Avaliando Fatores de Influência sobre Algoritmos de Aprendizado de Máquina na Etapa de Classificação da Resolução de Entidades

Milena Macedo Santos[1], **Dimas Cassimiro Nascimento**[1,2]

[1]Universidade Federal do Agreste de Pernambuco
[2]Universidade Federal de Campina Grande

milenasantosmcd@gmail.com,dimas.cassimiro@ufape.edu.br

Abstract. *Entity resolution is a process that seeks to identify pairs of records in databases that correspond to the same real world entity. In this work, we evaluate several classification algorithms based on Machine Learning (ML) in the context of entity resolution. We consider the following algorithms:* Adaboost, MLP, SVM, Random Forest *and* XGboost*. In the process of evaluating the ML algorithms, we analyze the impact of balanced and unbalanced training sets over the efficacy of the algorithms in the classification stage. Based on the obtained experimental results, the* Random Forest *algorithm has produced a more promising result considering the evaluated datasets. In addition, the* XGboost *model has also presented competitive results.*

Resumo. *A resolução de entidades é um processo que busca identificar pares de registros em bases de dados que correspondem à mesma entidade no mundo real. Neste trabalho, são avaliados diversos algoritmos de classificação baseados em Aprendizagem de Máquina (AM) no contexto de resolução de entidades. Os seguintes algoritmos foram explorados:* Adaboost, MLP, SVM, Random Forest *e* XGboost*. No processo de avaliação dos algoritmos de AM, são analisados o impacto do balanceamento e desbalanceamento das classes no conjunto de treinamento sobre a eficácia dos algoritmos. Com base nos resultados experimentais obtidos, o algoritmo* Random Forest *obteve resultado mais promissor, além do modelo* XGboost *ter apresentado resultados também competitivos.*

1. Introdução

Com o aumento do avanço tecnológico nas últimas décadas, a geração de informações e a quantidade de dados vêm crescendo exponencialmente. Nesse contexto, surge a necessidade da integração desses dados, seja para empresas privadas ou governamentais, com a finalidade de identificação de fraudes, comparação de dados no censo demográfico, identificação de produtos duplicados em comércio eletrônico, dentre outros. Para este objetivo, utiliza-se frequentemente a tarefa denominada Resolução de Entidades (RE), a qual visa identificar registros que representam a mesma entidade do mundo real em uma ou mais bases de dados, especialmente no contexto em que estas bases não possuem identificadores únicos [Christen 2012]. O processo de RE é crucial para diversos domínios, tais como segurança, integração de dados de saúde e consolidação de dados de citações bibliográficos [de Souza Silva et al. 2017].

A utilização de algoritmos de classificação baseados em Aprendizagem de Máquina (AM) para Resolução de Entidades é um processo que busca melhorar a correta classificação dos pares de registros, por meio da utilização de modelos inteligentes que fazem uso de um conjunto de treinamento englobando exemplos de pares de registros duplicados e não duplicados. No estado da arte, pode-se destacar alguns estudos na utilização de modelos de AM, por exemplo, o trabalho de [Ilangovan 2019], o qual buscou estudar o desempenho dos modelos *SVM* e *Random Forest*, a partir da utilização de uma base de dados com heterogeneidade. Outro trabalho [Ramezani Foukolayi 2021] comparou o uso dos algoritmos *Random Forest*, *Linear SVM*, *Radial SVM*, e *Dense Neural Networks*. No entanto, os trabalhos existentes não focam extensivamente em investigar a influência de características específicas das bases de dados e dos conjuntos de treinamento sobre a aficácia dos algoritmos de AM para RE.

Nesse contexto, a proposta desta pesquisa é avaliar a eficácia de cinco algoritmos de aprendizagem de AM para RE, englobando diversos modelos já utilizados na literatura como *Random Forest* e *SVM* ([Kaur et al. 2020, Ilangovan 2019]), além de explorar outros algoritmos como *MLP, XGboost* e *AdaBoost*.

Ademais, objetiva-se avaliar a influência do balanceamento de classes no conjunto de treinamneto e da dispersão dos níveis de similaridade entre os pares de registros sobre a eficácia dos algoritmos de AM no contexto de ER. O artigo apresenta três contribuições principais. Primeiro, foram considerados na avaliação algoritmos baseados em ensemble de classificadores pouco explorados no estado da arte: XGBoost e AdaBoost. Segundo, é investigada a influência do balanceamento da base de treinamento (em relação ao número de pares de registros duplicados e não duplicados) sobre a eficácia dos algoritmos de AM no contexto de ER. Terceiro, é proposta uma estratégia para analisar a complexidade das bases de dados avaliadas no contexto de ER com base no gráfico de dispersão; e discutido de que maneira essa análise gráfica se relaciona com a eficácia dos algoritmos de ER.

Sendo assim, o artigo investiga três hipóteses principais: i) A utilização de algoritmos baseados em ensemble de classificadores (AdaBoost e XGBoost) apresenta melhorias em relação a algoritmos clássicos de ML (SVM, Random Frests e MLP) no contexto de ER? ii) A geração de bases de treinamento balanceadas influencia no treinamento e eficácia de classificação produzida por algoritmos de AM no contexto de ER? iii) É possível correlacionar a complexidade das bases de dados no contexto de ER com a eficácia de classificadores baseados em AM utilizando gráficos de dispersão de similaridades entre os pares de registros?

O restante deste artigo está organizado da seguinte maneira. Na Seção 2, são apresentados os trabalhos relacionados. Na Seção 3, é apresentada a abordagem proposta, englobando formalização, pré-processamento dos dados e o processo de seleção dos pares de registros. Na Seção 4, é apresentada a avaliação experimental conduzida, incluindo resultados experimentais e discussão relacionada. Por fim, na Seção 5 são apresentadas as principais conclusões do trabalho, assim como perspectivas de trabalhos futuros.

2. Trabalhos Relacionados

A utilização de AM para RE visa diminuir a revisão feita por um ser humano para verificação de pares de registros que correspondem à mesma entidade do mundo real. Em [Pita et al. 2017], os autores objetivam não utilizar a revisão manual por ser uma tarefa

custosa e inviável a depender da quantidade de dados processada. A proposta parte da utilização de método probabilístico no pré-processamento para encontrar os dados correspondentes e não correspondentes. Em seguida, são empregados os modelos de aprendizagem de máquina *Decision Trees, Naive Bayes, Logistic Regression, Random Forest, Linear Support Vector Machines (SVM)* e *Gradient Boosted Trees*. Os autores de [Comber and Arribas-Bel 2019] visam realizar o processo de RE utilizando bases de dados de endereços. Para tal, são utilizados os algoritmos *XGboost, Random Forest* e *Logistic regression*. Os autores argumentam que, ao utilizar os algoritmos *ensemble*, o modelo tende a gerar um melhor resultado pelo fato dos dados presentes na etapa de classificação não permitirem a separação em um hiperplano com alta precisão, por serem não lineares

Este tema é investigado no trabalho de [Ramezani Foukolayi 2021], no qual são avaliados modelos de aprendizagem de máquina viabilizando a redução de revisão manual. Os algoritmos utilizados pelos autores foram: *Random Forest, Linear SVM, Radial SVM* e *Dense Neural Networks*. Além disso, é utilizada uma metodologia de transferência de modelos já treinados para outras bases de dados. Outra análise realizada pelos autores foi a aplicação da função de similaridade *Name2Vec* sobre os atributos nome e sobrenome presentes nas bases de dados. Outro trabalho que apresenta o mesmo objetivo de redução de revisão manual é apresentado em [Ilangovan 2019], o qual busca analisar o desempenho dos algoritmos *Random Forest, SVMs* e *Neural Nets*, englobando a inserção de erros nas bases de dados, visando a geração de heterogeneidade nas bases de dados. Por sua vez, na abordagem proposta por [Kim and Giles 2016], é empregado o algoritmo *Random Forest* para resolução de entidades no contexto de instituições financeiras, com objetivo de encontrar correspondências assertivas com uma alta precisão.

Em [Li et al. 2020], os autores propuseram uma abordagem para aplicar modelos de linguagem pré-treinados para RE. Primeiramente, os dados são pré-processados para remover ruídos, normalizar variações e extrair características relevantes das entidades. Em seguida, um modelo de linguagem pré-treinado é usado para aprender representações latentes dos dados. Essas representações são então alimentadas em um algoritmo de RE. Os resultados exprimentais obtidos demonstram que o uso de modelos de linguagem pré-treinados melhora significativamente a precisão e a eficácia do processo de RE em comparação com abordagens tradicionais. Por sua vez, os autores de [Mudgal et al. 2018] exploram o uso de técnicas de aprendizado profundo no contexto de ER. O aprendizado profundo, com suas capacidades de representação de dados complexos, surge como uma abordagem promissora para lidar com o processo de RE. Em relação à representação de entidades, os autores discutem a importância de capturar informações relevantes, como atributos estruturados, informações contextuais e representações de texto. São exploradas diversas arquiteturas de modelos de aprendizado profundo, como redes neurais convolucionais (CNNs), redes neurais recorrentes (RNNs) e redes neurais siamesas.

Este trabalho visa a avaliação de alguns algoritmos já explorados resolução de entidades, como foi destacado nos trabalhos descritos nesta seção, mas nosso diferencial consiste em realizar a comparação dos algoritmos *SVM* e *Random Forest* com um algoritmo baseado em rede neural (o *MLP*), além de considerar os algoritmos de *boosting*, como *XGboost* e *Adaboost*. Além disso, este trabalho visa investigar a influência do balanceamento de classes do conjunto de treinamento sobre a eficácia dos algoritmos

considerados.

3. Metodologia

Nesta seção, é apresentada a formalização do problema, assim como descritos os passos empregados para o cálculo de similaridade entre pares de registros e o processo de geração dos conjuntos de treinamento.

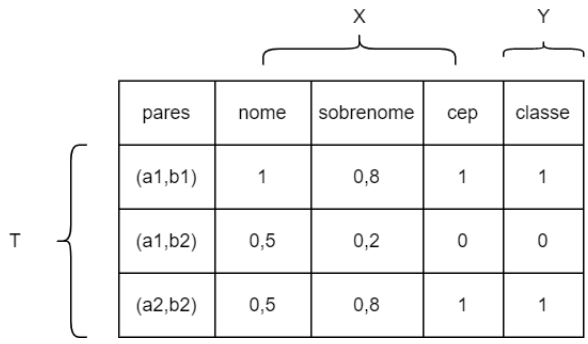
3.1. Formalização

Dados dois conjuntos de registros $A = \{a_1, a_2, \cdots, a_n\}$ e $B = \{b_1, b_2, \cdots, b_n\}$, objetiva-se identificar todos os pares de registros $(a, b) \in (A \times B)$, tal que a e b representam a mesma entidade no mundo real. Neste trabalho, é assumido que os conjuntos de dados A e B foram submetidos previamente a um processo de alinhamento de esquemas e, consequentemente, possuem a mesma quantidade de atributos.

O uso de AM para Resolução de Entidades é realizado por meio modelos matemáticos que são representados pela função $f : X \rightarrow Y$, sendo que X é o domínio dos valores de entrada e Y é a saída que são as classes de classificação (duplicado ou não duplicado).

O conjunto de treinamento T é definido como um subconjunto de $(A \times B \times S^m \times L)$, tal que $m \leq n$, $S \in [0, 1]$ representa o nível de similaridade do par de registros em relação a um dos atributos do esquema e $L = \{0, 1\}$ é o conjunto das possíveis classificações dos pares de registros: 0 (não duplicado) ou 1 (duplicado). Uma vez que são aplicadas funções de similaridade sobre valores de atributos de cada par de registros que compõe a base de treinamento, produz-se um resultado final de similaridade normalizado (i.e., entre $[0, 1]$) associado a cada par, conforme ilustrado na Figura 1.

A utilização de um classificador baseado em AM no processo de RE funciona como uma função do tipo: $f : (A \times B) \rightarrow L$, a qual tem como objetivo classificar um par de registros que não compõe a base de treinamento em uma das classificações possíveis do conjunto L: 0 ou 1.



pares	nome	sobrenome	cep	classe
(a1,b1)	1	0,8	1	1
(a1,b2)	0,5	0,2	0	0
(a2,b2)	0,5	0,8	1	1

Figura 1. Exemplo do conjunto de treinamento

3.2. Geração de Pares

Para a avaliação dos algoritmos de AM para ER, foram selecionadas bases de dados que disponham de gabarito (i.e., a indicação explícita de quais pares de registros contido na(s) base(s) de dado(s) são duplicados).

No processo de seleção de pares a serem empregados para compor os conjuntos de treinamento e teste, são utilizados três conjuntos de dados: sendo os conjuntos A e B os IDs dos registros a serem comparados e o conjunto C o conjunto de gabarito que aponta quais pares de registros são duplicados, com os IDs referentes aos registros dos conjuntos A e B. O processo de seleção de pares é realizado de forma distinta para os pares de registros duplicados e não duplicados, conforme mostrado na Figura 2.

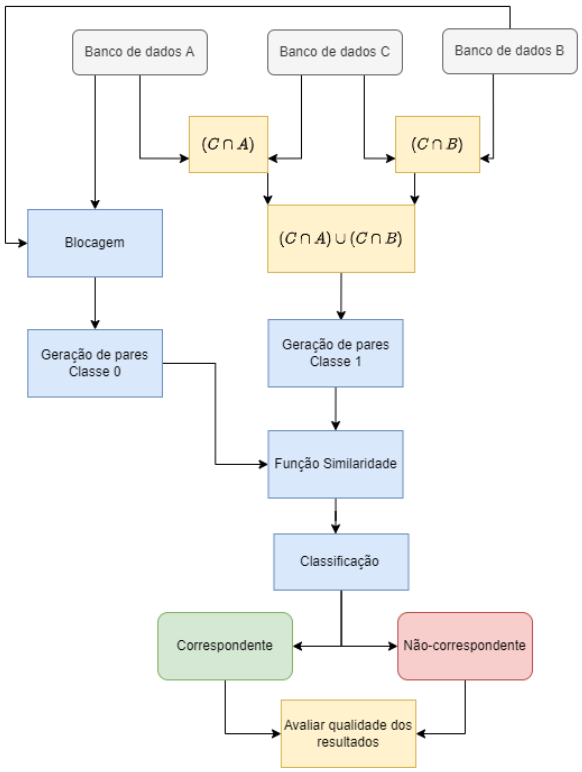


Figura 2. Fluxograma para o processo de Seleção dos Pares de Registros

O fluxograma mostrado na Figura 2 utiliza o conjunto C para garantir que todos os pares de registros duplicados (i.e., classe $= 1$) estejam presentes no conjunto de treinamento ou teste. Isto porque, o processo de RE lida com um cenário classicamente desbalanceado, ou seja, a quantidade de pares duplicados nas bases de dados é muito menor do que a quantidade de pares não duplicados. Por esta razão, todos os pares de registros duplicados (i.e., presentes do conjunto C) são selecionados para o processo de avaliação. Este processo é exemplificado na Figura 3.

Por sua vez, a seleção dos pares de registros não duplicados provenientes do conjunto $(A \times B)$ é realizado por meio de uma etapa de blocagem. Para a seleção dos pares, a blocagem é empregada para garantir que o conjunto de treinamento seja composto tanto por pares de registros pouco similares quanto por pares de registros muito similares, evitando assim, que o treinamendo dos modelos de AM seja afetado negativamente. Para tal, são escolhidas uma (ou mais) chave(s) de blocagem para geração dos blocos de dados. Após a realização do processo de blocagem, é necessária a remoção dos pares de registros que são correspondentes; esta etapa é feita a partir da comparação dos IDs dos pares de registros selecionados pela blocagem com os IDs presentes no conjunto C, o qual armazena todos os pares de registros duplicados.

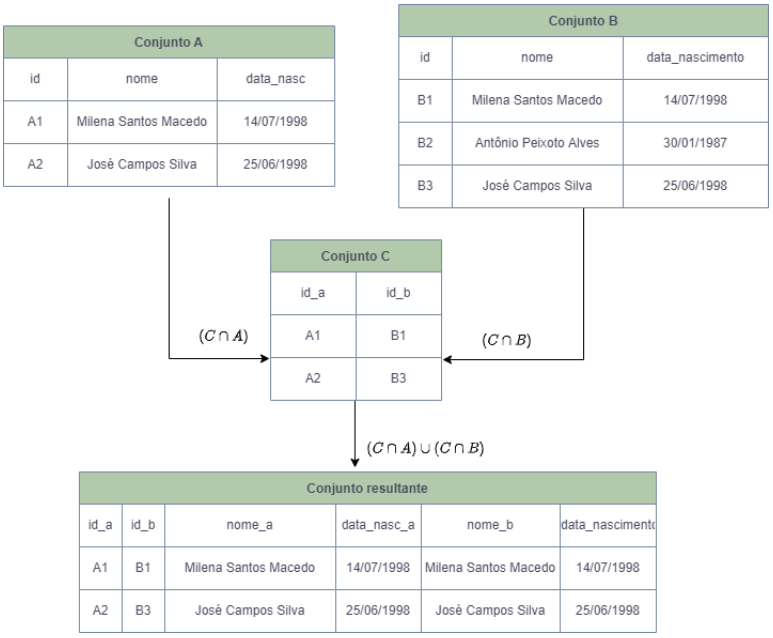


Figura 3. Exemplo de seleção de pares de registros duplicados para compor os conjuntos de treinamento e teste

4. Avaliação Experimental

Neste trabalho, o objetivo da avaliação experimental consiste em investigar a eficácia de diferentes algoritmo de AM na etapa de classificação de ER. Ademais, objetiva-se analisar a influência da composição do conjunto de treinamento (balanceado ou desbalanceado) assim como dos níveis de similaridade dos pares de registros a serem classificados sobre a eficácia dos algoritmos de classificação.

4.1. Base de dados

Foram empregados quatro pares bases de dados no contexto de RE que dispunham de gabarito ([Köpcke et al. 2010]), sendo duas bases de dados bibliográficas e duas de comércio eletrônico. Na Tablela 1, são apresentadas informações relacionadas às bases de dados utilizadas. As bases de dados utilizadas são individualmente deduplicadas, i.e., representam um cenário de avaliação denominado *Clean-Clean* [Papadakis et al. 2013].

Informações			Tamanho das bases de dados		
Contexto dos dados	Atributos	Fonte	Conjunto A	Conjunto B	Conjunto C
Bibliográficos	- título- autores- local- ano	DBLP-ACM	2,294	2,616	2,224
		DBLP-Scholar	2616	64,263	5,347
Comércio eletrônico	- nome- fabricante- descrição- preço	Amazon-GoogleProducts	1,363	3,226	1,300
		Abt-buy	1,081	1,092	1,097

Tabela 1. Informações gerais das bases de dados

4.2. Pré-Processamento

Na etapa de pré-processamento, foram realizadas tarefas de limpeza dos dados contidos nas bases de dados avaliadas. Nesta etapa, foram executadas as seguintes tarefas: i) remoção de caracteres especiais; e ii) formatação de todas as palavras para letras em

caixa baixa. O objetivo principal dessa etapa foi reduzir os ruídos e melhorar a qualidade dos dados para o processo de RE.

4.3. Métricas

Para avaliação dos modelos de AM, foi utilizada a medida F_1, a qual incorpora a média harmônica entre as métricas *precision* e *recall*. As métricas *precision* e *recall* são calculadas seguinte forma:

$$Precision = \frac{VP}{VP + FP} \tag{1}$$

$$Recall = \frac{VP}{VP + FN} \tag{2}$$

$$F_1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{3}$$

, tal que VP é a quantidade de pares Verdadeiro Positivos, VN é a quantidade de pares Verdadeiro Negativos, FP é a quantidade de pares Falso Positivos e FN é a quantidade de pares Falso Negativos.

4.4. Design Experimental

Foram projetados dois experimentos para investigar as seguintes questões experimentais: **1**) Qual é a eficácia produzida pelos cinco algoritmos de aprendizagem de máquina investigados no contexto de resolução de entidades, considerando diferentes níveis de balanceamento de classes no conjunto de treinamento? e **2**) Com base na análise de gráficos de dispersão, qual é a influência da dispersão dos níveis de similaridade entre os pares de registros sobre a eficácia dos algoritmos de AM avaliados?

No primeiro experimento (ver Tabela 2), para os pares de bases bibliográficas *DBLP-ACM* e *DBLP-Scholar*, foram empregadas as funções de similaridade Jaro-Winkler e Damerau-Levenshtein nos atributos título, autores e local de publicação, totalizando seis colunas. No atributo ano, foi utilizada a função de Damerau-Levenshtein. Com isso, foi produzido um total de sete colunas de características. Por sua vez, nas bases de dados *Amazon-GoogleProducts*, foram utilizados três atributos: nome, descrição e preço. As funções de similaridade Jaro-Winkler e Damerau-Levenshtein foram aplicadas nos atributos nome e descrição. Por fim, a função Damerau-Levenshtein foi aplicada ao atributo preço. Dessa maneira, são geradas cinco colunas de características. Por fim, no par de bases de dados *Abt-buy* foram explorados os atributos nome e descrição. A partir da aplicação das duas funções de similaridade exploradas, foram produzidas quatro colunas de características.

Foram investigados dois formatos para a geração do conjunto de treinamento: i) desbalanceada, no qual a maioria dos pares de registros são não duplicados; e ii) balanceada, no qual são eliminados de forma aleatória pares de registros não duplicados, visando manter a proporção entre pares de registros duplicados e não duplicados, conforme mostrado na Tabela 2.

Informações base de dados			Desbalanceada			Balanceada		
Fonte	Atributos	Colunas	Classe 0	Classe 1	Total	Classe 0	Classe 1	Total
DBLP-ACM	-título -autores -local -ano	7	13641	5347	18988	5347	5347	10694
DBLP-Scholar		7	6984	2224	9208	2224	2224	4448
Amazon-GoogleProducts	-nome -descricao-fabricante-preço	5	2844	1300	4144	1300	1300	2600
Abt-buy	-nome -descricao-preço	4	1466	1097	2563	1097	1097	2194

Tabela 2. Níveis de desbalanceamento do conjunto de treinamento

5. Ambiente e Implementação

Os experimentos executados neste trabalho foram realizados no Google Colaboratory, conhecido como Colab, um serviço em nuvem disponibilizado como um produto da empresa Google. O colab é usado para escrever e executar códigos em python por meio de um navegador, com intuito de facilitar o uso de AM em análise de dados. Para o processo de indexação, foi empregada a técnica de blocagem padrão para a seleção dos pares duplicados e não duplicados (ver Figura 2). A implementação da indexação utilizou a biblioteca Python Record Linkage Toolkit, versão 0.15. A blocagem padrão foi empregada utilizando o atributo mais discriminativo de cada par de bases de dados. Por sua vez, a execução dos algoritmos de AM empregou a bilioteca em Python scikit-learn. O código fonte para execução dos experimentos foi implementado utilizando a tecnologia jupyter notebook. Detalhes da implementação e da parametrização empregada pelos algoritmos estão disponíveis na seguinte URL[1].

6. Resultados

Nesta seção, serão apresentados os resultados obtidos a partir experimentos realizados, considerando as bases de dados exploradas. Na Figura 4, são apresentados os resultados do Experimento 1. Os eixos X e Y da Figura 4 representam a porcentagem do tamanho da base de treinamento empregada (em relação aos tamanhos dos conjuntos mostrados na Tabela 2) e os resultados da métrica F_1 reportados pelos algoritmos de AM, respectivamente. Por sua vez, na Figura 5, são mostrados os resultados do Experimento 2.

6.1. Análise de Influência do Balanceamento das Classes do Conjunto de Treinamento

No Experimento 1, foi investigado o impacto da utilização de conjuntos de treinamento balanceados e não balanceados sobre a eficácia produzida pelos algoritmos de AM. É importante ser ressaltado que, para esse estudo, são utilizados todos os pares de registros duplicados (componto o conjunto de treinamento ou teste), visando produzir uma maior quantidade de exemplo desta classe.

Com base nos resultados experimentais (Figura 4), é possível notar que a utilização de conjuntos de de treinamento balanceados produziram melhorias gradativas nos resultados da métrica F_1. Para exemplificar, analisando os dados das Figuras 4(a)-(b), referente ao par de bases de dados *Abt-buy*, percebe-se um aumento de em média 5.10^{-2} em relação à métrica F_1 ao empregar um conjunto de treinamento balanceado. Por sua vez, nos resultados experimentais reportados nas Figuras 4(c)-(d), referente ao par de bases de dados *Amazon-GoogleProducts*, observa-se um aumento ainda maior (de até 10^{-1}) sobre o resultado de eficácia, quando empregado o conjunto de treinamento balanceado.

[1]https://drive.google.com/drive/folders/106l4qwAPaPRAipu5am0Hci79KauB-IzV?usp=sharing

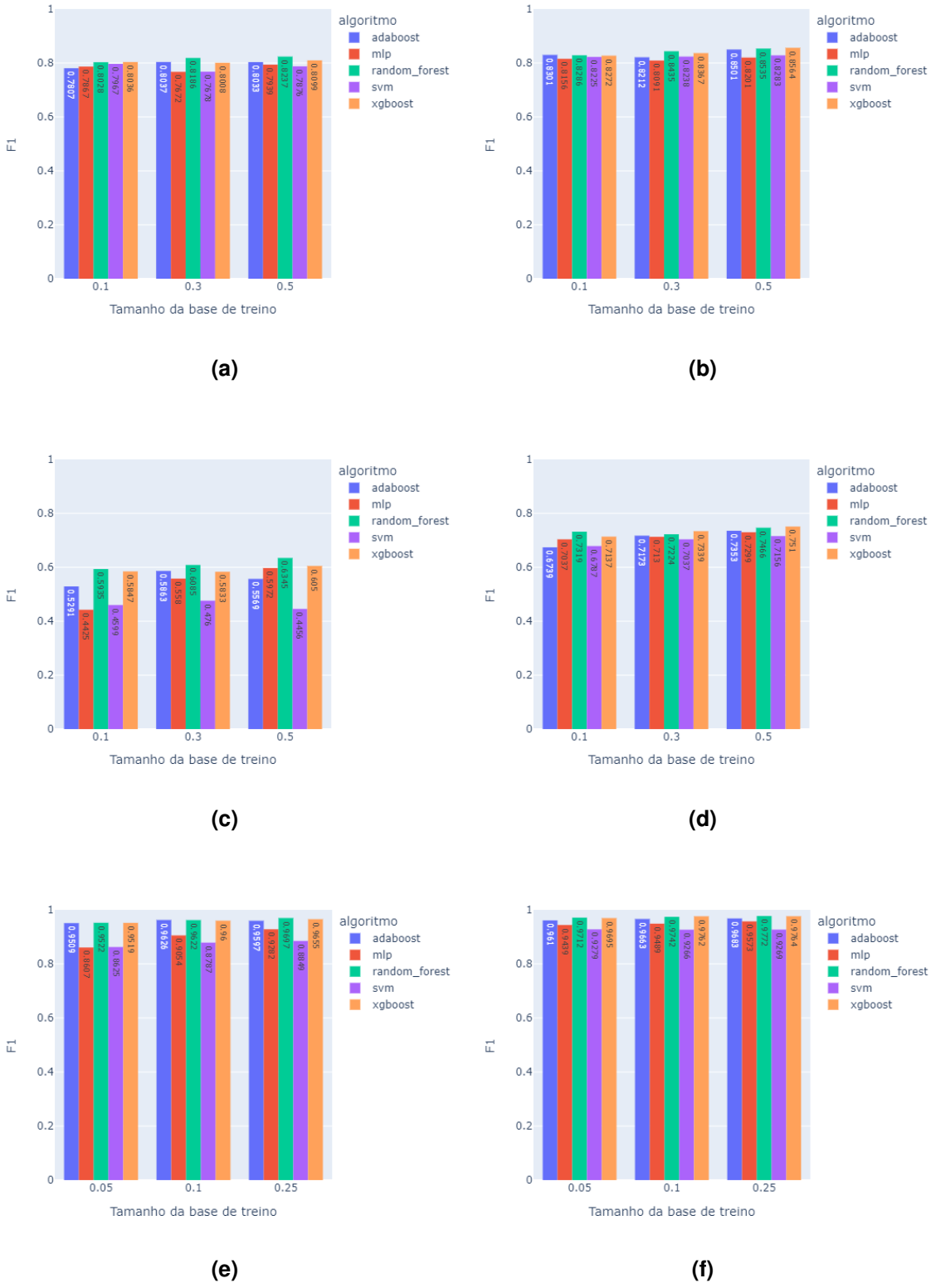


Figura 4. Resultado de Eficácia dos Algoritmos de AM, conisderando os seguintes pares de bases de dados: (a) *Abt-Buy* com conjunto de treinamento desbalanceado; (b) *abt-buy* com conjunto de treinamento balanceado; (c) *Amazon-GoogleProducts*, com conjunto de treinamento desbalanceado; (d) *Amazon-GoogleProducts*, com conjunto de treinamento balanceado; (e) *DBLB-ACM*, com conjunto de treinamento desbalanceado; (f) *DBLB-ACM*, com conjunto de treinamento balanceado.

Nos resultados obtidos, ocorre uma peculiaridade para o par de bases de dados *Amazon-GoogleProducts*, como pode ser visto nas Figuras 4(c-d), as quais reportam o menores resultados da métrica *F1*, especialmente quando comparados com os resultados produzidos pelo par de bases de dados bibliográficos (Figuras 1(e)-(f)). Este resultado é explicado pelo fato das bases de comércio eletrônico serem mais complexas, pois existem muitas formas de se referenciar um mesmo produto, seja em sua descrição ou em seu próprio nome. Contudo, as bases de dados bibliográficos se mostram mais promissoras de serem exploradas diretamente por algoritmos de Aprendizado de Máquina, atingindo um *F1* bastante significativo, uma vez que os pares de registros duplicados usualmente apresentam similaridade mais alta, o que reflete diretamente na eficácia dos algoritmos.

Com base nos resultados das Figuras 4(a)-(f), é também possível notar a influcência do tamanho do conjunto de treinamento sobre os resultados de eficácia dos algoritmos de AM. Em todos os pares de bases avaliados, o aumento no tamanho da base de treinamento acarretou em uma melhora moderada no resultado da métrica F_1. Em especial, nas bases de dados bibliográficos, os algoritmos de AM reportaram altos valores de eficácia até mesmo no cenário em que é empregado o menor conjunto de treinamento. Em cenários práticos, a geração manual de um conjunto de treinamento representa um processo lento e custoso. Neste contexto, os resultados experimentais apontam que é possível treinar um classificador eficaz baseado em AM ainda que seja empregado um conjunto de treinamento reduzido, a depender da complexidade dos pares de registros a serem classificados. Por fim, ao analisar os gráficos, é possível notar que o algoritmo *Random Forest* apresentou resultados de eficácia promissores em todas as bases de dados, obtendo um resultado superior em comparação aos demais algoritmos de AM considerados.

6.2. Análise da Dispersão dos Níveis de Similaridades entre Pares de Registros

No Experimento 2, por meio da utilização de gráficos de dispersão, pode-se obter subsídios que ajudam a explicar os diferentes níveis de eficácia produzidos pelos classificadores, a depender do par de bases de dados considerado, como pode ser observado nos resultados no Experimento 1. No primeiro exprimento, os resultados de eficácia reportados pelos algoritmos são mais altos para as bases de dados bibliográficas; e mais baixos ao processar o par de bases de dados de comércio eletrônico. Este resultado é fortemente correlacionado com a dispersão dos níveis de similaridade entre os pares de registros das bases de dados consideradas nos experiementos.

Nos resultados de dispersão dos níveis de similaridades entre os pares de registros reportados nas Figuras 5(a)-(d), é possível verificar graficamente a associação das características produzidas a partir das similaridades provenientes das distâncias Damerau-Levenshtein e Jaro-Winkler, de acordo com os atributos considerados para cada par de bases de dados. Nas Figuras 5(a)-(b), que reportam os níveis de dispersão de similaridades para bases de dados no contexto de comércio eletrônico, é possível observar uma considerável sobreposição dos pares de registros duplicados (em vermelho) e não duplicados (em azul) em regiões de similaridade próximas. Esta característica, evidenciada pelo gráfico de dispersão, torna mais complexo o processo de classificação realizado pelos algoritmos de AM. Por outro lado, ao analisar graficamente os resultados reportados nas Figuras 5(c)-(d), é possível notar que, no par de bases de dados bibliográficas, ocorre uma separação mais clara entre as regiões de ocorrência entre pares de registros duplicados e não duplicados, o que facilita o processo de classificação pelo algoritmo de AM.

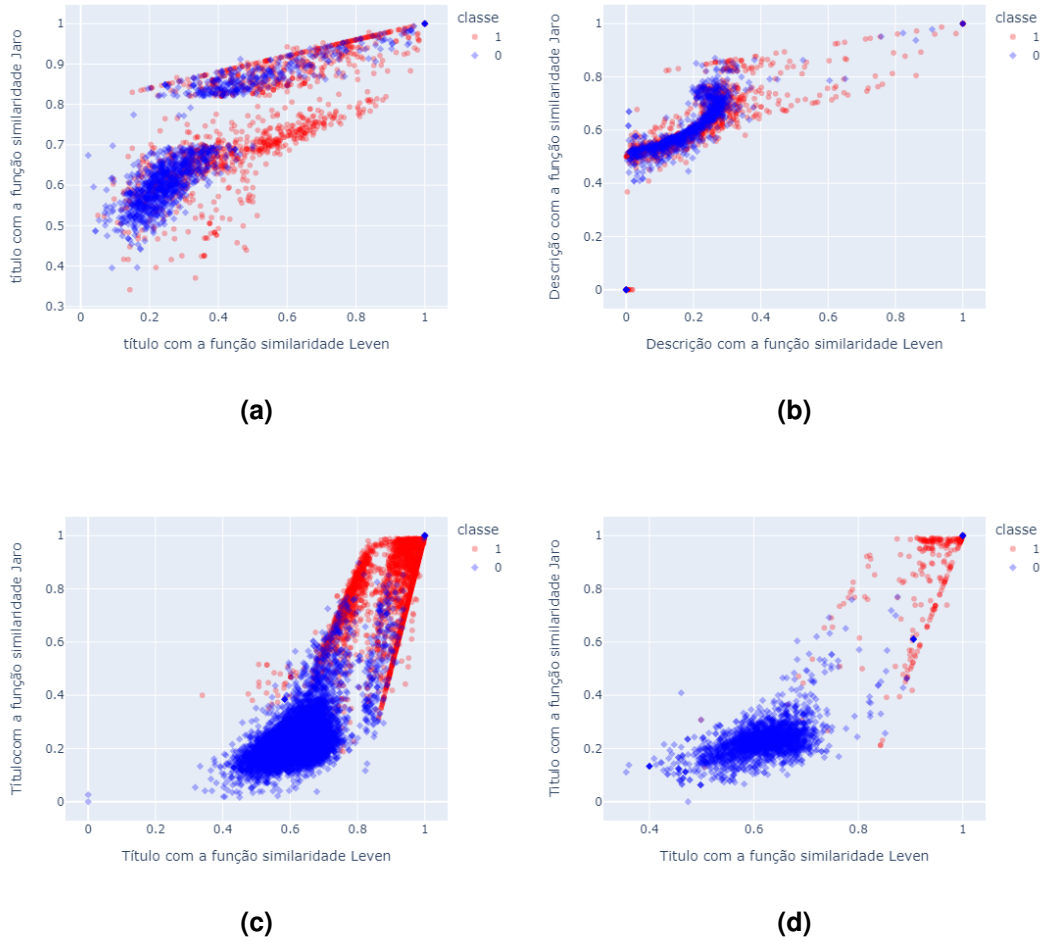


Figura 5. Gráfico de dispersão de similaridade (usando as funções *amerau-Levenshtein* e *Jaro-Winkler*) dos pares de registros das seguintes bases de dados: (a) *Amazon-GoogleProducts*, utilizando o atributo *título*; (b) *Amazon-GoogleProducts*, utilizando o atributo *descrição*; (c) *DBLB-Scholar*, utilizando o atributo *título*; (d) *DBLB-ACM*, utilizando o atributo *título*.

Um maior ou menor nível de separação entre pares de registros pertencentes a classes distintas no gráfico de dispersão pode ser considerado como um indicador do nível de dificuldade do par de bases de dados para o processo de ER. Ou seja, a existência de pares de registros duplicados que possuam entre si uma quantidade maior de erros de grafia (ou uma quantidade maior de abreviaturas ou palavras faltantes) acarreta na diminuição da similaridade entre os pares de registros duplicados (ver Figuras 5(a)-(b)), o que eleva consideravelmente a complexidade da etapa de classificação. Isto porque, a existência de pares de registros duplicados com baixa similaridade aumenta a quantidade de pares de registros fronteiriços [Peeters et al. 2023] na etapa de classificação, os quais representam pares de registros difíceis de serem corretamente classificados.

De modo análogo, a ocorrência de pares de registros não duplicados com similaridade alta (como também pode ser observado nas Figuras 5(a)-(b)), é também um fator de

complexidade associado ao processo de ER. Isto porque, tais pares de registros podem influenciar negativamente no processo de treinamento dos modelos de classificação, assim como reportado no trabalho de [Dal Bianco et al. 2018].

7. Conclusões e Trabalhos Futuros

Neste trabalho, foram avaliados cinco algoritmos de AM: *Adaboost, MLP, SVM, Random Forest* e *XGboost* na etapa de classificação para RE. Inicialmente, foi proposto um fluxo para a seleção dos pares de registros duplicados e não duplicados para comporem os conjuntos de treinamento e teste. Para avaliar os algoritmos considerados, foram projetados dois experimentos. No primeiro experimento, objetivou-se investigar a influência do nível de balanceamento do conjunto de treinamento sobre a eficácia dos modelos de AM. No segundo experimento, foi realizada uma investigação sobre os níveis de similaridade entre pares de registros, por meio de uma análise de gráficos de dispersão.

Com base nos resultados experimentais, é possível concluir que o modelo de aprendizagem *Random Forest* apresentou melhor desempenho considerando o contexto e os dados utilizados. Por sua vez, os algoritmos *SVM* e *MLP* apresentaram resultados inferiores em comparação aos demais algoritmos avaliados. Conclui-se também que a utilização de conjuntos de treinamentos balanceados tende a favorecer os resultados de eficácia reportados por algoritmos de AM no contexto de AR, chegando a produzir um aumento de mais de 10% sobre o resultado da métrica F_1. Por sua vez, com base na análise dos gráficos de dispersão, é notório que os pares de registros provenientes de bases de dados bibliográficas apresentam menos sobreposição no gráfico, o que facilita a etapa de classificação. Por sua vez, os pares de bases de dados no contexto de comércio eletrônico apresentam pares de registros bem mais desafiadores, englobando tanto pares de registros duplicados com menor similaridade quanto pares de registros não duplicados com alta similaridade, gerando uma sobreposição evidente de pares fronteiriços, o que dificulta consideravelmente a correta classificação dos pares de registros pelos algoritmos de AM. Logo, a análise de dispersão de similaridade pode ser empregada para estimar o nível de complexidade da etapa de classificação do processo de RE em um par de bases de dados. Para trabalhos futuros, pretende-se avaliar outras bases de dados considerando tamanhos e contextos distintos e gerar diversas amostras nos experimentos para viabilizar a aplicação de testes estatísticos. Além disso, pretende-se explorar técnicas mais recentes que consideram *embeddings* e *Deep learning* [Mudgal et al. 2018, Li et al. 2020].

Referências

Christen, P. (2012). *Data Matching: Concepts and Techniques for Record Linkage, Entity Resolution, and Duplicate Detection*. Springer.

Comber, S. and Arribas-Bel, D. (2019). Machine learning innovations in address matching: A practical comparison of word2vec and crfs. *Transactions in GIS*, 23(2):334–348.

Dal Bianco, G., Gonçalves, M. A., and Duarte, D. (2018). Bloss: Effective meta-blocking with almost no effort. *Information Systems*, 75:75–89.

de Souza Silva, L., Nascimento Filho, D. C., and Moro, M. M. (2017). Uma avaliação de eficiência e eficácia da combinaçao de técnicas para deduplicaçao de dados. In *Anais do XXXII Simpósio Brasileiro de Bancos de Dados*, pages 160–171. SBC.

Ilangovan, G. (2019). Benchmarking the effectiveness and efficiency of machine learning algorithms for record linkage. Master's thesis, Texas AM University.

Kaur, P. et al. (2020). A comparison of machine learning classifiers for use on historical record linkage. Master's thesis, University of Guelph.

Kim, K. and Giles, C. L. (2016). Financial entity record linkage with random forests. In *Proceedings of the second international workshop on data science for macro-modeling*, pages 1–2.

Köpcke, H., Thor, A., and Rahm, E. (2010). Evaluation of entity resolution approaches on real-world match problems. *Proceedings of the VLDB Endowment*, 3(1-2):484–493.

Li, Y., Li, J., Suhara, Y., Doan, A., and Tan, W.-C. (2020). Deep entity matching with pre-trained language models. *arXiv preprint arXiv:2004.00584*.

Mudgal, S., Li, H., Rekatsinas, T., Doan, A., Park, Y., Krishnan, G., Deep, R., Arcaute, E., and Raghavendra, V. (2018). Deep learning for entity matching: A design space exploration. In *Proceedings of the 2018 International Conference on Management of Data*, pages 19–34.

Papadakis, G., Koutrika, G., Palpanas, T., and Nejdl, W. (2013). Meta-blocking: Taking entity resolutionto the next level. *IEEE Transactions on Knowledge and Data Engineering*, 26(8):1946–1960.

Peeters, R., Der, R. C., and Bizer, C. (2023). Wdc products: A multi-dimensional entity matching benchmark. *arXiv preprint arXiv:2301.09521*.

Pita, R., Mendonça, E., Reis, S., Barreto, M., and Denaxas, S. (2017). A machine learning trainable model to assess the accuracy of probabilistic record linkage. In *Big Data Analytics and Knowledge Discovery: 19th International Conference, DaWaK 2017, Lyon, France, August 28–31, 2017, Proceedings 19*, pages 214–227. Springer.

Ramezani Foukolayi, M. (2021). Comparison of machine learning algorithms in a human-computer hybrid record linkage system. Master's thesis, Texas AM University.