. [S.l.]: Springer Berlin / Heidelberg, 2011. p.385–412. (Studies in Computational Intelligence, v.375).

DAL BIANCO, G. et al. Tuning large scale deduplication with reduced effort. In: INTERNATIONAL CONFERENCE ON SCIENTIFIC AND STATISTICAL DATABASE MANAGEMENT, 25., New York, NY, USA. **Proceedings...** ACM, 2013. p.18:1–18:12. (SSDBM).

DAL BIANCO, G.; GALANTE, R.; HEUSER, C. A. A fast approach for parallel deduplication on multicore processors. In: ACM SYMPOSIUM ON APPLIED COMPUTING, 2011., New York, NY, USA. **Proceedings...** ACM, 2011. p.1027–1032. (SAC '11).

DEAN, J.; GHEMAWAT, S. MapReduce: simplified data processing on large clusters. **Commun. ACM**, New York, NY, USA, v.51, p.107–113, Jan. 2008.

DEMPSTER, A. P.; LAIRD, N. M.; RUBIN, D. B. Maximum likelihood from incomplete data via the EM algorithm. **JOURNAL OF THE ROYAL STATISTICAL SOCIETY, SERIES B**, [S.l.], v.39, n.1, p.1–38, 1977.

DORNELES, C. F. et al. A strategy for allowing meaningful and comparable scores in approximate matching. In: CIKM. **Anais...** [S.l.: s.n.], 2007. p.303–312.

DORNELES, C. F. et al. A strategy for allowing meaningful and comparable scores in approximate matching. **Inf. Syst.**, [S.l.], v.34, n.8, p.673–689, 2009.

DORNELES, C. F.; GONçALVES, R.; SANTOS MELLO, R. dos. Approximate data instance matching: a survey. **Knowl. Inf. Syst.**, New York, NY, USA, v.27, p.1–21, April 2011.

ELMAGARMID, A. K.; IPEIROTIS, P. G.; VERYKIOS, V. S. Duplicate Record Detection: a survey. **IEEE Trans. on Knowl. and Data Eng.**, Piscataway, NJ, USA, v.19, n.1, p.1–16, 2007.

ERTEKIN, S. et al. Learning on the border: active learning in imbalanced data classification. In: ACM CONFERENCE ON CONFERENCE ON INFORMATION AND KNOWLEDGE MANAGEMENT, New York, NY, USA. **Proceedings...** ACM, 2007. p.127–136. (CIKM '07).

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, 1969.

FREITAS, J. de et al. Active Learning Genetic programming for record deduplication. In: EVOLUTIONARY COMPUTATION (CEC), 2010 IEEE CONGRESS ON. **Anais...** [S.l.: s.n.], 2010. p.1 –8.

FREUND, Y. et al. Selective sampling using the query by committee algorithm. **Machine learning**, [S.l.], v.28, n.2-3, p.133–168, 1997.

GEMMELL, J.; RUBINSTEIN, B. I. P.; CHANDRA, A. K. Improving Entity Resolution with Global Constraints. **CoRR**, [S.l.], v.abs/1108.6016, 2011.

GILES, C. L.; BOLLACKER, K. D.; LAWRENCE, S. CiteSeer: an automatic citation indexing system. In: ACM CONFERENCE ON DIGITAL LIBRARIES. **Proceedings...** [S.l.: s.n.], 1998. p.89–98.

GILL. **Methods For Automatic Record Matching And Linking And Their Use In National Statistics**. [S.l.: s.n.], 2001.

GODOI, T. A. et al. A Relevance Feedback Approach for the Author Name Disambiguation Problem. In: ACM/IEEE-CS JOINT CONFERENCE ON DIGITAL LIBRARIES, 13., New York, NY, USA. **Proceedings...** ACM, 2013. p.209–218. (JCDL '13).

GRAVANO, L. et al. Approximate String Joins in a Database (Almost) for Free. In: INTERNATIONAL CONFERENCE ON VERY LARGE DATA BASES, 27., San Francisco, CA, USA. **Proceedings...** Morgan Kaufmann Publishers Inc., 2001. p.491–500. (VLDB '01).

GUSFIELD, D. **Algorithms on strings, trees, and sequences**: computer science and computational biology. New York, NY, USA: Cambridge University Press, 1997.

HADJIELEFTHERIOU, M.; SRIVASTAVA, D. Approximate String Processing. **Found. Trends databases**, Hanover, MA, USA, v.2, p.267–402, April 2011.

HAGHANI, P.; MICHEL, S.; ABERER, K. Distributed similarity search in high dimensions using locality sensitive hashing. In: INTERNATIONAL CONFERENCE ON EXTENDING DATABASE TECHNOLOGY: ADVANCES IN DATABASE TECHNOLOGY, 12., New York, NY, USA. **Proceedings...** ACM, 2009. p.744–755. (EDBT '09).

HASSANZADEH, O. et al. Helix: online enterprise data analytics. In: WORLD WIDE WEB, 20. **Proceedings...** [S.l.: s.n.], 2011. p.225–228.

HERNáNDEZ, M. A.; STOLFO, S. J. The merge/purge problem for large databases. In: SIGMOD '95: PROCEEDINGS OF THE 1995 ACM SIGMOD INTERNATIONAL CONFERENCE ON MANAGEMENT OF DATA, New York, NY, USA. **Anais...** ACM, 1995. p.127–138.

HERNáNDEZ, M. A.; STOLFO, S. J. The merge/purge problem for large databases. In: ACM SIGMOD INTERNATIONAL CONFERENCE ON MANAGEMENT OF DATA, 1995., New York, NY, USA. **Proceedings...** ACM, 1995. p.127–138. (SIGMOD '95).

ISELE, R.; BIZER, C. Active learning of expressive linkage rules using genetic programming. **Web Semantics: Science, Services and Agents on the World Wide Web**, [S.l.], 2013.

KARTHIGHA, M.; ANAND, S. K. A Survey on Removal of Duplicate Records in Database. **Indian Journal of Science and Technology**, [S.l.], v.6, n.4, p.4306–4311, 2013.

KIM, H.; LEE, D. Parallel linkage. In: CIKM '07: PROCEEDINGS OF THE SIXTEENTH ACM CONFERENCE ON CONFERENCE ON INFORMATION AND KNOWLEDGE MANAGEMENT, New York, NY, USA. **Anais...** ACM, 2007. p.283–292.

KOPCKE, H.; RAHM, E. Training selection for tuning entity matching. In: QDB/MUD. **Anais...** [S.l.: s.n.], 2008. p.3–12.

KöPCKE, H.; RAHM, E. Frameworks for entity matching: a comparison. **Data Knowl. Eng.**, Amsterdam, The Netherlands, The Netherlands, v.69, p.197–210, February 2010.

KOPCKE, H.; THOR, A.; RAHM, E. Evaluation of entity resolution approaches on real-world match problems. **PVLDB**, [S.l.], v.3, n.1, p.484–493, 2010.

KOUDAS, N.; SARAWAGI, S.; SRIVASTAVA, D. Record linkage: similarity measures and algorithms. In: ACM SIGMOD INTERNATIONAL CONFERENCE ON MANAGEMENT OF DATA, 2006., New York, NY, USA. **Proceedings...** ACM, 2006. p.802–803. (SIGMOD '06).

KOZA, J.; POLI, R. Genetic Programming. In: BURKE, E. K.; KENDALL, G. (Ed.). **Search Methodologies**. [S.l.]: Springer US, 2005. p.127–164.

LAWRENCE, S.; GILES, C. L.; BOLLACKER, K. D. Digital Libraries and Autonomous Citation Indexing. **IEEE Computer**, [S.l.], v.32, n.6, p.67–71, 1999.

LEE, K.-H. et al. Parallel data processing with MapReduce: a survey. **ACM SIGMOD Record**, [S.l.], v.40, n.4, p.11–20, 2012.

MANNING, C. D.; RAGHAVAN, P.; SCHTZE, H. **Introduction to Information Retrieval**. New York, NY, USA: Cambridge University Press, 2008.

MCCALLUM, A.; NIGAM, K.; UNGAR, L. H. Efficient clustering of high-dimensional data sets with application to reference matching. In: KDD '00: ACM SIGKDD. **Anais...** ACM, 2000.

NAVARRO, G. A guided tour to approximate string matching. **ACM Comput. Surv.**, New York, NY, USA, v.33, p.31–88, March 2001.

NEWCOMBE, H. B. et al. Automatic linkage of vital records. **Science (New York, N.Y.)**, [S.l.], v.130, n.3381, p.954–959, Oct. 1959.

OLIVEIRA, P.; RODRIGUES, F.; HENRIQUES, P. R. A Formal Definition of Data Quality Problems. In: IQ. **Anais...** MIT, 2005.

PASULA, H. M. **Identity uncertainty**. 2003. Tese (Doutorado em Ciência da Computação) — . AAI3121641.

PETRICEK, V. et al. A Comparison of On-Line Computer Science Citation Databases. In: RESEARCH AND ADVANCED TECHNOLOGY FOR DIGITAL LIBRARIES, 2005. **Anais...** Springer Berlin / Heidelberg, 2005. v.3652, p.438–449.

PIERRE; MICHAUD. Clustering techniques. **Future Generation Computer Systems**, [S.l.], v.13, n.23, p.135 – 147, 1997. Data Mining.

QUINLAN, J. Improved Use of Continuous Attributes in C4. 5. **Journal of Artificial Intelligence Research**, [S.l.], v.4, p.77–90, 1996.

SANTOS, J. B. dos et al. Automatic threshold estimation for data matching applications. In: BRAZILIAN SYMPOSIUM ON DATABASES, 23., Porto Alegre, Brazil, Brazil. **Proceedings...** Sociedade Brasileira de Computação, 2008. p.106–119. (SBBD '08).

SANTOS, J. B. dos et al. Automatic threshold estimation for data matching applications. **Information Sciences**, [S.l.], v.181, n.13, p.2685 – 2699, 2011. Including Special Section on Databases and Software Engineering.

SARAWAGI, S.; BHAMIDIPATY, A. Interactive deduplication using active learning. In: ACM SIGKDD INTERNATIONAL CONFERENCE ON KNOWLEDGE DISCOVERY AND DATA MINING, New York, NY, USA. **Proceedings...** ACM, 2002. p.269–278. (KDD '02).

SARAWAGI, S.; KIRPAL, A. Efficient set joins on similarity predicates. In: ACM SIGMOD INTERNATIONAL CONFERENCE ON MANAGEMENT OF DATA, 2004., New York, NY, USA. **Proceedings...** ACM, 2004. p.743–754. (SIGMOD '04).

SCHMIDBERGER, G.; FRANK, E. Unsupervised discretization using tree-based density estimation. In: **Knowledge Discovery in Databases**: pkdd 2005. [S.l.]: Springer, 2005. p.240–251.

SCHROEDER, T.; GODDARD, S.; RAMAMURTHY, B. Scalable Web server clustering technologies. **Network, IEEE**, [S.l.], v.14, n.3, p.38 –45, may/jun 2000.

SETTLES, B. Active learning literature survey. **University of Wisconsin, Madison**, [S.l.], 2010.

SILVA, R.; GONCALVES, M.; VELOSO, A. Rule-Based Active Sampling for Learning to Rank. In: GUNOPULOS, D. et al. (Ed.). **Machine Learning and Knowledge Discovery in Databases**. [S.l.]: Springer Berlin Heidelberg, 2011. p.240–255. (Lecture Notes in Computer Science, v.6913).

VELOSO, A. A. et al. Learning to rank at query-time using association rules. In: ACM SIGIR CONFERENCE ON RESEARCH AND DEVELOPMENT IN INFORMATION RETRIEVAL, 31. **Proceedings...** [S.l.: s.n.], 2008. p.267–274.

VERNICA, R.; CAREY, M. J.; LI, C. Efficient parallel set-similarity joins using MapReduce. In: SIGMOD '10: PROCEEDINGS OF THE 2010 INTERNATIONAL CONFERENCE ON MANAGEMENT OF DATA, New York, NY, USA. **Anais...** ACM, 2010. p.495–506.

WANG, J.; LI, G.; FE, J. Fast-join: an efficient method for fuzzy token matching based string similarity join. In: DATA ENGINEERING (ICDE), 2011 IEEE 27TH INTERNATIONAL CONFERENCE ON. **Anais...** [S.l.: s.n.], 2011. p.458 –469.

WINKLER, W. E. **The State of Record Linkage and Current Research Problems**. [S.l.]: Statistical Research Division, U.S. Census Bureau, 1999.

WINKLER, W. E. **Approximate string comparator search strategies for very large administrative lists**. [S.l.]: STATISTICAL RESEARCH DIVISION, U.S. CENSUS BUREAU, 2005.

XIAO, C. et al. Efficient similarity joins for near duplicate detection. In: WORLD WIDE WEB, 17., New York, NY, USA. **Proceedings...** ACM, 2008. p.131–140. (WWW '08).

XIAO, C. et al. Efficient similarity joins for near-duplicate detection. **ACM Trans. Database Syst.**, New York, NY, USA, v.36, n.3, p.15:1–15:41, Aug. 2011.