Uma Análise Experimental do Impacto da Seleção de Atributos em Processos de Resolução de Entidades

Levy de Souza Silva[1,2], **Gabrielle Karine Canalle**[1], **Ana Carolina Salgado**[1], **Bernadette Farias Lóscio**[1], **Mirella M. Moro**[2]

[1]Programa de Pós Graduação em Ciência da Computação - Centro de Informática (CIn) Universidade Federal de Pernambuco (UFPE)

[2]Departamento de Ciência da Computação - Universidade Federal de Minas Gerais

{lss9, gkc, acs, bfl}@cin.ufpe.br, mirella@dcc.ufmg.br

Resumo. *Resolução de Entidades (RE) é a tarefa de identificar instâncias duplicadas em conjuntos de dados por meio de um processo de várias etapas. Um ponto em comum entre suas etapas é a seleção de atributos. Apesar de existirem trabalhos de seleção de atributos na RE, há uma falta de estudos experimentais que analisem o impacto da seleção de atributos no processo completo. Esta análise é importante pois a eficácia da RE varia conforme os atributos adotados. Assim, este trabalho aborda tal lacuna por meio de experimentos em dados reais e sintéticos de vários domínios. Por fim, os resultados mostram que a seleção de atributos afeta a eficácia da RE em até* 92%.

Abstract. *Entity Resolution is the task of identifying duplicate records in datasets by a multi-step process. A common aspect involving its steps is the attribute selection, and there is no experimental work evaluating the attribute selection impact over the complete ER process. Such an evaluation is important because the ER effectiveness varies according to the selected attributes. Therefore, we cover this gap by performing experiments over real and synthetic datasets from different domains. Finally, the results show attribute selection affects the ER effectiveness by up to* 92%.

1. Introdução

Resolução de Entidades (RE) é a tarefa de identificar instâncias duplicadas de entidades em conjuntos de dados. É um problema muito estudado com várias aplicações. Por exemplo, no contexto de Integração de Dados, a RE é aplicada para encontrar ofertas semelhantes de produtos em dados na *Web* [Barbosa et al. 2018]. Aplicações incluem Web Sites de comparação de preços que combinam dados oriundos de mais de 300 lojas, como o Buscapé[1] e o Zoom[2]. Em domínios particulares, a RE é utilizada para encontrar instâncias duplicadas em dados médicos, financeiros, governamentais, entre outros [Konda et al. 2019]. A RE também é imprescindível em áreas de Limpeza e Qualidade de Dados [Christen 2012].

O processo de RE é composto das etapas de *indexação*, *agrupamento*, *comparação*, *classificação* e *avaliação* [Christen 2012]. Um ponto em comum entre algumas etapas é a seleção de atributos, que encontra um subconjunto de atributos relevantes

[1]http://www.buscape.com.br

[2]https://www.zoom.com.br

para uma tarefa. Na RE, os atributos são utilizados para indexar, agrupar e comparar os registros, e os desafios incluem: escolher o melhor atributo para definir as chaves de bloco; selecionar o conjunto de atributos correto para agrupar as instâncias; e definir o grupo de atributos ideal e a importância de um atributo para comparar um par de registros. Alguns trabalhos abordam tais desafios e propõem métodos de seleção de atributos relevantes para etapas específicas da RE [Canalle et al. 2017, Silva et al. 2018]. No entanto, geralmente os atributos são escolhidos manualmente por um usuário especialista no domínio dos dados, o que requer tempo e aumenta o custo total do processo [Christen 2006, Christen 2012, Papadakis et al. 2015].

O problema da seleção de atributos tem recebido muita atenção, sobretudo nas áreas de Mineração de Dados e Aprendizagem de Máquina. Mas, na RE, há uma falta de estudos experimentais que avaliem o impacto da seleção de atributos em diferentes cenários e etapas do processo. Esta avaliação é necessária porque a eficácia da RE pode aumentar ou diminuir dependendo dos atributos adotados em cada uma das etapas. Por exemplo, na indexação, a eficácia do melhor atributo difere da do pior em 92% (resultados da Seção 5.1). Além disso, todos os algoritmos, métodos e funções aplicados no processo de RE dependem previamente da seleção de atributos (e.g., o algoritmo *Standard Blocking* que cria um conjunto de blocos utilizando os atributos disponíveis [Christen 2012]). No mais, os experimentos também proveem direcionamentos relacionados à eficácia da combinação de diferentes estratégias e atributos em vários cenários, bem como à importância da seleção de atributos no processo em comparação com outros fatores.

Sendo assim, o objetivo deste trabalho é analisar de forma experimental o impacto da seleção de atributos considerando o processo completo de RE. Para tal, um conjunto de experimentos são realizados utilizando dados reais e sintéticos de diferentes domínios e tipos de atributos. As avaliações experimentais são baseadas em quatro questões de pesquisa que cobrem todo o processo de RE (Seção 4). Os experimentos mostram que a seleção de atributos influencia a eficácia da RE em todas as etapas. As contribuições deste artigo são: (i) um projeto fatorial que avalia qual fator possui maior efeito no resultado da RE; (ii) uma análise da influência do atributo de indexação nos métodos de indexação *Schema-Agnostic* e *Configurations-Based*; (iii) uma avaliação do impacto da seleção de atributos nos algoritmos de agrupamento *Standard Blocking* e *Sorted Neighborhood*; (iv) uma análise do efeito da combinação de atributos na etapa de comparação; e (v) um conjunto de implementações e *datasets* disponíveis publicamente.[3]

O restante deste trabalho está organizado da seguinte forma. A Seção 2 apresenta os trabalhos relacionados. O processo de RE é detalhado na Seção 3. A Seção 4 descreve a metodologia experimental. Os resultados e análises experimentais são exibidos na Seção 5 e, finalmente, as conclusões e os trabalhos futuros são apontados na Seção 6.

2. Trabalhos Relacionados

Os estudos de RE são divididos conforme suas etapas. Assim, esta seção detalha trabalhos relevantes de cada uma das etapas e aponta as principais diferenças frente ao nosso estudo.

Indexação. Existem pelo menos duas estratégias de indexação: *Schema-Agnostic* e *Configuration-Based*, as quais são comparadas em Papadakis et al. (2015). Os autores

[3] `http://www.dcc.ufmg.br/~mirella/projs/deduplica`

realizam uma avaliação experimental dessas técnicas combinadas com nove métodos de blocagem do estado da arte. Os autores concluem que a técnica *Schema-Agnostic* oferece maior robustez para definição das chaves porque é não-supervisionada e independente do domínio. Em seguida, Silva et al. (2017) avaliam as funções de indexação *Soundex* e *Suffix* combinadas com algoritmos tradicionais de RE em 11 *datasets* de vários domínios. Eles concluem que as funções *Soundex* e *Suffix* têm resultados semelhantes, sem diferença significativa em termos da *F-Measure*. Silva et al. (2018) propõem um método de seleção de atributos para a indexação que utiliza métricas como densidade, repetição e distintividade para classificar os atributos. Os autores realizam experimentos que demostram a eficácia do método proposto considerando vários domínios de dados e tipos de atributos.

Agrupamento. Depois de indexados, os registros são agrupados utilizando um dos vários métodos existentes. Em Baxter et al. (2003) os métodos *Bigram Indexing* e *Canopy Clustering* são comparados com as abordagens *Standard Blocking* e *Sorted Neighborhood*, sendo *Bigram Indexing* o mais eficiente e de melhor acurácia. Após, Draisbach e Naumann (2009) comparam os métodos de agrupamento com os algoritmos de janela deslizante (e.g, *Sorted Neighborhood*). Os experimentos demostram que os algoritmos de janela são melhores que os métodos de agrupamento no quesito eficiência. Um estudo mais completo sobre agrupamento é realizado por Christen (2012), o qual apresenta um *survey* com avaliação experimental de 12 variações de seis técnicas de agrupamento existentes. No estudo, os métodos analisados são: *Standard Blocking, Sorted Neighborhood, Suffix Array-Based Indexing, Canopy Clustering, Q-gram-Based Indexing* e *String-Map-Based Indexing*. Resultados mostram que a técnica *Q-gram-Based Indexing* é uma das mais lentas e não é adaptável para grandes conjuntos de dados, e as abordagens tradicionais são as mais rápidas (i.e., blocos e vizinhos). Por fim, Caldeira e Ferreira (2018) apresentam um método para blocagem e processamento dos blocos considerando a relevância dos termos (meta-blocagem). O método proposto supera técnicas do estado da arte em termos de eficácia e reduz o tempo de criação dos blocos pela metade.

Comparação e Classificação. No fim do processo de RE, um par de registros é classificado. Assim, as etapas de comparação e classificação são complementares, pois depois que os atributos são comparados, a classificação é baseada em um limiar de similaridade entre os atributos. Várias funções foram propostas na literatura. Cohen et al. (2003), por exemplo, analisam as funções *TFIDF, SoftTFIDF, Levenshtein, Scaled Levenstein, Jaro, Jaro-Winkler, Jaccard* e *NaiveAvgOverlap* em dados de nomes pessoais. Os resultados mostram que o melhor método para comparação de nomes pessoais é uma versão escalada do algoritmo de *Levenshtein*. Em seguida, Christen (2006) compara 20 funções de similaridade incluindo: *Soundex, Phonex, phonix, Jaro, Winkler* e *Edit Distance*. Os experimentos utilizam quatro *datasets* contendo nomes pessoais. Segundo o autor, a melhor função de classificação não é clara. Entretanto, a técnica *Simple Phonex* tem desempenho melhor que as técnicas *Complex Phonix* e *Double-Metaphone*. Além disso, os algoritmos de *Jaro* e *Jaro Winkler* são eficazes em todos os conjuntos de dados utilizados. Considerando a seleção de atributos, Canalle et al. (2017) apresentam uma abordagem que seleciona atributos relevantes para a etapa de comparação, utilizando critérios como densidade, repetição e qualidade da fonte. Experimentos são executados em dados reais e sintéticos com diferentes cenários de dados duplicados. Os resultados demonstram que a estratégia proposta seleciona atributos eficazes para a comparação em todos os cenários.

Tabela 1. Visão geral dos trabalhos relacionados frente ao nosso estudo.

Trabalho	Indexação	Agrupamento	Comparação	Classificação	Avaliação do Impacto do Atributo na Etapa
Silva et al. (2018)	X				Sim
Silva et al. (2017)	X				Não
Papadakis et al. (2015)	X				Não
Caldeira e Ferreira (2018)		X			Não
Christen (2012)		X			Não
Baxter et al. (2003)		X			Não
Canalle et al. (2017)			X		Sim
Christen (2006)			X	X	Não
Cohen et al. (2003)			X	X	Não
Nosso Estudo	X	X	X	X	Sim

Finalmente, a Tabela 1 apresenta um resumo dos trabalhos comparando-os com o nosso estudo. Em sua maioria, os estudos analisam apenas uma etapa da RE isoladamente. Por exemplo, Papadakis et al. (2015) investigam só as funções de indexação, enquanto nosso trabalho analisa em um mesmo ambiente experimental o processo completo da RE. Ademais, considerando a seleção de atributos, a maioria dos trabalhos citados não considera o impacto da seleção automática de atributos nos experimentos, pois os atributos são escolhidos manualmente por um especialista. Apesar de existirem trabalhos recentes, e.g., Silva et al. (2018) e Canalle et al. (2017), esses estudos não consideram avaliações experimentais para medir o impacto do atributo no processo. Nosso estudo difere, pois investigamos o impacto da tarefa de seleção de atributos em todo o processo de RE.

3. Processo de Resolução de Entidades

O processo de RE é dividido em *Indexação*, *Agrupamento*, *Comparação*, *Classificação* e *Avaliação*. Um ponto em comum entre algumas etapas é a tarefa de seleção de atributos. Nesse sentido, esta seção apresenta detalhadamente o processo de RE conforme Figura 1 e discute os problemas relacionados à seleção de atributos no processo.

(1) - Indexação. Inicialmente, todos os registros são indexados por um valor de chave de bloco (do termo em inglês *Block Key Value* - BKV). Para tal, atributos são escolhidos e seus valores são utilizados como chave (*Schema-Agnostic*), ou uma regra de codificação é aplicada no valor dos atributos para gerar a chave (*Configurations-Based*). No fim da etapa, com exceção das instâncias que contêm valores nulos nos atributos, cada registro está associado a um BKV. Existem diversas técnicas para criar um BKV. Entretanto, uma das principais é a *Soundex* que codifica os valores baseado na pronúncia [Christen 2012]. Outra opção é a *Suffix*, que cria sufixos de tamanho K a partir de um valor.

(2) - Agrupamento. Depois da indexação, cada BKV define grupos de registros similares. Sem o agrupamento, os registros são comparados todos com todos (i.e., *Naive Duplicate Detection*). Existem diferentes abordagens para esta etapa, entretanto os principais métodos incluem comparação por blocos e vizinhos mais próximos. O algoritmo ***Standard Blocking* (SB)** cria blocos que agrupa registros semelhantes. Logo, os registros são comparados entre eles apenas dentro dos blocos. Os grupos são definidos de acordo com o BKV dos registros. Diferente deste, o algoritmo ***Sorted Neighborhood* (SN)** combina os registros por meio de uma chave ordenada, que é similar a uma chave de bloco. Porém, antes de executar as comparações, todos os registros são ordenados pelo BKV. Depois, uma janela deslizante de tamanho $w > 1$ percorre todos os registros de D, e o primeiro registro da janela é comparado com todos os outros dentro da mesma janela.

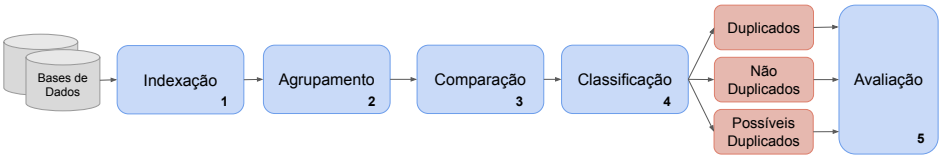


Figura 1. Processo de Resolução de Entidades (adaptado de Christen (2012))

(3) - Comparação. Após o agrupamento, os registros de cada grupo são comparados com todos os outros do mesmo grupo, par a par, utilizando medidas de similaridade. Geralmente, a semelhança entre duas instâncias é calculada comparando-se vários atributos, e quanto maior a similaridade dos atributos, mais provável que sejam instâncias de uma mesma entidade. Ou seja, uma função de similaridade calcula a correspondência entre dois valores e retorna um valor no intervalo $[0, 1]$, no qual 1 representa semelhança máxima. Funções populares são *Jaro*, *Jaro Winkler* e *Levenshtein* [Christen 2006].

(4) - Classificação. A partir da comparação, os pares de registros são classificados como *duplicados*, *não duplicados* e *possíveis duplicados*. A terceira classe necessita de uma avaliação mais detalhada, geralmente feita por um usuário especialista no domínio dos dados. Para classificá-los, existem duas abordagens: não supervisionada e supervisionada. Na primeira, os pares são classificados com base apenas na similaridade entre os atributos. Então, a forma mais simples é aplicar um limiar *(threshold)* sobre este valor. Na segunda, um conjunto de treinamento com pares verdadeiros e não correspondentes é utilizado, formando um classificador supervisionado [Christen 2012].

(5) - Avaliação. Finalmente, a etapa de avaliação analisa a eficácia dos algoritmos por meio de métricas como *Precision*, *Recall*, *F-Measure* [Christen 2012].

Seleção de Atributos na RE. Um ponto em comum entre as etapas indexação, agrupamento e comparação é a seleção de atributos. Um dos problemas é escolher atributos que proporcionem os melhores resultados para o processo em termos de eficácia. Por exemplo, considere o cenário em que as etapas de RE mostradas anteriormente são executadas sobre os dados exibidos na Tabela 2. Estes dados são oriundos de duas fontes identificadas pela coluna *ID* e referem-se a três Filmes (coluna *Filme*). Assim, o objetivo é encontrar os filmes duplicados nestas fontes. Considerando a seleção de atributos na *indexação/agrupamento*, a seguinte situação pode ocorrer:[4] (*i*) indexando por *Fonte* são definidos um bloco para os registros de *IMDb* e outro para *TheMovieDB* – Filmes 1 e 2 não são comparados pois estão em blocos distintos, o que prejudica a eficácia; (*ii*) indexando por *Diretor* ou *Título* terá maior eficácia, pois o processo compara os Filmes 1 e 2, bem como 4 e 5 que agora estão no mesmo bloco. Para os demais atributos, as situações são análogas. De forma semelhante, na *comparação/classificação*, têm-se os problemas: (*i*) comparando por meio dos atributos *Título*, *Duração* e *Gênero*, os Filmes 1 e 2, e 4 e 5 não são considerados cópia, pois não existe similaridade nos atributos *Duração* e *Gênero*; (*ii*) comparando por *Título* e *Diretor*, todas as duplicatas são identificadas corretamente porque os valores dos atributos são similares. Para outras combinações de atributos, situações

[4]Utilizando o próprio valor do atributo como chave de bloco.

Tabela 2. Identificando registros duplicados em duas fontes de dados

Filme	ID	Fonte	Título	Ano	Diretor	Duração	Gênero	Indicação
A	1	IMDb	Robin Hood	2010	Ridley Scott	2h 20m	Aventura	14
	2	TheMovieDB	Robin Hood	2010	Ridley L. Scott	2h 22m	Ação	Null
B	3	IMDb	Avatar	2009	James Cameron	2h 42m	Fantasia	12
C	4	IMDb	The Matrix	1999	Lana Wachouski	2h 20m	Ação	12
	5	TheMovieDB	Matrix	1999	Lana Wachouski	2h 16m	Ficção Científica	Null

semelhantes ocorrem. Por fim, estes problemas destacam a importância de selecionar atributos relevantes para cada uma das etapas do processso de RE.

4. Metodologia Experimental

Para avaliar a seleção de atributos na RE, as seguintes questões são adotadas: **(Q1)** Qual fator possui maior impacto no processo de RE entre a função de indexação, o atributo de indexação e o algoritmo de agrupamento? **(Q2)** O mesmo atributo de indexação é eficaz nos métodos *Schema-Agnostic* e *Configurations-Based*? **(Q3)** Os resultados da RE são eficazes utilizando o mesmo atributo de indexação nos algoritmos *Standard Blocking* e *Sorted Neighborhood*? **(Q4)** O conjunto de atributos utilizados na comparação afeta o resultado da RE? Destas questões são criados os cenários descritos a seguir (Figura 2).

Cenário 1 - Projetos Fatoriais. Este cenário considera projetos fatoriais para analisar o efeito dos fatores sobre o processo de RE. O projeto fatorial avalia o efeito de k fatores sobre uma variável y [Jain 1992]. Desse modo, um projeto fatorial $2^k r$ com $(k = 3)$ e $(r = 10)$ é executado, isto é, cada configuração do projeto é analisada sobre dez *datasets* sintéticos. As replicações são definidas com uma confiança de 95% e um erro máximo de 5%. Os fatores examinados são: o atributo de indexação, a função de indexação e o algoritmo de agrupamento, e a métrica *F-Measure* é a variável resposta Y.

Cenário 2 - Métodos de Indexação. Este cenário analisa a eficácia dos métodos *Agnostic* e *Configurations-Based*. Geralmente, apenas um atributo é adotado para indexar. Assim, o valor do atributo é utilizado como BKV, ou a codificação *Soundex* é aplicada sobre o valor do atributo para gerar os BKVs. Nas outras etapas, os algoritmos *Standard Blocking* ou *Sorted Neighborhood* e *Jaro Winkler* são aplicados. Especificamente, o objetivo é avaliar se o mesmo atributo de indexação tem resultados eficazes nos dois métodos. Para tal, a *F-Measure* é computada para cada configuração de indexação.

Cenário 3 - Algoritmos de Agrupamento. Neste cenário a RE é executada variando tanto o atributo de indexação quanto o algoritmo de agrupamento, ou seja, cada atributo é combinado com cada algoritmo. Na comparação, a função de *Jaro Winkler* é utilizada. Por fim, a *F-Measure* é computada para cada atributo em cada algoritmo.

Cenário 4 - Comparação das Instâncias. Este cenário executa a RE com o algoritmo *Naive Duplicate Detection* variando o grupo de atributos na comparação. A função de *Levenshtein* é utilizada para comparar os atributos. Alguns *datasets* contêm muitos atributos. Assim, combinações de até cinco atributos que melhor descrevem a entidade são consideradas. Por fim, a *F-measure* é calculada para cada combinação.

***Datasets* Experimentais.** Dados reais e sintéticos com vários domínios e tipos de atributos são utilizados para experimentação dos cenários. Os sintéticos são criados com o

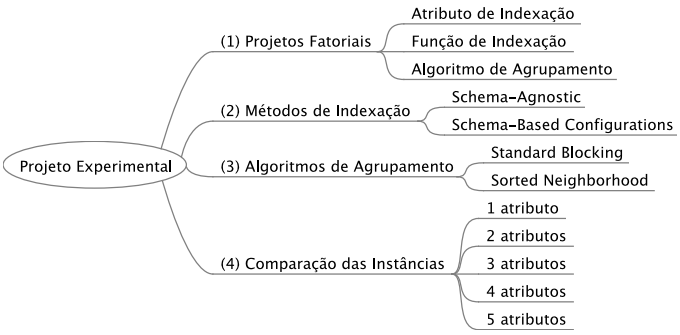


Figura 2. Projeto experimental com as dimensões avaliadas e variáveis em cada dimensão

Data Set Generator Program [Christen 2012], e os reais são extraídos do *DuDe toolkit*[5]: *CORA*, *Restaurant* e *CD Information*. Os *datasets* escolhidos são amplamente utilizados em pesquisas de RE, e.g., Canalle et al. (2017) , Christen (2012b), Papadakis et al. (2015).

5. Análise Experimental

Nesta seção, os resultados experimentais são apresentados e discutidos conforme questões de pesquisas elencadas na seção de metodologia experimental, como segue.

5.1. Avaliação do Impacto dos Fatores no Processo de RE

O resultado da RE depende da escolha de vários fatores incluindo o atributo de indexação, a função de indexação e o algoritmo de agrupamento. Assim, esta seção apresenta o resultado do projeto fatorial $2^3 10$ que avalia quais dos três fatores da RE têm maior efeito. O objetivo é responder a questão **Q1**: Qual fator possui maior impacto no processo de RE entre a função de indexação, o atributo de indexação e o algoritmo de agrupamento?

Para abordar **Q1**, os seguintes fatores e níveis são analisados: (*Fator A*) as funções de indexação *Soundex* e *Suffix*, (*Fator B*) os algoritmos de agrupamento *Standard Blocking* e *Sorted Neighborhood*, e (*Fator C*) o melhor e o pior atributo de indexação conforme a *F-Measure* (*Given Name* e *Address2*). A Tabela 3 exibe os valores da *F-Measure* para cada configuração do projeto fatorial ordenadas pelo fator *C*. O maior valor da *F-Measure* é alcançado no experimento *4*, $0,73 \pm (0,01)$, onde o atributo de indexação *Given Name* é utilizado. Por outro lado, o experimento *6* tem o resultado mais ineficaz para a RE - atributo *Address2* e *F-Measure* de $0,38 \pm (0,03)$. O melhor resultado de *Given Name* e o pior de resultado *Address2* diferem em aproximadamente $0,35$ em termos da *F-Measure* (experimentos *4* e *6* respectivamente), ou seja, quando o atributo *Given Name* é utilizado na indexação, a eficácia do processo aumenta em cerca de 92%.

Comparando os níveis da função de indexação (*Soundex* e *Suffix*) e do algoritmo de agrupamento (blocos e vizinhos), os resultados mostram que a mudança das técnicas não produz uma variação significante da *F-Measure* porque as diferenças são mínimas. Por exemplo, os resultados do processo são eficazes utilizando o algoritmo *Standard Blocking* (experimento *1*) e o algoritmo *Sorted Neighborhood* (experimento *3*), e a diferença é apenas $0,01$ da *F-Measure*. O mesmo acontece com as funções de indexação *Soundex*

[5] `https://hpi.de/naumann/projects/data-quality-and-cleansing/dude-duplicate-detection`

Tabela 3. Configurações do Projeto Fatorial

#	Fator (A) Indexação	Fator (B) Algoritmo	Fator (C) Atributo	*F-Measure*
1	Soundex	SB	Given Name	$0,71 \pm 0,01$
2	Suffix	SB	Given Name	$0,67 \pm 0,01$
3	Soundex	SN	Given Name	$0,72 \pm 0,01$
4	Suffix	SN	Given Name	$0,73 \pm 0,01$
5	Soundex	SB	Address2	$0,39 \pm 0,03$
6	Suffix	SB	Address2	$0,38 \pm 0,03$
7	Soundex	SN	Address2	$0,43 \pm 0,03$
8	Suffix	SN	Address2	$0,40 \pm 0,03$

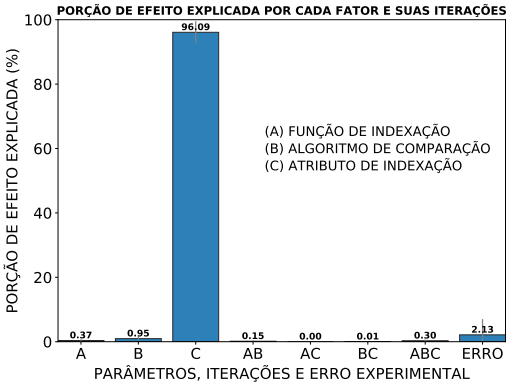


Figura 3. Efeito dos Fatores

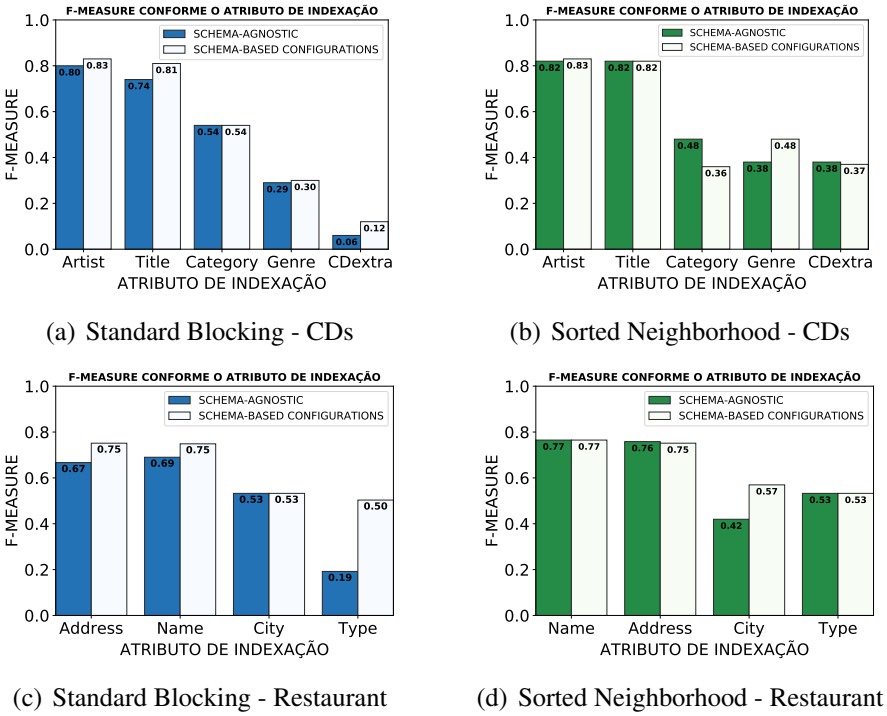
e *Suffix*, porque nos experimentos *1* e *2*, e *3* e *4*, os resultados são eficazes em termos da *F-Measure* quando comparado com os demais resultados. Por outro lado, os piores resultados da RE são alcançados quando o atributo *Given Name* é trocado pelo atributo *Address2* em qualquer combinação (experimentos *5*, *6*, *7* e *8*).

Complementando essas discussões, a Figura 3 exibe os resultados do projeto fatorial que sintetiza os experimentos da Tabela 3. O eixo-X é cada fator analisado, bem como suas iterações, e o eixo-Y a porção de efeito explicada em relação a *F-Measure* por cada fator e suas iterações. Note que o intervalo de confiança e o erro experimental são exibidos. Os resultados mostram que o atributo de indexação sozinho tem maior efeito sobre todo o processo de RE - efeito de $96,09 \pm (3,46)$, ou seja, o atributo usado na indexação explica a maior variação nos resultados em termos da *F-Measure*. Ademais, as iterações entre os fatores são insignificantes, porque o efeito explicado pelas iterações não atinge nem 1% dos resultados, isto é, as combinações de dois ou três fatores como *AB* e *ABC* não explicam os resultados da *F-Measure*, pois o atributo domina de forma isolada.

Finalmente, conclui-se que dependendo do atributo escolhido para indexar, a *F-Measure* pode aumentar ou diminuir prejudicando a eficácia da RE, pois utilizar o melhor ou o pior atributo de indexação altera significativamente a *F-Measure*. Além disso, nota-se que a eficácia da RE está mais relacionada com a escolha do atributo de indexação que é combinado com as outras funções, do que com as funções propriamente ditas, pois os resultados variam significativamente quando o atributo é alterado em ambas as técnicas.

5.2. Análise do Atributo nos métodos *Schema-Agnostic* e *Configurations-Based*

Na indexação, uma chave de bloco é atribuída a cada registro. Uma das formas é utilizar o valor dos atributos (*Schema-Agnostic*), e outra é aplicar uma regra de codificação (*Configurations-Based*). Nesse contexto, esta seção analisa o impacto da escolha do atributo de indexação nos dois métodos. Os experimentos são executados nos algoritmos *Standard Blocking* e *Sorted Neighborhood*. Especificamente, o objetivo é responder a questão **Q2**: O mesmo atributo de indexação é eficaz nos métodos *Schema-Agnostic* e *Configurations-Based*? Para responder **Q2**, experimentos em dados sintéticos e reais variando o atributo e o método de indexação são executados. A Figura 4(a)-(d) apresenta os resultados nos dados reais, onde o eixo-X exibe os atributos de indexação e o eixo-Y o valor da *F-Measure* em cada configuração de indexação para cada atributo.



(a) Standard Blocking - CDs

(b) Sorted Neighborhood - CDs

(c) Standard Blocking - Restaurant

(d) Sorted Neighborhood - Restaurant

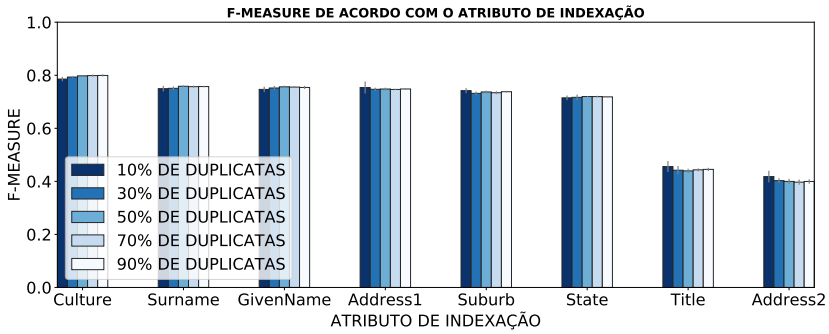
Figura 4. Atributo de indexação nos métodos *Schema-Agnostic* e *Based Configurations*.

Os resultados mostram que os atributos mais eficazes o são em ambas as técnicas de indexação. Analisando a Figura 4(a), por exemplo, o atributo *Artist* é o mais eficaz em ambas as técnicas. Ademais, comparando os algoritmos de agrupamento, os atributos mais eficazes mantêm a eficácia em ambos algoritmos com os dois métodos de indexação. Por exemplo, o atributo *Name* é eficaz em *Standard Blocking* e em *Sorted Neighborhood*, independente da indexação, conforme Figura 4(c)-(d). Assim, a eficácia da RE está mais relacionada ao atributo do que à função de indexação, pois em ambas as funções os resultados são semelhantes. Porém, o valor da *F-Measure* difere quando o atributo é alterado.

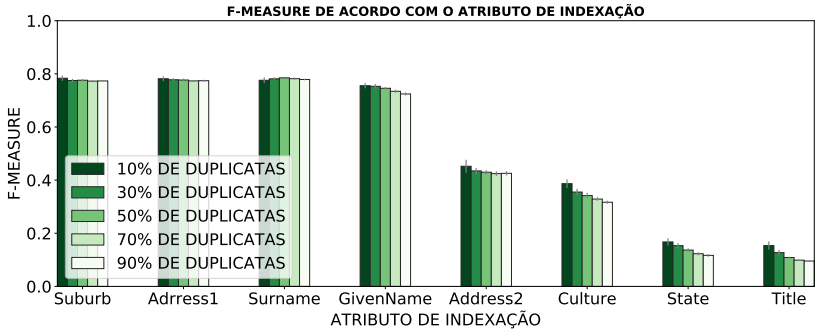
5.3. Avaliação do Atributo nos Algoritmos *Standard Blocking* e *Sorted Neighborhood*

Na RE, os registros podem ser agrupados de acordo com o vizinho mais próximo ou pelo BKV. Nesta seção, a questão **Q3** é abordada: Os resultados da RE são eficazes utilizando o mesmo atributo de indexação nos algoritmos *Standard Blocking* e *Sorted Neighborhood*? Outro objetivo é analisar se os atributos eficazes mantêm a eficácia em conjunto de dados com muitas duplicatas (e.g., 90%). Para tal, dados sintéticos com distintos cenários de duplicidade são utilizados (10% - 90%). A Figura 5(a)-(b) apresenta os resultados da *F-Measure* para cada atributo de indexação nos dois algoritmos, agrupados por atributo (eixo-X). O eixo-Y exibe a *F-Measure*, e as cores das barras representam o conjunto de dados utilizado em cada experimento. Note que o intervalo de confiança é exibido.

Os resultados demostram que: (i) atributos eficazes mantêm a eficácia em ambos algoritmos independente da quantidade de duplicatas, porque os valores da *F-Measure* são similares em todos os *datasets* (e.g., atributo *Culture* em *Standard Blocking* e atributo *Suburb* em *Sorted Neighborhood*); e (ii) atributos ineficazes o são em todos os conjuntos de dados, porque os baixos valores da *F-Measure* são mantidos quando o cenário de dados



(a) F-Measure dos atributos de indexação no algoritmo Standard Blocking



(b) F-Measure dos atributos no algoritmo Sorted Neighborhood

Figura 5. Avaliação dos atributos de indexação nos conjuntos de dados sintéticos.

duplicados muda (e.g., atributo *Address2* em *Standard Blocking* e *Title* em *Sorted Neighborhood*). Além disso, comparando os atributos de indexação entre os algoritmos, os resultados demonstram que alguns atributos de indexação são eficazes em um algoritmo e no outro não. Por exemplo, o atributo *Culture* é eficaz no algoritmo *Standard Blocking* e ineficaz no algoritmo *Sorted Neighborhood*. O atributo *State* é eficaz na abordagem de blocos e ineficaz no método de vizinhança. No entanto, há atributos que possuem resultados significantes em termos da *F-Measure* em ambos algoritmos (e.g., *Given Name*, *Surname* e *Address1*). Por fim, percebe-se que a escolha do atributo depende do método de agrupamento pois um atributo pode ser eficaz em um método e ineficaz em outro.

5.4. Análise da Combinação de Atributos na Etapa de Comparação

Na comparação dos registros, um subconjunto de atributos é selecionado com o intuito de remover aqueles atributos que não contribuem para o processo. Assim, esta seção aborda a questão **Q4**: O conjunto de atributos utilizados na comparação afeta o resultado da RE?

Nesse sentido, a Figura 6 exibe os resultados em termos da *F-meausre* para cada combinação de atributos nos conjuntos de dados *Synthetic*, *Cora* e *Restaurant*. Pode-se pensar que geralmente apenas um atributo não é suficiente para comparar um par de registros. Contudo, os resultados mostram que em alguns domínios de dados um único atributo consegue distinguir as instâncias e classificá-las corretamente. Por exemplo, no conjunto de dados *Cora*, apenas o atributo *Title* tem a maior *F-Measure*. Em *Restaurant*, a segunda maior *F-Measure* é só do *Name*. Mas, existem domínios em que somente um atributo não separa os registros de forma eficaz, como ocorre no conjunto de dados *Synthetic*, onde a melhor *F-Measure* é alcançado com um grupo de quatro atributos.

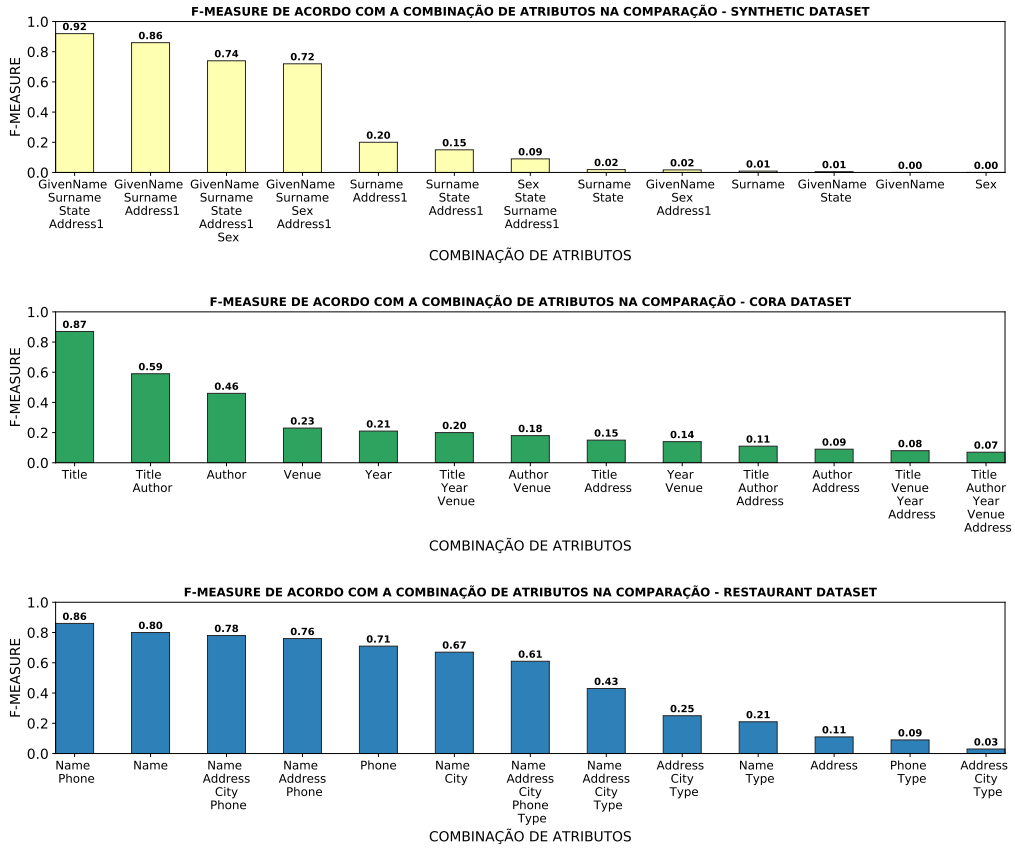


Figura 6. RE considerando diferentes combinações de atributos na comparação.

Outra discussão consiste em pensar que o resultado da RE melhora com uma maior quantidade de atributos na comparação. Porém, geralmente isso não ocorre. Por exemplo, no *Cora*, os atributos *Title*, *Author*, *Year*, *Venue* e *Address* obtêm apenas $0,07$ da *F-Measure*. No *dataset Restaurant*, *Name*, *Address*, *City*, *Phone* alcançam uma *F-Measure* de $0,78$. Quando o atributo *Type* é adicionado ao grupo, o resultado é $0,61$, ou seja, um decréscimo de $0,17$. Aqui, tem-se um exemplo de que a *F-Measure* diminui quando um atributo que não distingue os registros de forma eficaz é utilizado na comparação, pois todas as combinações com o *Type* decresceram. O mesmo ocorre com o *Sex* no *Synthetic*.

No geral, os atributos utilizados na comparação afetam o resultado da RE, porque diferentes combinações produzem diferentes valores da *F-Measure*. Considerando os três *datasets*, a diferença média da *F-Measure* entre o grupo mais eficaz e o grupo menos eficaz é quase $0,85$, ou seja, o impacto da seleção de atributos é significante. Ademais, a quantidade de atributos selecionados está totalmente relacionada com o domínio dos dados, pois o maior valor da *F-Measure* de cada *dataset* é alcançado com grupos de quantidades diferentes. Por exemplo, no *dataset Synthetic*, os atributos *GivenName*, *Surname*, *State* e *Address1* alcançam uma *F-measure* de $0,92$. No *Cora*, apenas *Title* obtém $0,87$.

6. Conclusões e Trabalhos Futuros

Este trabalho apresentou uma análise do impacto da seleção de atributos no processo de RE. Os experimentos avaliaram as etapas da RE e realizaram-se em dados reais e sintéticos. Ao final, o projeto fatorial indica que o atributo utilizado na indexação explica

uma maior variação nos resultados, ou seja, o atributo adotado tem maior influência na RE quando comparado com os métodos de indexação e/ou algoritmos de agrupamento. Na indexação, os métodos *Agnostic* e *Configurations-Based* não possuem diferença significativa, isto é, o atributo tem maior relação com a variação nos resultados do que os métodos de indexação. No agrupamento, alguns atributos são eficazes no algoritmo *Standard Blocking* e ineficazes no *Sorted Neighborhood*. Logo, o critério de seleção de atributos pode mudar conforme algoritmo escolhido. Finalmente, na comparação, o grupo de atributos selecionados afeta a RE, pois diferentes combinações produzem distintos valores da *F-Measure*. Além disso, a quantidade de atributos ideal está relacionada ao domínio dos dados. Em geral, os experimentos destacam a importância do desenvolvimento de novas estratégias que selecionem atributos relevantes para cada uma das etapas da RE. Tais estratégias podem tornar a RE mais eficaz e menos custosa, uma vez que a seleção de atributo é feita automaticamente. No futuro, pretendemos investigar quais métricas estão relacionadas aos melhores atributos do processo de RE em cada uma das etapas.

Referências

Barbosa, L. et al. (2018). Big data integration for product specifications. *Technical Committee on Data Engineering*, 41(2):71–81.

Baxter, R. et al. (2003). A comparison of fast blocking methods for record linkage. In *ACM SIGKDD*, volume 3, pages 25–27, Washington, USA.

Caldeira, L. S. and Ferreira, A. A. (2018). Melhorias no processo de blocagem para resolução de entidades baseadas na relevância dos termos. In *SBBD*, pages 61–72, Rio de Janeiro, Brasil.

Canalle, G. K. et al. (2017). A strategy for selecting relevant attributes for entity resolution in data integration systems. In *ICEIS*, pages 80–88, Porto, Portugal.

Christen, P. (2006). A comparison of personal name matching: Techniques and practical issues. In *ICDM*, pages 290–294, Hong Kong, China.

Christen, P. (2012). A survey of indexing techniques for scalable record linkage and deduplication. *TKDE*, 24(9):1537–1555.

Cohen, W. W. et al. (2003). A comparison of string distance metrics for name-matching tasks. In *WIIW*, pages 73–78, Acapulco, México.

Draisbach, U. and Naumann, F. (2009). A comparison and generalization of blocking and windowing algorithms for duplicate detection. In *QDB*, pages 51–56, Lyon, France.

Jain, R. (1992). *The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling*. Wiley.

Konda, P. et al. (2019). Executing entity matching end to end: A case study. In *EDBT*, pages 489–500, Lisbon, Portugal.

Papadakis, G. et al. (2015). Schema-agnostic vs schema-based configurations for blocking methods on homogeneous data. *PVLDB*, 9(4):312–323.

Silva, L. S. et al. (2017). Uma avaliação de eficiência e eficácia da combinação de técnicas para deduplicação de dados. In *SBBD*, pages 160–171, Uberlândia, Brasil.

Silva, L. S. et al. (2018). Automatic identification of best attributes for indexing in data deduplication. In *AMW*, Cali, Colombia.