Uma Estratégia Híbrida para o Pareamento de Textos Curtos Baseada em Similaridade Léxica e Embeddings Semânticos

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. This work proposes a new hybrid strategy focused on the matching of short texts, such as names of products, brands, and services. The proposed strategy is based on the combination of lexical similarity measures and semantic embeddings generated with the Word2vec model. Preliminary experiments on a real-world dataset comprised of product and service names have shown promising results.*

Resumo. *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. Este trabalho propõe uma nova estratégia híbrida que tem por foco o pareamento de textos curtos, como nomes de produtos, marcas e serviços. A estratégia proposta baseia-se na combinação de medidas de similaridade léxica e embeddings semânticos gerados através do modelo Word2vec. Experimentos preliminares realizados em uma base de dados real contendo nomes de produtos e serviços revelam resultados promissores.*

1. Introdução

Pareamento de textos consiste na 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 [Anuar et al. 2016, 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.

Apesar do sucesso em várias aplicações [Davis Jr. and Salles 2009, Francisco and Ambrosio 2016, Silva et al. 2010], 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 [Jurafsky and Martin 2020]. 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 léxica com uma abordagem híbrida, que incorpora uma medida de similaridade semântica baseada em embeddings gerados pelo modelo Word2vec [Mikolov et al. 2013a, Mikolov et al. 2013b]. 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

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 no nível 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, é definida de acordo com a Equação 1:

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

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 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. Um exemplo é a Similaridade Jaccard [Lescovec et al., 2020], definida na Equação 3. Nesta equação, $tok(s_1)$ e $tok(s_2)$ representam respectivamente, o conjunto de tokens que formam s_1 e s_2.

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

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 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 utilizada neste estudo é composta por 4.956 pares de descrições de produtos e serviços objeto de despesas de famílias residentes em regiões metropolitanas das principais cidades brasileiras[1]. A Tabela 1 relaciona alguns exemplos.

[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

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

Esta subseção apresenta os resultados obtidos nos experimentos de pareamento. O experimento foi conduzido utilizando a linguagem Python v. 3.8.6, com o uso das bibliotecas Gensim[2] para carregamento dos embeddings semânticos de 300 dimensões disponibilizados em [NILC 2017] e strsimpy[3], para cálculo das funções de similaridade.

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.

Tabela 3. Performance de Pareamento – Estratégias Simples

Estratégia	Acurácia Estrita	Acurácia Ponderada	Posição Média
Levenshtein (S_L)	0,3803	0,3946	592,7
Jaro (S_J)	**0,4514**	**0,4522**	558,3
Jaccard (S_{JC})	0,3731	0,4124	692,5
Semântica (S_W)	0,3900	0,3900	**375,8**

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

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

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 4. Performance de Pareamento – Estratégias Híbridas

Estratégia	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. 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.

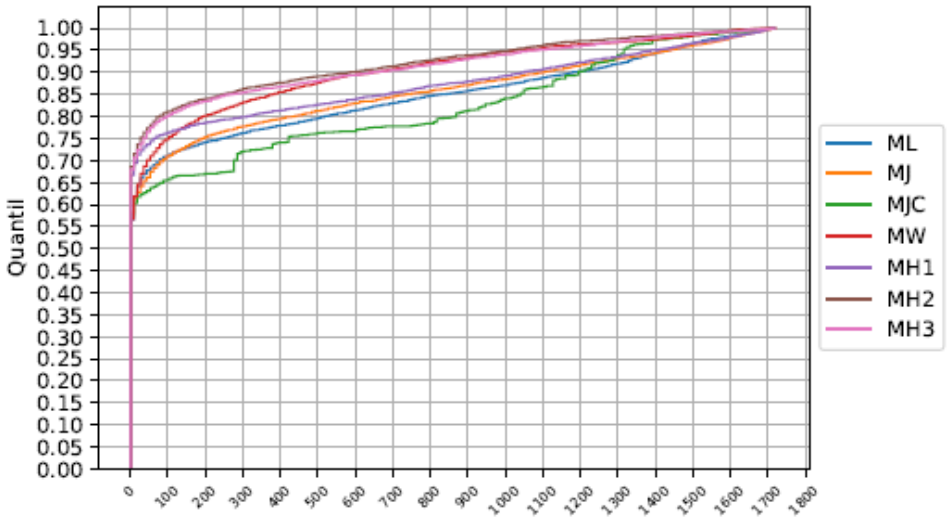

Figura 1. Rank do par correto produzido pela estratégia

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 dois diferentes níveis: léxico e semântico. 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.

Refinamentos metodológicos também poderiam ser utilizados para futuros trabalhos. Por exemplo, não foi empregada uma importante dimensão das estruturas textuais: a dimensão sintática [Sinoara et al. 2017]. Técnicas 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.

Referências

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

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

Francisco, R. E. e Ambrosio, A. P. (2016). Uso do algoritmo distância de edição com técnicas de pré-processamento para apoiar a identificação de plágio em códigos-fonte de problemas de programação introdutória. In *iSys*, 9(2), pages 32–52.

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.

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

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.

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.