Pareamento de Nomes de Produtos e Serviços Utilizando Medidas de Similaridade Textual nos Níveis Alfabético, Léxico e Semântico

Thiago Pereira Meirelles[1], Eduardo Corrêa Gonçalves[1], Daniel Takata Gomes[1]

[1]Escola Nacional de Ciências Estatísticas (ENCE/IBGE)
Rio de Janeiro – RJ – Brasil

thiagopmeirelles@gmail.com, eduardo.correa@ibge.gov.br,
daniel.gomes@ibge.gov.br

Abstract. *Text matching is the task of choosing, among a set of texts, which one refers to the same concept or object as a given input text. Based on textual similarity measures that address the alphabetic, lexical, and semantic levels, this work compares the performance of automated matching strategies that use these measures separately or in combination. The performance was evaluated through an experiment in which the measures were applied to perform the matching of product and service descriptions obtained from the questionnaires of two surveys conducted by the Brazilian Institute of Geography and Statistics (IBGE): Consumer Expenditure Survey (POF) and Consumer Price Index (IPC). In line with what has been found in earlier studies, a strategy that combines different similarity measures, which acts on the three aforementioned levels, performed better, obtaining a superior number of correct matches when compared to strategies that employ solely one of the measures. A further investigation of the incorrect pairings produced by the best strategies was done with the goal of categorizing the types of errors and proposing additional approaches to improve accuracy.*

Resumo. *O pareamento de textos é a tarefa de escolher, dentre um conjunto de textos possíveis, qual deles faz menção a um mesmo conceito ou objeto que outro determinado texto de entrada faz. Baseando-se em medidas de similaridade textual que atuam nos níveis alfabético, léxico e semântico, este trabalho compara a performance de estratégias automatizadas de pareamento que utilizam tais medidas de forma separada ou combinada. Essa performance foi avaliada através de experimento que consistiu no pareamento de descrições de produtos e serviços obtidos dos questionários de duas pesquisas do Instituto Brasileiro de Geografia e Estatística (IBGE): Pesquisa de Orçamentos Familiares (POF) e Índices de Preços do Consumidor (IPC). Em consonância com o observado em outros trabalhos, uma estratégia que combina medidas de similaridade diferentes, que atuam nos três níveis mencionados, obteve melhor performance, realizando um maior número de pareamentos corretos, quando comparada a estratégias que empregam apenas uma das medidas isoladamente. Uma investigação dos pareamentos incorretos produzidos pelas melhores estratégias foi feita com os objetivos de categorizar tipos de erros e propor abordagens adicionais que melhorem a acurácia.*

Cadernos do IME - Série Informática
e-ISSN: 2317-2193 (online)
DOI: 10.12957/cadinf.2021.68557

1. Introdução

O pareamento de informações consiste no problema de decisão de escolher, dentre um conjunto de informações candidatas, aquela que mais se assemelha, em algum sentido intuitivo, a uma dada informação inicial, baseando-se em algum critério de similaridade. Ele é dito probabilístico quando não há meios, a priori, de estabelecer um processo de pareamento livre de erros [Fellegi and Sunter 1969].

Como as informações agrupam-se em diversos meios diferentes – sons, imagens, textos, entre outros – foram muitas as técnicas de pareamento probabilístico desenvolvidas, cada uma fazendo uso de características próprias de cada problema concreto. Ou seja, temos métodos de pareamento de imagens, sons, sinais ou códigos, em diversos contextos e aplicações [Dumont and Mérialdo 2010]. Dentre todos esses meios, uma das mais comuns aplicações encontra-se no pareamento de textos, em função da grande disponibilidade de registros armazenados em meio escrito e da abundância de informações que podem ser representadas como texto, que são sequências ordenadas de caracteres [Jurafsky and Martin 2020].

Em função dessa abundância, o problema do pareamento textual deu origem a diversas maneiras de responder à pergunta sobre como medir a similaridade entre duas informações escritas. O pareamento de partituras musicais, por exemplo, precisa levar em conta a invariância por transposição – quando um segmento musical tem uma de suas qualidades sonoras (como timbre ou altura) uniformemente alteradas, resultando em uma notação musical diferente, mas que mantém alto grau de similaridade sonora [Lemstrom and Perttu 2000]. Já para sequências genéticas, a possibilidade de cortar uma sequência em certos pontos e religar as partes resultantes em ordem diferente é importante fator para medir a similaridade entre genomas [Rubert 2019]. Alternativamente, nomes de pessoas ou locais aparecem muitas vezes abreviados, o que aumenta a relevância de porções iniciais da cadeia de caracteres [Winkler 1990].

Grande parte das técnicas de pareamento textual faz uso da chamada similaridade ou distância léxica, que infere a similaridade dos textos baseando-se nas suas partes constituintes – caracteres ou sequências finitas de caracteres. A distância de Levenshtein [Levenshtein 1966], por exemplo, infere a similaridade entre dois textos baseando-se na quantidade de operações de inserção, remoção e substituição de caracteres necessárias para transformar o primeiro texto no segundo. De acordo com tal medida, a distância entre os textos “assento” e “acento” é 2, pois basta remover um caractere e substituir outro. A similaridade de Jaro-Winkler [Winkler 1990], por outro lado, baseia-se na operação de transposição de caracteres. Tais técnicas têm sido usadas com sucesso para realizar o pareamento de nomes próprios e endereços [Silva et al. 2010, Davis Jr. and Salles 2009] e para a tarefa de correção automática de textos, muito comum em telefones celulares e navegadores [Lhoussain et al. 2015].

Apesar do sucesso em várias aplicações, o pareamento de texto baseado unicamente na similaridade léxica pode ser insatisfatório em certas situações. Na presença de palavras parônimas, como ilustrado anteriormente, temos uma alta similaridade léxica, o que pode ser indesejado em aplicações onde a semântica é relevante. Termos sinônimos também podem ser exemplos em que a similaridade léxica se mostra inadequada – a

distância de Levenshtein entre ‘mandioca’ e ‘aipim’ é 6, apesar de representarem o mesmo conceito. Por fim, quando os textos possuem tamanhos diferentes, tende-se a atribuir uma baixa similaridade léxica, o que também pode ser indesejado em tarefas de parear textos e resumos, por exemplo.

Dessa forma, a incorporação de uma medida de similaridade semântica entre textos pode mitigar os problemas mencionados anteriormente, melhorando a acurácia do pareamento. Uma das maneiras de capturar a semântica de palavras é representá-las como vetores densos e de baixa dimensão – *embeddings* –, construídos de acordo com a posição relativa das palavras dentro de uma biblioteca de textos. Essa é a abordagem da metodologia Word2vec [Mikolov et al. 2013a, Mikolov et al. 2013b, Word2Vec 2013], que constrói embeddings com base em um classificador logístico que responde à pergunta se duas palavras ocorrem em posições adjacentes na biblioteca de textos. Assim, a combinação dos critérios léxico e semântico, formando uma estratégia híbrida, pode produzir melhores resultados, com maior acurácia na tarefa de pareamento e menor quantidade de falsos positivos.

O objetivo deste trabalho é comparar abordagens de pareamento baseadas puramente na similaridade alfabética / léxica com uma abordagem híbrida, que incorpora uma medida de similaridade semântica baseada em embeddings gerados pelo modelo Word2vec. O restante do artigo está dividido da seguinte forma. A Seção 2 apresenta o referencial teórico e trabalhos relacionados. A proposta de uma nova estratégia híbrida para o pareamento de textos curtos é realizada na Seção 3. Na Seção 4, apresentam-se os resultados experimentais em uma base de dados com nomes de produtos e serviços. Por fim, as conclusões do estudo e ideias para trabalhos futuros são apresentadas na Seção 5.

2. Referencial Teórico e Trabalhos Relacionados

Com o intuito de uniformizar a linguagem deste trabalho, inicialmente apresenta-se uma lista de definições.

Um alfabeto será qualquer conjunto finito de símbolos, também chamados caracteres. Nesse trabalho, o alfabeto será o usual alfabeto latino, juntamente com os diacríticos do português e os sinais de pontuação, além do símbolo que denota um espaço em branco – *whitespace*.

Um texto, ou *string*, será uma sequência finita de símbolos do alfabeto. Além disso, será definido *token*, ou palavra, como uma string envolta por caracteres especiais do alfabeto que chamaremos de separadores. Tais separadores não fazem parte do referido token, servindo apenas para delimitá-lo. Usualmente são utilizados como separadores as pontuações e whitespaces, este último o utilizado neste trabalho, quando indicado.

Dadas as strings s_1 e s_2, uma função de similaridade entre estas strings é uma função S: $(s_1, s_2) \rightarrow [0;1]$ que satisfaz três propriedades [Winkler 1990]:

1. $S(s_1, s_2) = 1$ se $s_1 = s_2$;
2. $S(s_1, s_2) \approx 1$ quando s_1 é muito parecida com s_2, em algum sentido;
3. $S(s_1, s_2) \approx 0$ quando s_1 é muito diferente de s_2, em algum sentido.

Caracterizada a função de similaridade S, sua construção pode ser feita de diversas formas. Serão abordadas neste trabalho funções baseadas em: (i) distância de edição; (ii) tokens; (iii) embeddings semânticos. Os dois primeiros tipos de função avaliam

similaridade nos níveis alfabético e/ou léxico, enquanto o terceiro é focado no nível semântico.

2.1. Similaridade baseada em Distância de Edição

Medidas de similaridade baseadas em distância de edição utilizam o conceito de distância de edição mínima, definida como a quantidade mínima de operações de edição necessária para transformar s_1 em s_2. Uma das mais simples utiliza a Distância de Levenshtein, denotada $d_L(s_1, s_2)$, onde são permitidas apenas as operações de inserção, remoção e substituição de caracteres [Levenshtein, 1966]. Com isso, a Similaridade de Levenshtein, entre as strings s_1 e s_2, denotada por $S_L(s_1, s_2)$, é definida de acordo com a Equação 1. Nesta equação, $|s_1|$ e $|s_2|$ representam respectivamente, o comprimento das strings s_1 e s_2.

$$S_L(s_1, s_2) = 1 - \left(\frac{d_L(s_1, s_2)}{max\,(|s_1|, |s_2|)}\right) \quad (1)$$

A Similaridade de Jaro [Winkler 1990] baseia-se na quantidade de caracteres iguais que se encontrem em posições próximas nas strings e na operação de transposição de caracteres. Os caracteres i de s_1 e j de s_2 são ditos correspondentes se $i = j$ e suas ocorrências em cada string não estejam afastadas por mais de $(max(|s_1|, |s_2|) / 2) - 1$ posições. Denotando por c a quantidade de caracteres correspondentes e por t a quantidade de caracteres correspondentes que aparecem com ordem trocada em s_1, relativo a s_2, a Similaridade de Jaro, denotada por $S_J(s_1, s_2)$, é definida de acordo com a Equação 2:

$$S_J(s_1, s_2) = \frac{1}{3}\left(\frac{c}{|s_1|} + \frac{c}{|s_2|} + \frac{c - t}{c}\right) \quad (2)$$

2.2. Similaridade baseada em Tokens

Medidas de similaridade baseadas em tokens usam, como unidade básica de análise, os tokens (palavras) que compõem as strings. Este tipo de medida atua estritamente no nível de similaridade léxico e não no alfabético. Um exemplo é a Similaridade Jaccard, que foi originalmente proposta como uma medida de similaridade entre a flora de determinadas regiões [Jaccard 1912], mas é empregada em diversos contextos em que se deseja medir a similaridade entre dois conjuntos devido à sua generalidade [Lescovec et al., 2020].

Dadas duas strings s_1 e s_2, denota-se $tok(s_1)$ e $tok(s_2)$ como o conjunto de tokens que formam s_1 e s_2, respectivamente, de acordo com algum conjunto de caracteres separadores previamente definidos. Com isso, a Similaridade de Jaccard entre s_1 e s_2, denotada por $S_{JC}(s_1, s_2)$, é definida de acordo com a Equação 3.

$$S_{JC}(s_1, s_2) = \frac{|tok(s_1) \cap tok(s_2)|}{|tok(s_1) \cup tok(s_2)|} \quad (3)$$

2.3. Similaridade baseada em Embeddings Semânticos

Embeddings semânticos são vetores numéricos densos e de baixa dimensão – usualmente entre 50 e 1000 posições –, que tentam representar, de algum modo, o significado das palavras de um determinado idioma. Seu uso baseia-se no fato de que uma única palavra pode apresentar múltiplos sentidos e na chamada hipótese distribucional, que postula que palavras que ocorrem em contexto similar possuem significado similar [Jurafsky and Martin 2020]. Dentre os diversos métodos para construção de embeddings, utilizou-se neste trabalho a metodologia Word2vec [Mikolov et al. 2013a, Mikolov et al. 2013b] que constrói embeddings com base em um classificador logístico que responde à pergunta se duas palavras ocorrem em posições adjacentes na biblioteca de textos. A escolha da técnica Word2vec se deu pela disponibilidade de embeddings pré-construídos em grandes conjuntos de textos da língua portuguesa [NILC 2017].

Neste trabalho, a similaridade semântica entre as strings s_1 e s_2, denotada por $S_W(s_1, s_2)$, é computada através do cosseno dos embeddings associados a elas, representados, respectivamente, por $emb(s_1)$ e $emb(s_2)$ na Equação 4. No caso em que uma string é composta por múltiplas palavras, o embedding associado a tal string será simplesmente a média entre os vetores de cada palavra que a compõe.

$$S_W(s_1, s_2) = max(0, \cos(emb(s_1), emb(s_2)) \quad (4)$$

3. Método Proposto

A partir das medidas de similaridade apresentadas na seção anterior, apresenta-se a seguir a técnica utilizada neste trabalho para a construção de matrizes de similaridade para cada medida considerada. Dada uma lista s_o de N_o textos de origem e uma lista s_d de N_d textos de destino, uma matriz de similaridade M relacionada a uma função de similaridade S é uma matriz $N_o \times N_d$, definida por $M(i, j) = S(s_o[i], s_d[j])$, onde $i = 1,2,..., N_o$ e $j = 1,2,..., N_d$. Ou seja, o elemento (i, j) da matriz de similaridade M representa a similaridade entre o i-ésimo texto de origem e o j-ésimo texto de destino, de acordo com uma função de similaridade S escolhida

Desta forma, a partir de duas listas s_o e s_d, torna-se possível construir 4 matrizes de similaridade, denotadas por M_L, M_J, M_{JC} e M_W, associadas às funções de similaridade S_L (Levenshtein), S_J (Jaro), S_{JC} (Jaccard) e S_W (Word2vec), respectivamente. Isso possibilitou a definição das estratégias de pareamento avaliadas neste trabalho. Uma estratégia de pareamento é uma função que associa um texto origem e uma ou mais matrizes de similaridade a um subconjunto dos textos de destino, que será denominado de conjunto pareado, denotado s_p. Este trabalho comparou sete estratégias de pareamento, subdivididas em estratégias simples e híbridas.

3.1. Estratégias Simples

Nas estratégias simples, o i-ésimo texto origem $s_o[i]$ será pareado com o conjunto de textos que tiverem máxima similaridade δ_i, de acordo com uma determinada matriz de similaridade. Por exemplo, para a matriz M_L, tem-se $\delta_{L,i} = \max\{M_{(i,j)}: j = 1,2,..., N_d\}$ e o conjunto pareado será definido por $s_{p,i} = \{s_d[j] \in s_d : M_{(i,j)} = \delta_{L,i}\}$.

3.2. Estratégias Híbridas

Uma estratégia híbrida envolve a construção de uma matriz de similaridade que combina valores de duas ou mais matrizes diferentes. Neste trabalho, uma matriz híbrida de similaridade M_H terá elementos da forma apresentada na Equação 5:

$$M_H(i,j) = \frac{1}{n}\left(M^{\alpha}_{1\,(i,j)} + M^{\alpha}_{1\,(i,j)} + \cdots + M^{\alpha}_{n\,(i,j)}\right) \quad (5)$$

onde M_1, M_2, ..., M_n são n matrizes de similaridade escolhidas previamente e o parâmetro $\alpha \in \mathbb{R}^+$ atua como ponderação, valorando proporcionalmente mais os valores de similaridade extremos. Foram testadas três estratégias híbridas, denotadas M_{H1} a M_{H3} e assim definidas:

a) $M_{H1}(i,j) = \frac{1}{3}\left(M_{L\,(i,j)} + M_{J\,(i,j)} + M_{JC\,(i,j)}\right)$, funcionando como estratégia híbrida base;

b) $M_{H2}(i,j) = \frac{1}{4}\left(M_{L\,(i,j)} + M_{J\,(i,j)} + M_{JC\,(i,j)} + M_{W(i,j)}\right)$, incorporando a dimensão semântica;

c) $M_{H3}(i,j) = \frac{1}{4}\left(M^2_{L\,(i,j)} + M^2_{J(i,j)} + M^2_{JC\,(i,j)} + M^2_{W\,(i,j)}\right)$, aumentando o peso relativo dos valores de similaridade maiores, através de uma transformação convexa.

Construídas as matrizes híbridas, o pareamento ocorrerá de acordo com a sistemática explicada para a estratégia simples, i.e., de acordo com a máxima similaridade.

4. Experimento

A base de dados[1] utilizada neste estudo é composta por 4.956 pares de descrições de produtos e serviços [Meirelles et al. 2021]. A Tabela 1 relaciona alguns exemplos. O conjunto de textos de origem consistiu das descrições de produtos e serviços objeto de despesas de famílias residentes em regiões metropolitanas do Brasil, obtidos através do questionário da mais recente da Pesquisa de Orçamentos Familiares (POF) do IBGE, realizada entre os anos de 2017 e 2018. Já os textos de destino constituem-se as descrições de despesa monetária de consumo doméstico, que possibilitam a construção de cestas de consumo usadas na medição da evolução de preços de segmentos populacionais e criação de Índice de Preços ao Consumidor (IPC) do IBGE. A metodologia de tradução dos questionários de Descrição POF → Descrição IPC foi descrita em [IBGE, 2020].

[1] Disponível em: https://www.ibge.gov.br/estatisticas/economicas/precos-e-custos/9256-indice-nacional-de-precos-ao-consumidor-amplo.html?=&t=downloads.

Tabela 1. Exemplos de pares de descrições da base de dados

Descrição Origem	Descrição Destino
ARROZ POLIDO	Arroz
COCO BURITI	Buriti (Coco)
MAIZENA	Amido de Milho
QUEIJEIRA	Utensílios de Plástico
ACADEMIA	Atividades Físicas
TRAILLER	Trailer
AUTO ESCOLA	Autoescola
AULA DE LIBRAS	Curso de libras
DETERGENTE EM PO	Detergente
PIMENTAO	Pimentão

Para conduzir os experimentos reportados neste trabalho, as seguintes tarefas de pré-processamento foram executadas sobre a base de dados: (i) conversão das descrições para minúsculo; (ii) remoção de sinais de pontuação; (iii) remoção de stop words, como artigos, preposições etc.; (iv) exclusão de linhas com o pareamento idêntico (ou seja, onde a Descrição Origem é igual a Descrição Destino). Ao final do pré-processamento, chegou-se a uma base de dados com um total de 3.910 pares de descrições.

4.1. Procedimento para Comparação das Estratégias

Cada estratégia de pareamento foi avaliada de acordo com três diferentes métricas de performance, tentando capturar seu grau de acurácia, assim como o quão longe ela se encontra de parear corretamente um determinado texto de origem. A seguir, estas métricas são apresentadas:

- **Acurácia Estrita**: para capturar a performance de uma estratégia de maneira geral, a acurácia estrita será definida como a proporção de vezes em que a estratégia pareou única e corretamente o texto de origem.
- **Acurácia Ponderada**: relaxa a restrição de que o pareamento deva ser único, mas penaliza proporcionalmente estratégias que produzam conjuntos pareados com muitos elementos.
- **Posição Média**: por fim, de forma a capturar o quão distante uma estratégia fica de realizar o pareamento correto, computa-se a posição média do par correto. Para isso, após ordenada a lista de textos de destino para um determinado texto de origem, registra-se o rank do par correto. A posição média será, então, a média de todos os ranks registrados.

Considere a matriz de similaridade hipotética *M* apresentada na Tabela 2, com as descrições de origem e destino por linha e coluna, respectivamente. As células da matriz apresentam os valores de similaridade para cada par de descrição origem-destino. Os pareamentos corretos foram indicados pelas cores correspondentes. Com essa matriz de similaridade, ‘arroz polido’ é incorretamente pareado com ‘arroz pré-cozido’, assim como ‘maizena’ é incorretamente pareado com ‘arroz’. Por outro lado, ‘queijeira’ é corretamente pareado com ‘utensílios de plástico’ e ‘academia’ é pareado tanto com ‘atividades físicas’ quanto com ‘jogos de azar’. Assim, as medidas de desempenho são:

- Acurácia Estrita = (0 + 0 + 1 + 0) / 4 = 0,250.
- Acurácia Ponderada = (0 + 0 + 1 + 0,5) / 4 = 0,375.

- Posição Média = (2 + 6 + 1 + 1) / 4 = 2,500.

Tabela 2. Matriz de similaridade hipotética *M*

		DESTINO					
		arroz	**amido de milho**	**utensílios de plástico**	**atividades físicas**	**jogos de azar**	**arroz pré-cozido**
ORIGEM	**arroz polido**	0,88	0,59	0,49	0,43	0,46	0,92
	maizena	0,56	0,40	0,41	0,51	0,47	0,47
	queijeira	0,00	0,40	0,57	0,47	0,41	0,41
	academia	0,44	0,52	0,39	0,56	0,56	0,56

O pareamento de ‘academia’ não é computado para a acurácia estrita pois ele não é único. Por outro lado, como ele foi 0,5-correto, ele é computado na acurácia ponderada. O rank para ‘maizena’ é 7 pois seu par correto – ‘amido de milho’ – possui sétima maior similaridade com ‘maizena’ de acordo com a matriz *M*.

4.2. Resultados

Essa seção descreve e analisa os dados e resultados obtidos nos experimentos de pareamento, de acordo com a metodologia apresentada na subseção anterior. Todos os experimentos foram realizados localmente em um computador com sistema operacional Windows 10 Home, processador Intel Core(TM) i5-3337U e 6GB de memória RAM, utilizando a linguagem Python v. 3.8.6, em especial a biblioteca Gensim [2] para carregamento dos embeddings semânticos de 300 dimensões disponibilizados em (NILC, 2017) e a biblioteca strsimpy[3], para cálculo das funções de similaridade. Os embeddings semânticos são carregados em memória, ocupando cerca de 2,6 GB, e do conjunto completo são extraídos apenas os embeddings associados a palavras presentes na base de dados deste trabalho, ocupando cerca de 20 MB. As matrizes de similaridade foram calculadas e guardadas em disco, ocupando cerca de 120 MB cada, em formato csv.

Em consonância com o descrito na motivação deste trabalho, desejou-se investigar a contribuição da dimensão semântica para a tarefa de pareamento, tanto isoladamente quanto em conjunto com medidas de similaridade léxica baseadas em distância de edição e em tokens. Apresenta-se na Tabela 3 os resultados das estratégias simples. Nota-se, inicialmente, que há bastante proximidade na performance de pareamento das estratégias simples, com acurácia estrita sempre situando-se na faixa de 37% a 45%. Também, há diferença pouco apreciável entre os valores de ambas as acurácias (estrita e ponderada), com exceção para a estratégia simples que usa a similaridade de Jaccard, que produz muitos empates. A dimensão semântica, isoladamente, não consegue produzir pareamentos melhores do que as outras estratégias, ficando com uma acurácia estrita de 39%. A maior diferença encontrada está na posição média do par correto, onde o uso da semântica é capaz de aproximar a descrição correta de destino do valor de máxima similaridade.

[2] Gensim: https://radimrehurek.com/gensim/
[3] Strsimpy: https://pypi.org/project/strsimpy/

Os resultados das estratégias de pareamento híbridas são apresentados na Tabela 4. Esperava-se conseguir uma melhora de performance com o uso de estratégias híbridas, o que de fato ocorreu. A combinação de três medidas de similaridade de M_{H1} – Levenshtein, Jaro e Jaccard – aumentou a acurácia de pareamento, além de diminuir sensivelmente a posição média do par correto, quando comparadas às estratégias simples que utilizam apenas uma dessas medidas. Subsequente melhora ocorreu com a introdução da medida de similaridade semântica (M_{H2} e M_{H3}) – pequena melhora de acurácia, da ordem de 4%, e grande melhora no rank médio do par correto.

Tabela 3. Performance de Pareamento – Estratégias Simples

Matriz	Acurácia Estrita	Acurácia Ponderada	Posição Média
Levenshtein (M_L)	0,3803	0,3946	592,7
Jaro (M_J)	**0,4514**	**0,4522**	558,3
Jaccard (M_{JC})	0,3731	0,4124	692,5
Semântica (M_W)	0,3900	0,3900	**375,8**

Tabela 4. Performance de Pareamento – Estratégias Híbridas

Matriz	Acurácia Estrita	Acurácia Ponderada	Posição Média
M_{H1}	0,4884	0,4887	498,1
M_{H2}	0,5281	0,5281	**312,2**
M_{H3}	**0,5294**	**0,5294**	339,5

Uma vez que obteve-se cerca de 53% de acertos na tarefa de pareamento – o que corresponde a obter rank 1 para o par correto –, combinado ao resultado de que o rank médio do par correto produzido pelas melhores estratégias foi cerca de 300, decidiu-se pela exploração da distribuição dos ranks produzidos por algumas das estratégias, com o objetivo de evidenciar possíveis outliers que exercem grande influência sobre o rank médio. Foram comparadas as estratégias simples e as três estratégias híbridas, conforme a Figura 1.

A Figura indica que, apesar de as estratégias testadas classificarem os pares corretos em ranks elevados, em média, uma relevante fração das descrições corretas é classificada em baixos ranks. Isso permitiria, em futuros trabalhos, a investigação de estratégias em dois estágios, onde a primeira estratégia funcionaria como filtro inicial.

Também fica ilustrada a diferença entre as estratégias. Para um quantil de 80%, precisa-se verificar as 436 descrições de destino com maiores similaridades para que seja garantido encontrar o par correto, caso seja usada a estratégia M_J. Por outro lado, com o uso de M_{H2}, apenas as 82 descrições de maior similaridade produziriam o mesmo quantil. Alguns dos principais quantis são apresentados na Tabela 5.

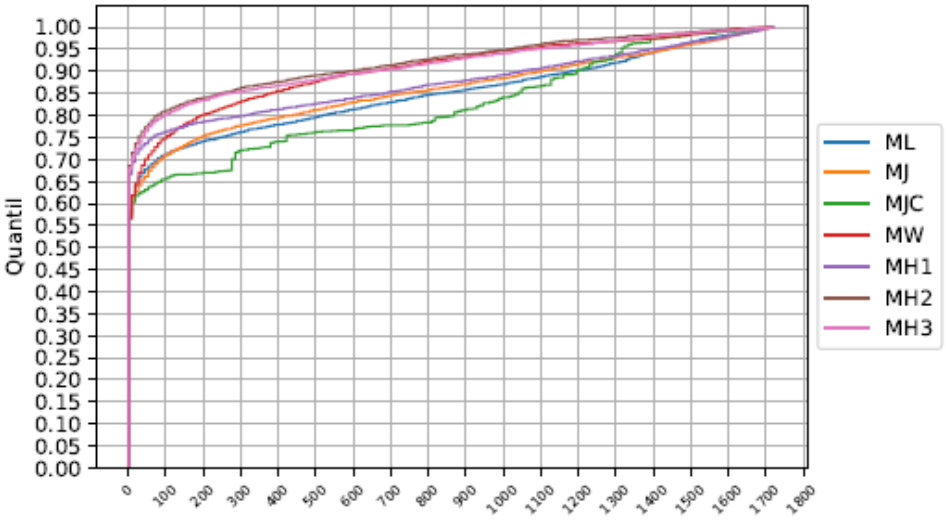

Figura 1. Rank do par correto produzido pela estratégia

Tabela 5. Rank do par correto para diferentes quantis e matrizes

		Quantil							
		50	**60**	**70**	**80**	**85**	**90**	**95**	**97,5**
Matriz	M_J	2	14	89	436	741	1103	1440	1596
	M_{H1}	2	3	20	317	689	1053	1404	1575
	M_{H2}	1	3	14	82	252	592	1016	1275
	M_{H3}	1	3	16	103	273	648	1087	1356

Por fim, de forma a ilustrar casos concretos de pareamentos produzidos pelas estratégias abordadas, são apresentados alguns casos específicos. Em cada caso são apresentados os pares corretos e os pareamentos produzidos pelas estratégias. Entre parênteses indica-se em que rank de similaridade a descrição correta foi classificada de acordo com a respectiva estratégia.

O primeiro caso apresentado na Tabela 6 ilustra a dificuldade de realização de um correto pareamento através de medidas de similaridade que atuam apenas nos níveis alfabético e léxico quando as descrições são razoavelmente longas. Neste caso, algumas desinências cumprem papel importante para aumentar a medida de similaridade, como a terminação “mento”, encontrada nos pares de maior similaridade retornados pela medida de Levenshtein – “Conjunto de latas de mantimentos”, “Varal (de apartamento)” e “Balança para alimentos”. Já a similaridade semântica retornou pares que possuíam alta similaridade com as strings “Vasilhame” e “plástico” – “Utensílios de isopor”, “Utensílios de vidro e louça” e “Utensílios de plástico”. Em que pese a quase perfeita similaridade entre ‘plastico’ e ‘plástico’ – já que diferem pelo acento gráfico, conforme grafia encontrada na base de dados –, o par correto foi apenas o terceiro de maior similaridade, devido as maiores similaridades semânticas entre “Vasilhame” e “mantimentos” com “isopor” e “vidro”.

No segundo exemplo, apresentado na Tabela 7, nota-se ainda mais fortemente a dificuldade que a estratégia baseada na similaridade de Levenshtein (ou outras que baseiam-se nos componentes alfabéticos) enfrenta. Uma vez que há significativa diferença entre os comprimentos das strings que compõem as Descrições POF e IPC,

juntamente com a existência de descrições de destino que possuem porção relevante de caracteres iguais com a Descrição POF – “Consulta” e “Conserto”, neste caso –, o pareamento não é corretamente realizado, além de classificar o par correto – “Médico” – como altamente dissimilar. Por outro lado, a similaridade semântica retorna descrições que possuem alta similaridade com as palavras componentes.

No terceiro exemplo, apresentado na Tabela 8, tem-se a ilustração de um caso diferente: quando a falta de uma acentuação produz palavras parônimas, onde há grande similaridade na grafia, mas grande dissimilaridade no sentido. Aqui as medidas baseadas em distância de edição produzem alta similaridade entre “Cara” e “Cará”, uma vez que há apenas uma edição necessária para que se transforme a primeira palavra na segunda. Por outro lado, a palavra “Cara” possui alta similaridade com diversas palavras presentes na base de dados, como “cabeça”, “mulher”, “moça” e “criança”, o que explica o incorreto resultado do emprego da estratégia baseada em semântica. Pode-se notar, nesses exemplos e nos anteriores, o efeito suavizador causado pela mistura de métricas, que impede que classificações discrepantes prejudiquem em demasia o desempenho na tarefa.

Finalizam-se os exemplos na Tabela 9, um caso em que nenhuma das medidas, tanto isoladamente quanto em conjunto, chegou próxima de realizar o pareamento correto.

5. Conclusões e Trabalhos Futuros

Este trabalho abordou o problema da tarefa de pareamento entre textos curtos, através da utilização de medidas de similaridades textuais que atuam em diferentes níveis: alfabético, léxico e semântico.

Os resultados dos experimentos realizados sugerem que a combinação de diferentes medidas melhorou, ainda que discretamente, a acurácia das tarefas de pareamento, além de aproximar o par correto das posições de máxima similaridade. A estratégia que emprega a média simples das medidas de similaridade utilizadas acerta cerca de 53% dos 3.910 casos propostos. Além disso, 80% dos pares corretos encontram-se entre os 80 textos candidatos de maior similaridade. Os resultados podem servir para futuros trabalhos, em que uma estratégia em dois estágios poderia ser conduzida, onde o primeiro estágio serviria como filtro inicial. Ademais, tal resultado poderia ser utilizado como facilitador para uma abordagem semiautomatizada, em que um supervisor humano precisaria escolher o par candidato dentre um conjunto substancialmente menor de textos, quando comparado ao conjunto total.

Diversos refinamentos metodológicos poderiam ser utilizados para futuros trabalhos. Inicialmente, cumpre mencionar que apenas uma medida de similaridade semântica foi investigada: o cosseno entre embeddings semânticos construídos através da arquitetura Word2vec. Conforme mencionado no texto, tal arquitetura baseia-se fortemente na proximidade espacial entre as palavras em um texto, o que pode não se adequar de maneira ótima para o experimento proposto. Com a grande presença de hipônimos e hiperônimos na base de textos utilizadas, medidas baseadas em ontologias [Anuar et al., 2016], que capturam naturalmente as estruturas de dependência entre as palavras, podem produzir melhores acurácias. Também, o uso de um embedding médio para descrições com múltiplas palavras pode diluir o núcleo semântico da descrição entre as várias palavras auxiliares, prejudicando a performance.

Ademais, é crítica a escolha dos textos que serão usados para construção dos embeddings semânticos, conforme apresentado em [Gomes et al. 2018, Gomes et al. 2021]. Esses trabalhos demonstram a diferença de performance que ocorre quando utiliza-se embeddings construídos a partir de textos de notícias, como os de [NILC 2017], e embeddings construídos com base em textos do domínio específico do problema que está sendo tratado. Desta forma, para o experimento proposto aqui, embeddings construídos com base em catálogos de produtos e serviços – como os disponíveis em portais governamentais Painel de Preços e Comprasnet – poderiam melhorar a performance da tarefa de pareamento.

Por fim, não foi empregada uma importante dimensão das estruturas textuais: a dimensão sintática [Sinoara et al. 2017]. Técnicas de análise sintática como *PoS tagging* [Jurafrsky and Martin 2020] poderiam ser empregadas para atribuir pesos desiguais para funções sintáticas diferentes, o que poderia ajudar a dar maior relevância para palavras que denotam o núcleo sintático de um texto.

Tabela 6. Pareamento do nome "Vasilhame plastico de mantimentos"

		Descrição IPC		
		Utensílios de plástico		
		M_L	**M_W**	**M_{H2}**
Descrição POF	**Vasilhame plastico de mantimentos**	Conjunto de latas de mantimentos (231)	Utensílios de isopor (3)	Artigos de lona ou plástico para acampamento (7)

Tabela 7. Pareamento do nome "Consulta medica com otorrinolaringologista"

		Descrição IPC		
		Médico		
		M_L	**M_W**	**M_{H2}**
Descrição POF	**Consulta medica com otorrinolaringologista**	Conserto de máquina fotográfica, flash (1375)	Consulta com terapeuta ocupacional (3)	Consulta com terapeuta ocupacional (8)

Tabela 8. Pareamento do nome "Cara"

		Descrição IPC		
		Cará		
		M_L	**M_W**	**M_{H2}**
Descrição POF	**Cara**	Cará (1)	Faixa de cabeça (mulher) (1233)	Cera (2)

Tabela 9. Pareamento do nome "Buzina"

		Descrição IPC		
		Acessórios e peças		
		M_L	**M_W**	**M_{H2}**
Descrição POF	**Buzina**	Abobrinha (738)	Campainha musical (646)	Cortina (1200)

Referências

Anuar, F. M., Setchi, R., Lai, Y-K. (2016). Semantic retrieval of trademarks based on conceptual similarity. In *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 46(2), pages 220–233. IEEE.

Davis Jr., C. A. and Salles, E. (2009) "Approximate String Matching for Geographic Names and Personal Names", In: Proc. of the IX GEOINFO, INPE, p. 49–60.

Dumont, E. and Mérialdo, B (2010). Rushes video summarization and evaluation. In *Multimedia Tools and Applications*, 48(1), p. 51–68. Springer.

Fellegi, I. and Sunter, A. A. (1969). A theory for record linkage. In *Journal of the American Statistical Association*, 64, pages 1183–1210. Taylor & Francis Group.

Gomes, D. S. M.; Cordeiro, F. C.; Evsukoff, A. G. (2018). "Word Embeddings em Português para o Domínio Específico de Óleo e Gás". In: Rio Oil Gas 2018.

Gomes, D. S. M et al. (2021). Portuguese word embeddings for the oil and gas industry:Development and evaluation. In *Computers in Industry*, 124, 103347. Elsevier.

IBGE (2020). Estruturas de ponderação a partir da Pesquisa de Orçamentos Familiares 2017-2018. Rio de Janeiro. Disponível em: https://biblioteca.ibge.gov.br/visualizacao/livros/liv101711.pdf. Acesso em: 17 dez. 2021.

Jaccard, P. (1912). The distribution of the flora in the alpine zone. In *New Phytologist*, 11, p. 37–50.

Jurafsky, D. e Martin, J. H. (2020), Speech and Language Processing, Stanford, 3rd edition.

Leskovec, J., Rajaraman, A. e Ullman, J. (2020), Mining of Massive Datasets Cambridge University Press, 3rd edition.

Lemstrom, K. and Perttu, S. (2000). "Semex – An Efficient Music Retrieval Prototype". In: ISMIR - International Symposium on Music Information Retrieval. USA: [S. n.], 2000

Levenshtein, V. I. Binary codes capable of correcting deletions, insertions, and reversals. In *Cybernetics and Control Theory*, 10(8), pages 707–710.

Lhoussain, A. S., Hicham, G., Abdellah, Y. (2015). Adaptating the levenshtein distance to contextual spelling correction. In *International Journal of Computer Science and Applications*, 124, pages 127–133. ENSIAS.

Meirelles, T., Gonçalves, E., Gomes, D. (2021). "Uma Estratégia Híbrida para o Pareamento de Textos Curtos Baseada em Similaridade Léxica e Embeddings Semânticos". In: Anais da IV Escola Regional de Informática do Rio de Janeiro (ERI-RJ), SBC, p. 33–40.

Mikolov, T., et al. (2013). "Distributed Representations of Words and Phrases and Their Compositionality", In: Proc. of the 26th Intl' Conf. on Neural Information Processing Systems (NIPS), Neurips, p. 3111–3119.

Mikolov, T., Chen, K., Corrado, G., e Dean, J. (2013). Efficient estimation of word representations in vector space", In *CoRR, abs/1301.3781*.

NILC - Núcleo Interinstitucional de Linguística Computacional (2017). Repositório de Word Embeddings do NILC. Disponível em: http://www.nilc.icmc.usp.br/embeddings. Acesso em: 17 dez. 2021.

Rubert, D. P. (2019). Distance and Similarity Measures in Comparative Genomics. Tese (Doutorado em Ciência da Computação) – Faculdade de Computação, Universidade Federal de Mato Grosso do Sul, Campo Grande.

Silva et al. (2010). "Inovações no Sistema de Pareamento de Domicílios e Pessoas para a Pesquisa de Avaliação da Cobertura da Coleta do Censo 2010". In: Anais do XVII Encontro Nacional de Estudos Populacionais, ABEP, p. 1–19.

Sinoara, R., Antunes, J., Rezende, S. O. (2017). Text mining and semantics: A systematic mapping study. In *Journal of the Brazilian Computer Society*, 23(9), pages 1–20.

Winkler, W. E. (1990). "String Comparator Metrics and Enhanced Decision Rules in the Fellegi-Sunter Model of Record Linkage". In: Proc. of the Sect. on Surv. Research, ERIC, p. 354–359.

Word2Vec (2013). Disponível em: https://code.google.com/archive/p/word2vec/. Acesso em: 17 dez. 2021.