# Estudo Comparativo de Estratégias para o Pareamento de Nomes de Entidades na Língua Portuguesa

**Antônio Mamede Araújo de Medeiros[1], Eduardo Corrêa Gonçalves[1]**

[1]Escola Nacional de Ciências Estatísticas (ENCE/IBGE)
Rua André Cavalcanti, 106 – 20.231-050 – Rio de Janeiro – RJ – Brasil

aammamede@gmail.com, eduardo.correa@ibge.gov.br

***Abstract.*** *Entity name matching is the task of automatically linking entity names from a list A with those from another list B, according to their similarity. There are several modern and important applications for this task, varying from the identification of duplicate records in databases to the optimization of price comparison systems. In this work, we perform an experimental analysis of two techniques for entity name matching in Portuguese. Experiments carried out on a dataset containing product and service names showed that the combination of Jaro-Wikler measure with TF-IDF bag-of-words and word2vec embeddings leads to more accurate results.*

***Resumo.*** *O pareamento de nomes de entidades é a tarefa que consiste em realizar a correspondência automática entre nomes de uma lista A com os de uma outra lista B, considerando a semelhança entre eles. Há muitas aplicações modernas e importantes para a tarefa, variando desde a identificação de registros duplicados em bases de dados até a otimização de sistemas comparadores de preços. Este trabalho realiza a avaliação de duas técnicas propostas para o pareamento de nomes em português. Experimentos realizados em uma base contendo nomes de produtos e serviços mostraram que a combinação da medida Jaro-Winkler com vetores TF-IDF e embeddings word2vec foi capaz de produzir os melhores pareamentos.*

## 1. Introdução

O processamento de linguagem natural [Jurafsky and Martin 2023] é a linha de pesquisa do campo da ciência da computação que tem como principal objetivo fazer com que o computador entenda uma ou mais linguagens naturais, possibilitando a comunicação entre o homem e a máquina.

Um dos problemas que o processamento de linguagem natural procura resolver é o do pareamento de nomes de entidades. Neste problema, o objetivo é realizar a correspondência automática entre nomes contidos em uma lista A com os de uma outra lista B. Dentre as diferentes aplicações práticas, destaca-se o emprego dos algoritmos de pareamento nos sistemas comparadores de preços [Hillen 2019]. Considere, por exemplo, um sistema que compara de preços de produtos vendidos em supermercados. Suponha que um usuário deseje comparar os preços do item “Costela de Porco”. Neste caso, é possível que o produto seja vendido com o nome de “Costela de Porco” em um supermercado, em outro como “Costela Suína” (uso de sinônimo) e em outro como “Cost. Suína” (uso de abreviação). Um sistema comparador eficaz deve ser capaz de tratar estes três nomes com escrita diferente como uma mesma entidade.

Os algoritmos de pareamento empregam medidas de similaridade para executar sua tarefa. Uma medida de similaridade entre nomes de entidades pode ser vista como uma função $S:(s_1, s_2)\rightarrow[0, 1]$, onde $s_1$ e $s_2$ são duas strings, cada uma armazenando um nome [Lin 1998]. Quanto mais próxima a medida é de 1, mais similares as entidades são. De modo geral, as medidas são voltadas para um dos três diferentes níveis de similaridade indicados a seguir: alfabético, léxico ou semântico [Meirelles et al. 2021].

- No nível alfabético, a similaridade é medida pelos caracteres das duas strings. Por exemplo, "barro" e "carro" são palavras similares no nível de alfabético, pois elas possuem uma escrita similar (compartilham muitos caracteres em comum em posições correspondentes), mesmo tendo significados bem distintos. Exemplos de medidas de similaridade que atuam nesse nível são as tradicionais medidas de Levenshtein e Jaro-Winkler [Freire et al. 2009, Gali et al. 2019].
- No nível léxico, a similaridade é medida pela quantidade de palavras (*tokens*) iguais entre duas strings. Por exemplo, "arroz com feijão" e "feijão com arroz" são idênticas sob o ponto de vista da similaridade léxica, uma vez que todas as palavras contidas nas strings são iguais, mesmo estando em ordem diferente. Exemplos de medidas de similaridade que atuam nesse nível são as de Jaccard e o cosseno de vetores TF-IDF [Gali et al. 2019, Jurafsky and Martin 2023].
- No nível semântico, a similaridade é medida pelo significado das palavras. Por exemplo, "porco" e "suíno" são semanticamente similares, pois apesar de terem a escrita diferente, são palavras sinônimas. O cosseno de vetores word2vec [Jurafsky and Martin 2023, Mikolov et al. 2013] é um exemplo de medida de similaridade semântica.

O presente trabalho tem por objetivo comparar e combinar duas técnicas propostas para o casamento de textos curtos em português. A primeira é a técnica de [Hartmann 2016], que foi originalmente elaborada para avaliar similaridade entre frases curtas (e não nomes de entidades) e baseia-se na utilização conjunta de vetores TF-IDF e word2vec. A segunda é a técnica de [Meirelles et al. 2021], que utiliza matrizes de similaridade para combinar e aplicar, de forma simultânea, medidas que atuam nos níveis alfabético, léxico e semântico. Os experimentos foram realizados em uma base de dados contendo 3.305 pares de nomes de produtos e serviços em português.

O restante do artigo está dividido da seguinte forma. Na Seção 2 é apresentada a revisão bibliográfica. Na Seção 3 encontra-se a descrição da base de dados. Na Seção 4 apresenta-se a metodologia empregada para realizar os experimentos. A Seção 5 reporta os resultados. Por fim, as conclusões são apresentadas na Seção 6.

## 2. Trabalhos Relacionados

Esta seção revisa trabalhos sobre similaridade semântica de textos curtos em Português, tendo como foco principal os trabalhos apresentados na competição ASSIN – Avaliação de Similaridade Semântica e de Inferência Textual [Fonseca et al. 2016], que aconteceu durante a Conferência Internacional sobre o Processamento Computacional da Língua Portuguesa (PROPOR) de 2016. Este foi o primeiro workshop específico sobre algoritmos para similaridade de textos curtos em português. A base de dados utilizada na competição possui uma coleção de pares de sentenças em português e para cada par, há um valor de 1 a 5 indicando sua similaridade, de forma que quanto maior o valor,

mais similares são as duas sentenças[1]. A Tabela 1 [Fonseca et al. 2016], exemplifica algumas sentenças da base, apresentando o escore de similaridade entre ambas. Para critério de avaliação dos resultados obtidos pelos competidores, foi usada a correlação de Pearson e o erro quadrático médio. A seguir são apresentadas as técnicas que obtiveram os melhores resultados.

[Alves et al. 2016] extraem do texto características lexicais, sintáticas e semânticas para então as utilizar em algoritmos heurísticos, baseados em conhecimento, e de aprendizagem de máquina. Sua melhor performance obtida foi de um coeficiente de correlação de Pearson de 0,59 e um erro quadrático médio de 1,26.

[Barbosa et al. 2016] utilizam vetores semânticos de palavras de duas formas: vetores de características de pequena dimensão e estratégias de deep learning para vetores de características de grande dimensão. Uma das conclusões é que a primeira estratégia seria mais promissora para a tarefa de similaridade semântica. Obtiveram um coeficiente de correlação de 0,65 e um erro quadrático médio de 0,44.

[Freire et al. 2016] combinam diversos componentes, como *parsers* morfológicos e sintáticos, bases de conhecimento e lexicais, algoritmos de aprendizagem automática e algoritmos de alinhamento e cálculo da similaridade. O treinamento do algoritmo se deu por modelo de *ridge regression*. Em seu melhor resultado, foi obtido um coeficiente de correlação de 0,62 e um erro quadrático médio de 0,47.

[Hartmann 2016] utilizou TF-IDF e vetores semânticos word2vec [Mikolov et al. 2013] para gerar representações vetoriais de cada sentença, somando os vetores de cada palavra. Esses vetores são utilizados como variáveis explicativas em um modelo de regressão linear, o qual a variável resposta é o escore de similaridade. Dos trabalhos apresentados na ASSIN, este foi o que obteve o melhor resultado (foi o campeão da competição), com coeficiente de correlação de 0,70 e erro quadrático médio de 0,38.

**Tabela 1. Exemplos de pares de sentenças da base ASSIN. [Fonseca et al. 2016]**

| Par de sentenças | Escore de similaridade |
|---|---|
| $s_1$: "Mas esta é a primeira vez que um chefe da Igreja Católica usa a palavra em público."<br>$s_2$: "A Alemanha reconheceu ontem pela primeira vez o genocídio armênio." | 1 |
| $s_1$: "Como era esperado, o primeiro tempo foi marcado pelo equilíbrio."<br>$s_2$: "No segundo tempo, o panorama da partida não mudou." | 2 |
| $s_1$: "Houve pelo menos sete mortos, entre os quais um cidadão moçambicano, e 300 pessoas foram detidas."<br>$s_2$: "Mais de 300 pessoas foram detidas por participar de atos de vandalismo." | 3 |
| $s_1$: "A organização criminosa é formada por diversos empresários e por um deputado estadual."<br>$s_2$: "Segundo a investigação, diversos empresários e um deputado estadual integram o grupo." | 4 |
| $s_1$: "Outros 8.869 fizeram a quadra e ganharão R$ 356,43 cada um."<br>$s_2$: "Na quadra 8.869 apostadores acertaram, o prêmio é de R$ 356,43 para cada." | 5 |

[1] Essa faixa de valores de similaridade difere da tradicionalmente adotada pela literatura, que atribui o valor 1,0 para denotar a similaridade máxima entre duas entidades, frases ou documentos [Lin 1998].

No presente trabalho, a técnica proposta por Hartmann (2016), que obteve o melhor desempenho na ASSIN e foi originalmente proposta para determinar o escore de similaridade semântica entre duas frases, será empregada para realizar o pareamento de nomes de produtos e serviços. Ou seja, ela será aplicada em um outro tipo de problema que envolve textos curtos em português. A abordagem de [Hartmann 2016] será comparada e combinada com a técnica de matrizes de similaridade recentemente proposta em [Meirelles et al. 2021], que foi especificamente projetada para o casamento de nomes de entidades e baseia-se na utilização simultânea de funções de similaridade que atuam nos níveis alfabético, léxico e semântico (detalhes são apresentados na Seção 4). Os experimentos foram realizados em uma base composta por nomes de produtos e serviços de duas pesquisas do IBGE, nomes estes que são ainda mais curtos e com menos informação do que os textos da base de dados da ASSIN, conforme apresenta-se na seção a seguir.

## 3. Base de Dados

A base de dados[2] deste trabalho contém pares de nomes de produtos e serviços da Pesquisa de Orçamentos Familiares (POF) [IBGE 2021], realizada entre os anos de 2017 e 2018 pelo IBGE, casados com os nomes de entidades adotados pelo Sistema Nacional de Índices de Preços (SNIPC) [IBGE 2016]. A Tabela 2 mostra alguns pares de nomes da base. A primeira coluna, Descrição POF, contém os nomes das entidades no banco de dados da POF, enquanto a segunda, Descrição SNIPC, apresenta os nomes no banco de dados do SNIPC. A base contém todos os nomes de itens da POF devidamente casados com o item adequado do SNIPC. Os casamentos foram feitos manualmente por técnicos do IBGE. De uma maneira geral, os nomes dos produtos e serviços são muito curtos em ambas as pesquisas. Para a POF em média, cada item possui 22 caracteres e 3 palavras, já para o SNIPC 15 caracteres e 2 palavras.

A quantidade de palavras únicas na POF é quase o dobro do SNIPC, indicando que os produtos e serviços do SNIPC são mais agregados. Por exemplo, a POF possui itens como “Costela Suína”, “Lombo Suíno”, “Carré” enquanto no SNIPC só existe “Carne de Porco”. A Figura 1 apresenta boxplots com o tamanho das descrições da POF e do SNIPC, medidas em caracteres e palavras. Note a baixa dispersão no número de palavras das descrições do SNIPC, ilustrada no segundo boxplot da Figura 1. Quanto menos palavras, mais difícil se torna a tarefa de pareamento, visto que há pouca informação no texto (são muito curtos, podendo ser compostos por apenas 1 ou 2 palavras, ao contrário do que ocorre nas sentenças da base de dados da ASSIN).

**Tabela 2. Alguns exemplos de paras de nomes de entidades da base POF-SNIPC**

| Nome POF | Nome SNIPC |
|---|---|
| Cesto de Lixo (Plástico) | Lata de Lixo |
| Cinema (Ingresso) | Cinema, Teatro e Concertos |
| Telefone Fixo | Plano de Telefonia Fixa |
| Carré | Carne de Porco |
| Ultrassonografia | Exame de Imagem |
| Aula de Judô | Atividades Físicas |

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

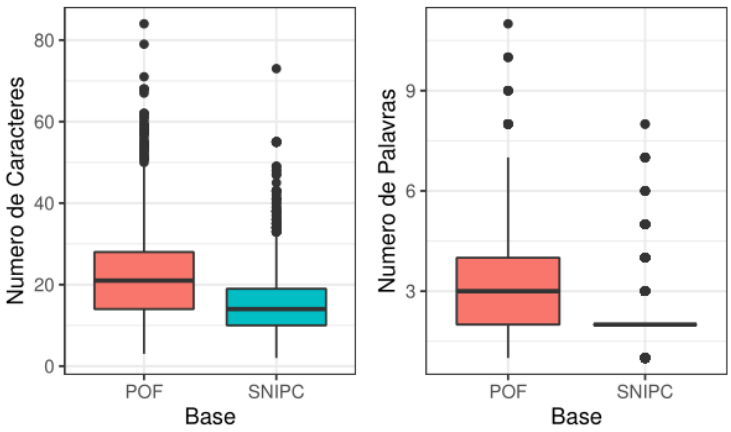


**Figura 1. Boxplots com os tamanhos de caracteres e palavras dos nomes de entidades da POF e do SNIPC**

## 4. Metodologia

A metodologia utilizada para a realização dos experimentos reportados no presente artigo teve por base a geração das matrizes de similaridade propostas em [Meirelles et al. 2021]. Dada qualquer medida de similaridade $S$, é possível criar uma matriz de similaridade $M$ de forma que cada linha representa um nome de entidade presente na lista origem ($l_o$) e cada coluna um nome constante na lista destino ($l_d$). Os valores nas células da matriz representam a similaridade entre o $i$-ésimo elemento de $l_o$ e o $j$-ésimo elemento de $l_d$.

A Tabela 3 apresenta um exemplo de matriz de similaridade hipotética obtida por uma dada medida $S$. Nesse exemplo, $l_o$ e $l_d$ contêm 4 e 6 nomes, respectivamente. Veja que a similaridade entre os nomes "arroz polido" e "arroz" é igual a 0,88, de acordo com $S$, enquanto o par "academia" e "jogos de azar" possui similaridade de 0,56. Na tabela apresentada, os pares corretos encontram-se representados por cores correspondentes (por exemplo, o par correto de "arroz polido" é "arroz", o de "maisena" é "amido de milho" etc.).

### 4.1. Estratégias de Pareamento

As matrizes de similaridade podem ser utilizadas para resolver pareamentos de forma direta e simples: cada nome de entidade representado em uma linha da matriz (nome proveniente de $l_o$) será pareado com o nome representado na coluna (nome proveniente de $l_d$) que tiver maior valor de similaridade.

**Tabela 3. Matriz de similaridade hipotética [Meirelles et al. 2021]**

| | | 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 |

Entretanto, o trabalho de Meirelles et al. (2021) propôs a utilização de uma estratégia que consiste na geração de uma matriz de similaridade híbrida $M_H$. Esta matriz combina valores de duas ou mais matrizes diferentes. Neste caso, a vantagem está na possibilidade de combinar resultados obtidos por matrizes geradas por medidas que atuam nos níveis alfabético, léxico e semântico. Uma matriz $M_H$ possui os elementos definidos de acordo com a Equação 1. Nesta equação, $M_1$, $M_2$, ..., $M_n$ são $n$ matrizes de similaridade escolhidas previamente e $\alpha \in \mathbb{R}^+$ atua como ponderação, possibilitando dar maior peso para valores de similaridade maiores.

$$M_H(i,j) = \frac{1}{n}\left(M^{\alpha}_{1\,(i,j)} + M^{\alpha}_{2\,(i,j)} + \cdots + M^{\alpha}_{n\,(i,j)}\right) \quad (1)$$

Neste trabalho, será utilizado $\alpha = 2$, pois este valor produziu os melhores resultados em [Meirelles et al. 2021].

### 4.2. Métricas de Desempenho

As seguintes métricas de desempenho podem ser empregadas para avaliar a qualidade dos pareamentos obtidos a partir de uma matriz de similaridade:

- **Acurácia Estrita**: É definida como a proporção de vezes em que a estratégia pareou única e corretamente o texto de origem. Considerando a matriz da Tabela 3, o valor resultante desta medida seria (0 + 0 + 1 + 0) / 4 = 0,25. Ou seja, para a acurácia estrita foi considerado apenas um valor, pois houve apenas uma ocorrência onde há o pareamento correto, sem empates, no caso de “queijeira” e “utensílios de plástico”, com uma similaridade hipotética de 0,57.
- **Acurácia Ponderada**: Relaxa a restrição de que o pareamento deva ser único, mas penaliza proporcionalmente estratégias que produzam conjuntos pareados com muitos elementos. No exemplo da Tabela 3, o valor computado seria (0 + 0 + 1 + 0,33) / 4 = 0,33. Isso porque, na acurácia ponderada, contabiliza-se o pareamento correto citado anteriormente e, além disso, acrescenta-se o pareamento de “academia” com “atividades físicas”. No entanto, note que com peso menor, de apenas 0,33 pois apesar do pareamento correto ocorrer, há um empate no valor da similaridade do nome “academia” com os nomes “atividade física” (par correto), “jogos de azar” e “arroz pré cozido” (pares incorretos).
- **Posição Média**: É a posição média do par correto, isto é, após ordenada a lista de nomes de destino para um determinado nome de entidade de origem, registra-se o rank do par correto. A posição média, será a média dos ranks registrados. Por exemplo, para a matriz da Tabela 3, a posição média é dada por: (2 + 6 + 1 + 1) / 4 = 2,50. Neste caso, no pareamento do nome “arroz polido”, o par correto, “arroz”, possui o 2º maior valor de similaridade, enquanto o par “maisena” e “amido de milho”, possui o 6º maior valor de similaridade. Já para os pares “queijeira” e “utensílios de plástico” e “academia” e “atividades físicas”, o maior valor de similaridade é o do par correto.

## 5. Experimentos

Nesta seção são descritos os resultados obtidos nos experimentos de pareamento de nomes da base do IBGE. Todos os experimentos foram realizados localmente em um computador com as seguintes especificações: Windows 11, processador Ryzen 5 3600,

placa de vídeo NVIDIA RTX 2060 Super e 16GB de memória RAM. Para os cálculos de similaridade foi utilizada a linguagem Python v. 3.9.12, sendo o pacote Gensim [Radim and Sojka 2010] utilizado para carregar os embeddings semânticos word2vec de 300 dimensões desenvolvidos por [NILC 2017], o pacote strsimpy [Strsimpy 2023] para o cálculo das funções de similaridade de Levenshtein, Jaro-Winkler e Jaccard, além do pacote scikit-learn [Pedregosa et al. 2011], utilizado para calcular os valores TF-IDF. Todos os códigos utilizados neste trabalho estão disponíveis num repositório online do GitHub[3]. Para a utilização do word2vec foi necessária a realização de uma etapa adicional de pré-processamento, pois algumas palavras contidas na base de dados não possuíam um embedding semântico treinado pelo NILC, como por exemplo, "mp3" e "polenguinho". Nesses casos, foi usado um embedding de 300 dimensões com o valor 0.

Com relação à base de dados, as seguintes tarefas de pré-processamento foram executadas: (i) conversão das descrições para minúsculo; (ii) acentuação de palavras dos nomes de entidades da POF; (iii) remoção de sinais de pontuação; (iv) remoção de stop words, como artigos, preposições etc.; (v) exclusão de linhas com o pareamento idêntico (ou seja, onde o Nome POF é igual ao Nome SNIPC). Ao final do pré-processamento, chegou-se a uma base de dados com um total de 3.305 pares de nomes.

O primeiro experimento realizado consistiu em uma comparação entre o desempenho isolado das matrizes de similaridade produzidas pelas medidas de Levenshtein (nível alfabético), Jaro-Winkler (nível alfabético), Jaccard (nível léxico), cosseno de vetores TF-IDF (nível léxico) e cosseno de vetores word2vec (nível semântico). Ou seja, foram geradas 5 matrizes, uma para cada medida (para detalhes sobre essas medidas consulte [Jurafsky and Martin 2023, Gali et al. 2019]). Os resultados são apresentados na Tabela 4. A primeira coluna indica o nome da medida utilizada, enquanto as colunas 2, 3 e 4 mostram os valores de Acurácia Estrita, Acurácia Ponderada e Posição Média, respectivamente, calculadas de acordo com o procedimento descrito na seção anterior. Os melhores resultados estão em negrito.

A medida do cosseno de vetores TF-IDF obteve desempenho destacado, produzindo o maior valor para ambas as acurácias, 0,5371 e 0,5376 para estrita e ponderada, respectivamente. Isto é, conseguiu acertar pouco mais da metade dos pareamentos da base. Este resultado é consonante com [Hartmann 2016], onde a medida TF-IDF empregada de forma isolada na base de dados ASSIN obteve desempenho superior ao da maioria das outras estratégias (incluindo as baseadas em deep learning). Já o melhor desempenho com relação à medida da Posição Média foi obtido pelo word2vec: 73,83. Ou seja, apesar do word2vec não ser a técnica que mais acerta é a que, em média, o casamento correto está mais próximo do que as demais medidas.

**Tabela 4. Performance de pareamento – matrizes individuais**

| Matriz | Acurácia Estrita | Acurácia Ponderada | Posição Média |
|---|---|---|---|
| Levenshtein ($M_L$) | 0,3192 | 0,3401 | 190,79 |
| Jaro-Winkler ($M_J$) | 0,4118 | 0,4127 | 184,43 |
| Jaccard ($M_{JC}$) | 0,4738 | 0,5185 | 172,94 |
| TF-IDF ($M_{TF}$) | **0,5371** | **0,5376** | 173,49 |
| Word2vec ($M_{W2V}$) | 0,4291 | 0,4291 | **73,83** |

[3] https://github.com/antmamede/Casamento-Semantico-de-Textos-Curtos

O segundo experimento realizou a comparação de quatro matrizes híbridas ($M_{H1}$, $M_{H2}$, $M_{H3}$ e $M_{H4}$) geradas a partir de quatro diferentes combinações de medidas de similaridade. A escolha da combinação presente em cada matriz híbrida se deu pelos motivos descritos a seguir.

- Em $M_{H1}$ foi utilizada a combinação que obteve as melhores acurácias no trabalho de Meirelles et al. (2021): Levenshtein, Jaro-Winkler, Jaccard e word2vec;
- Em $M_{H2}$, utilizou-se todas as cinco medidas de similaridade avaliadas no primeiro experimento;
- Em $M_{H3}$, utilizou-se a combinação adotada no trabalho de Hartmann (2016), vencedor na competição ASSIN: TF-IDF e word2vec;
- Por fim, em $M_{H4}$ foram escolhidas: a medida de similaridade no nível alfabético com melhor desempenho de acordo com o primeiro experimento (Jaro-Winkler); a medida do nível léxico com melhor desempenho no primeiro experimento (TF-IDF); e, por fim, uma medida que atua no nível semântico (word2vec).

Os resultados obtidos são apresentados na Tabela 5. A estratégia $M_{H1}$ obteve acurácias menores que as do melhor resultado do primeiro experimento (TF-IDF isoladamente), mas em compensação obteve uma melhor posição média, em relação ao TF-IDF. Já a estratégia $M_{H2}$ obteve tanto acurácias maiores quanto posição média menor, em relação ao TF-IDF utilizado isoladamente. Para $M_{H3}$ as acurácias foram menores que as do melhor resultado do experimento 1, mas a posição média foi a menor obtida até agora, indicando que em média, o casamento correto está mais próximo do que as demais estratégias. Por fim, $M_{H4}$ obteve acurácias maiores, porém uma posição média maior do que a da estratégia $M_{H3}$ (piora causada pela introdução da função de Jaro-Winkler que não gera bons resultados para a posição média).

Como terceiro e último experimento, a base de dados TeP 2.0 [Mazieiro et al. 2008], que contém sinônimos para diversas palavras da língua portuguesa, foi empregada para acrescentar sinônimos para as palavras presentes nos nomes de entidades da POF e SNIPC. Essa mesma estratégia foi adotada por Hartmann (2016). Após a inserção dos sinônimos, utilizou-se método $M_{H4,}$ que obteve a melhor acurácia no experimento anterior. O resultado é apresentado na Tabela 6.

**Tabela 5. Performance de pareamento – matrizes híbridas**

| Matriz | Acurácia Estrita | Acurácia Pond. | Posição Média |
|---|---|---|---|
| Levenshtein + Jaro-Winkler + Jaccard + word2vec ($M_{H1}$) | 0,5322 | 0,5322 | 95,16 |
| Levenshtein + Jaro-Winkler + Jaccard, + TF-IDF + word2vec ($M_{H2}$) | 0,5625 | 0,5625 | 94,44 |
| TF-IDF + word2vec ($M_{H3}$) | 0,5340 | 0,5340 | **69,55** |
| Jaro-Winkler + TF-IDF + word2vec ($M_{H4}$) | **0,5673** | **0,5673** | 89,26 |

**Tabela 6. Performance de pareamento – base acrescida dos sinônimos do TeP 2.0**

| Matriz | Acurácia Estrita | Acurácia Pond. | Posição Média |
|---|---|---|---|
| Jaro-Winkler + TF-IDF + word2vec ($M_{H4}$) na base de dados acrescida dos sinônimos do TeP 2.0. | 0,4708 | 0,4708 | 97,78 |

O resultado apresentado na Tabela 6 indica que a inserção dos sinônimos na realidade levou a uma queda no desempenho do pareamento. A principal suspeita para o motivo dessa piora se dá pelo fato de que diversas palavras podem possuir sinônimos que pertencem a contextos diferentes. Por exemplo, com a inserção de sinônimos, o nome de entidade "doce de abóbora" se torna "doce de abóbora afável meigo jerimum". A palavra "doce" de fato é sinônima "afável" e "meigo" mas no contexto dos nomes de produtos da POF ou SNIPC, o resultado desejado seria algo como "açucarado" ou "guloseima".

## 5. Conclusões

Este trabalho realizou um estudo comparativo de diferentes medidas de similaridade, usadas isoladamente e combinadas, para resolver o problema de pareamento de nomes de entidades. Os experimentos foram realizados em uma base de dados contendo 3.305 pares de nomes de produtos e serviços em português utilizando as matrizes de similaridade propostas em [Meirelles et al. 2021] como ferramenta para a realização dos pareamentos.

Ao utilizar cada uma das medidas de similaridade de forma isolada, duas se destacaram: o cosseno de vetores TF-IDF, por apresentar uma acurácia ponderada de 0,5376, a maior dentre todas as outras medidas, e o cosseno de vetores word2vec por ter obtido uma posição média de 73,83, a menor entre os métodos individuais. Com o intuito de combinar medidas de forma que se consiga empregar de forma simultânea a avaliação dos níveis alfabético, léxico e semântico de similaridade, foram produzidas matrizes híbridas. Ao todo, quatro combinações foram testadas e destacou-se a combinação entre os métodos Jaro-Winkler, TF-IDF e word2vec, que obteve acurácia ponderada de 0,5673. Este resultado sugere que a combinação de medidas que atuam nos três diferentes níveis de similaridade aumenta a eficácia do processo de pareamento.

Alguns trabalhos recentes apontam que o emprego de técnicas como BERT e Redes Neurais Siamesas vêm se demonstrando eficazes para a avaliação da similaridade entre documentos e textos curtos [de Souza et al. 2019, Romualdo et al. 2021]. Como trabalho futuro, pretende-se avaliar a eficácia destas técnicas no problema do pareamento de nomes de entidades.

## Referências


Alves, A. O., Rodrigues, R. and Oliveira, H. G. (2016). ASAPP: alinhamento semântico automático de palavras aplicado ao português. In *Linguamática*, 8(2):43–58.

Barbosa, L., Cavalin, P., Guimarães, V. and Kormaksson, M. (2016). Blue man group no assin: Usando representações distribuídas para similaridade semântica e inferência textual. In *Linguamática*, 8(2):15–22.

de Souza, J. V. A., et al. (2019). "Multiple Feature Groups to a Siamese Neural Network for Semantic Textual Similarity Task in Portuguese Texts", In: Proc. of the ASSIN 2 Shared Task: Evaluating Semantic Textual Similarity and Textual Entailment in Portuguese (ASSIN@STIL), SBC, p. 59–68.

Fonseca, E. R., Santos, L. B., Criscuolo, M. and Aluísio, S. M. (2016). Visão geral da avaliação de similaridade semântica e inferência textual. In *Linguamática*, 8(2):3–13.

Freire, J., Pinheiro, V. and Feitosa, D. (2016). FlexSTS: Um framework para similaridade semântica textual. In *Linguamática*, 8(2):23–31.

Freire, S. M., et al. (2009). "Análise da Efetividade de Comparadores de Strings para Discriminar Pares de Verdadeiros de Pares Falsos no Relacionamento de Registros". In: IX Workshop de Informática Médica, SBC, p. 2119 – 2128

Gali, N., Mariescu-Istodor, R., Hostettler and D., Fränti, P. (2019). Framework for syntactic string similarity measures. In *Expert Systems with Applications*, 129(2019):169–185.

Hartmann, N. S. (2016). Solo queue at ASSIN: Combinando abordagens tradicionais e emergentes. In *Linguamática*, 8(2):59–64.

Hillen, J. (2019). Web scraping for food price research. In *British Food Journal*, 121(12):3350–3361.

IBGE (2016), Para compreender o INPC (um texto simplificado), IBGE, 7a. ed.

IBGE (2021), Pesquisa de orçamentos familiares 2017-2018, IBGE. https://www.ibge.gov.br/estatisticas/sociais/saude/24786-pesquisa-de-orcamentos-familiares-2.html?=&t=o-que-e. Acesso em: 27 fev. 2023.

Jurafsky, D. and Martin, J. H. (2023), Speech and Language Processing, Stanford, 3rd edition (draft).

Lin, D. (1998) "An Information-Theoretic Definition of Similarity", In: Proc. of the 15th Int'l Conf. on Machine Learning (ICML), ACM, p. 296–304.

Mazieiro, E. G., et al. (2008). "A base de dados lexical e a interface web do TeP 2.0: thesaurus eletrônico para o Português do Brasil", In: Proc. of the XIV Brazilian Symposium on Multimedia and the Web (WEBMEDIA), ACM, p. 390–392.

Meirelles, T. P., Gonçalves, E. C. and Gomes, D. T. (2021). Pareamento de nomes de produtos e serviços utilizando medidas de similaridade textual nos níveis alfabético, léxico e semântico. In *Cadernos do IME – Série Informática*, 46:104–117.

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.

NILC - Núcleo Interinstitucional de Linguística Computacional (2017). Repositório de Word Embeddings do NILC. http://www.nilc.icmc.usp.br/embeddings. Acesso em: 17 fev. 2023.

Pedregosa et al. (2011). Scikit-learn: Machine learning in python. In *Journal of Machine Learning Research*, 12:2825–2830.

Radim, R. and Sojka, P. (2010). "Software Framework for Topic Modelling with Large Corpora", In: Proc. of the LREC 2010 Workshop on New Challenges for NLP Frameworks, ELRA, p. 45–50.

Romualdo, A. S., Real, L. and Caseli, H. M. (2021). "Measuring Brazilian Portuguese Product Titles Similarity using Embeddings", In: Anais do XIII Simpósio Brasileiro de Tecnologia da Informação e da Linguagem Humana (STIL), SBC, p. 121–132.

Strsimpy 0.2.1 (2023) https://pypi.org/project/strsimpy/. Acesso em: 27 fev. 2023.