UNIVERSIDADE ESTADUAL DE CAMPINAS - UNICAMP
SISTEMAS DE INFORMAÇÃO

ANÁLISE E IMPLEMENTAÇÃO DE ALGORITMOS DE COMPARAÇÃO POR SIMILARIDADE DE CONCEITOS EM MAPAS CONCEITUAIS

KAIQUE CHIOVETTO SIQUEIRA

LIMEIRA, 2019

KAIQUE CHIOVETTO SIQUEIRA

ANÁLISE E IMPLEMENTAÇÃO DE ALGORITMOS DE COMPARAÇÃO POR SIMILARIDADE DE CONCEITOS EM MAPAS CONCEITUAIS

Monografia apresentada à Faculdade de Tecnologia da Universidade Estadual de Campinas como parte dos requisitos para a obtenção do título de Bacharel em Sistemas de Informação.

Orientadora: Profa. Dra. Gisele Busichia Baioco

LIMEIRA, 2019

RESUMO

Mapas Conceituais são ferramentas gráficas que representam as relações entre conceitos. Eles são úteis para a aprendizagem, avaliação, organização e representação do conhecimento, auxiliando na retenção e recuperação de informações durante o processo de aprendizagem. Portanto, eles são usados com os mais diferentes propósitos. No entanto, há dificuldades em comparar Mapas Conceituais em relação à sintaxe e semântica dos conceitos utilizados. Este trabalho teve como objetivo implementar, analisar e qualificar algoritmos de comparação por similaridade de conceitos presentes em Mapas Conceituais. Por meio de um programa implementado em Java, foi possivel determinar que os algoritmos baseados na comparação por similariade semântica são mais eficazes em relação a acurácia dos resultados que os algoritmos baseados na comparação por similaridade sintática.

Palavras-chave: Mapas Conceituais, Comparação por Similaridade, Conceitos.

ABSTRACT

Concept Maps are graphical tools that represent the relationships between concepts. They are useful for learning, assessing, organizing and representing knowledge, helping to retain and retrieve information during the learning process. Therefore, they are used for many different purposes. However, there are difficulties in comparing Concept Maps in relation to the syntax and semantics of the concepts used. This work aimed to implement, analyze and qualify comparison algorithms for similarity of concepts present in Concept Maps. Through a program implemented in Java, it was possible to determine that algorithms based on semantic similarity comparison are more effective in relation to the accuracy of results than algorithms based on syntactic similarity comparison.

Keywords: Conceptual Maps, Similarity Comparison, Concepts.

SUMÁRIO

1 INTRODUÇÃO

Mapas Conceituais são instrumentos gráficos utilizados para organizar e representar o conhecimento. Eles contêm proposições que são expressões sobre algum objeto ou evento no universo. Cada proposição contém dois conceitos conectados por uma palavra de ligação de modo a compor uma afirmação com sentido (NOVAK; CANÃS, 2008).

Por se tratar de um mecanismo visual, facilita a compreensão de assuntos complexos e promove o conhecimento coletivo.

Não existe um Mapa Conceitual “correto”, mas sim um Mapa Conceitual para determinado conteúdo segundo os significados que uma pessoa atribui aos conceitos e às relações entre eles (MOREIRA, 2012). Sua estrutura é formada por uma questão central que identifica o tema do Mapa. Além disso é formado por palavras de ligação e conceitos. As palavras de ligação servem para descrever o relacionamento entre dois conceitos, formando uma frase coerente.

Os conceitos são rotulados por uma ou mais palavras que tem por finalidade conseguir descrever uma ideia que uma pessoa tem sobre um assunto. Como os conceitos são flexíveis, é difícil comparar dois conceitos de Mapas Conceituais diferentes a fim de saber se há equivalência de ideias.

Existem vários modos de descrever uma percepção de uma situação. Visto que cada autor de um Mapa Conceitual tem uma forma de representar seu conhecimento, a sintática e a semântica empregada nos conceitos passam a ser um fator importante para a comparação de mapas. Por exemplo, a proposição Cachorro – come – Alimentos é representada em um Mapa Conceitual, enquanto Cão – consome – Comidas é representada em outro. Ao analisar as duas proposições, verifica-se que todas as palavras são diferentes. Porém, são compostas por palavras sinônimas e constituem um mesmo contexto. Logo, são proposições diferentes, mas semanticamente podem ser consideradas iguais.

Como exemplificado anteriormente, uma análise por igualdade de conceitos seria incompleta, pois essa abordagem não considera o contexto das palavras. Assim, é válida a utilização de uma técnica de comparação por similaridade em conceitos, que leva em consideração o seu significado.

A comparação por similaridade é uma técnica que busca identificar o quão semelhante são dois objetos de diferentes fontes. Essa técnica, no escopo de Mapas

Conceituais, é uma opção para resolver os problemas sintáticos e semânticos da comparação entre os conceitos dos mesmos. Por meio da comparação por similaridade é possível definir o quão semelhante é um conceito em relação ao outro conceito comparado (BARIONI, 2006).

1.1 Justificativa

Mapa Conceitual é uma metodologia criada por NOVAK, que tem como objetivo facilitar que um indivíduo consiga representar e absorver conhecimento sobre determinado tema. Podem ser usados em diversas áreas, com diferentes intuitos, porém com o mesmo objetivo, a aprendizagem. Como cada Mapa Conceitual tem sua singularidade, existem diferentes formas de representar um conceito. A comparação entre conceitos representa um esforço para vencer a dificuldade de interpretação sintático-semântica. Às vezes, palavras grafadas iguais não representam o mesmo objeto ou sentimento, ou seja, embora sejam sintaticamente iguais, são semanticamente diferentes. Conseguir comparar dois conceitos e avaliar se eles são similares ajuda na comparação de Mapas Conceituais como um todo. Por exemplo, um gerente de projetos pede para que cada um dos seus funcionários faça um Mapa Conceitual com ideias para resolver um problema que surgiu durante a realização de um projeto. Ao final, o gerente vai comparar todos os Mapas Conceituais para ver quais conceitos são similares, a fim de chegar em uma única solução. Essa comparação é feita por algoritmos de comparação por similaridade.

Assim, é de grande importância facilitar a comparação entre conceitos dos Mapas Conceituais, de modo a aprimorar a absorção e entendimento de determinado assunto, expandido os níveis de conhecimento. A utilização de algoritmos de comparação por similaridade de conceitos entre Mapas Conceituais possibilita lidar com problemas gerados pelas linguagens. A comparação dos conceitos seria simples se não fossem pelos problemas sintáticos e semânticos.

1.2 Objetivos

O objetivo geral deste trabalho foi implementar, analisar e testar algoritmos para comparação por similaridade de conceitos presentes em Mapas Conceituais.

Os objetivos específicos foram:

- Investigar processos, métodos, técnicas e funções que explorem a comparação por similaridade de conceitos que compõem os Mapas Conceituais.

- Selecionar, implementar e testar os processos, métodos, técnicas e funções identificadas, considerando que os resultados influenciem diretamente na análise de Mapas Conceituais como um todo.

1.3 Metodologia

O trabalho seguiu uma abordagem quantitativa de caráter exploratório. Um enfoque especial foi dado aos conceitos que compõem os Mapas Conceituais. Assim, foi possível identificar e analisar algoritmos de comparação por similaridade e aplicá-los para determinar uma comparação efetiva de conceitos. Foram realizadas pesquisas para encontrar algoritmos de comparação por similaridade e posteriormente a implementação desses algoritmos encontrados. Em seguida, foram realizados testes nos algoritmos selecionados, utilizando modelos reais de Mapas Conceituais, de temas diversificados, para abranger os resultados, e assim, determinar sua efetividade. Alguns dos algoritmos analisados foram selecionados dependendo do seu tempo de execução, complexidade da implementação e principalmente a facilidade de adequação para tratar conceitos de Mapas Conceituais. Por último, os resultados obtidos foram devidamente documentados e sintetizados.

1.4 Organização do Trabalho

O próximo capítulo apresenta os materiais e os métodos utilizados para a realização desta monografia. Em seguida, o capítulo 3 mostra o desenvolvimento e a seleção dos algoritmos. No capítulo 4 expõe-se os resultados obtidos por meio da comparação por similaridade de conceitos. Por fim, o capítulo 5 relata a conclusão por meio da interpretação e análise dos resultados.

2 MATERIAIS E MÉTODOS

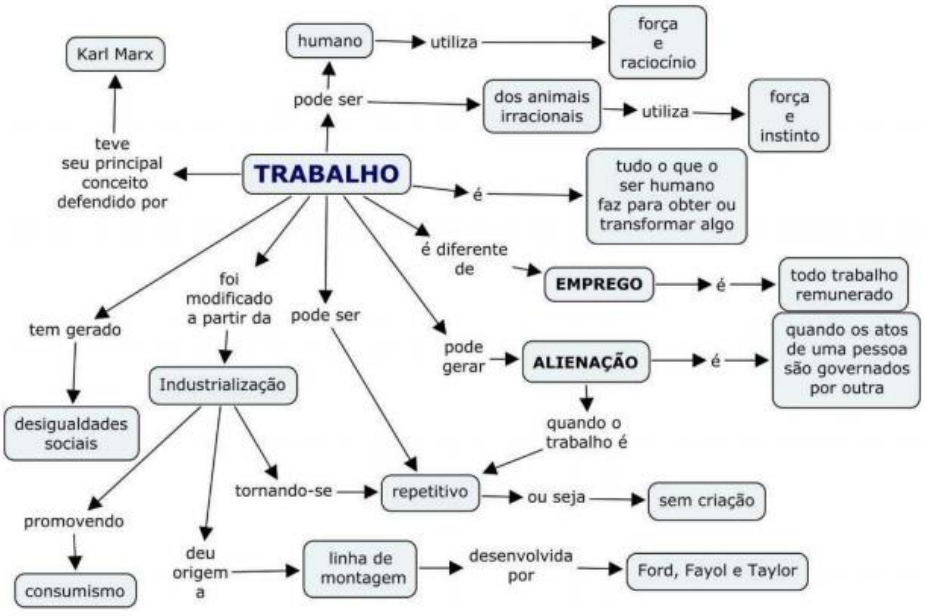
O objetivo deste trabalho consistiu em pesquisar, analisar, implementar e testar algoritmos de comparação por similaridade de conceitos presentes em Mapas Conceituais. Para isso foi importante compreender o que são Mapas Conceituais e conceitos. Desse modo, este capítulo esclarece e demonstra o que é um Mapa Conceitual. Em seguida, explica e exemplifica o que são conceitos. Depois versa sobre a comparação por similaridade entre conceitos que é a o foco da monografia.

2.1 Mapas Conceituais

Inicialmente foram realizadas pesquisas voltadas a entender o que é um Mapa Conceitual. Além de levantar informações sobre a sua definição, foram estudadas também as características dos componentes da sua estrutura e sua aplicação no mundo real, a fim de entender o seu funcionamento como um todo para então partir para os conceitos, que são o foco deste trabalho.

Mapas Conceituais são diagramas gráficos utilizados para organizar e representar o conhecimento por meio da relação entre os conceitos. Por meio deles, é possível a compreensão de temas complexos e o aprendizado colaborativo. A Figura 1 apresenta um exemplo de um Mapa Conceitual com o tema Relação do Homem com o Trabalho que ilustra o que são Mapas Conceituais.

Figura 1: Mapa Conceitual - Relação do Homem com o Trabalho



Fonte: http://fernandoscpimentel.blogspot.com/2010/06/ciencias-sociais.html

Como a Figura 1 mostra, um Mapa Conceitual apresenta conceitos que são relacionados por palavras de ligação, resultando em proposições. Por exemplo, a relação entre os conceitos “Trabalho” e “humano” por meio das palavras de ligação “pode ser” forma a proposição “Trabalho pode ser humano”. O conjunto formado pelas proposições formam um Mapa Conceitual.

2.2 Conceitos

Foi feito um levantamento teórico sobre Conceitos presentes em Mapas Conceituais, com o intuito de compreender a sua estrutura. Conseguir descrever as propriedades de um objeto ou construir um enunciado lógico sobre eventos ou situações que possuem atributos comuns caracteriza a função de um conceito em um Mapa Conceitual (PÉREZ; VIEIRA, 2005).

Conceito é um termo que é apresentado por uma ou mais palavras e representa uma ideia sobre eventos ou situações (NOVAK; CANÃS, 2008). Cada autor representa um conceito de acordo com a sua preferência e sua vivência, então não existe regras e padrões de como retratar essas ideias nos Mapas Conceituais.

A Tabela 1 apresenta os conceitos do Mapa Conceitual da Figura 1.

Tabela 1: Conceitos do Mapa Conceitual cujo tema é Relação do Homem com o Trabalho.

Conceitos
Trabalho
Emprego
todo trabalho remunerado
quando os atos de uma pessoa são governados por outra
linha de montagem
Taylor, Fayol e Ford
Industrialização
tudo o que o ser humano faz para obter ou transformar algo
dos animais irracionais
humano
força e raciocínio
força e instinto
repetitivo
Consumismo
Alienação

Continua

Continuação

Karl Marx
sem criação
desigualdades sociais

Fonte: Próprio Autor

2.3 Comparação por Similaridade de Conceitos

Foram pesquisados materiais sobre comparação por similaridade e como ela funciona. Dado que os conceitos podem ser considerados frases curtas, houve a possibilidade de utilizar algoritmos de comparação por similaridade que servem para medir a semelhança entre frases. Com isso, foi possível perceber a utilidade da comparação por similaridade para os conceitos. Também foi notado que, como os conceitos são formados por palavras, haveria problemas sintáticos e semânticos. Então, foram levantados algoritmos de comparação por similaridade sintática e similaridade semântica.

2.3.1 Algoritmos de Comparação por Similaridade Sintática

Após identificar a possibilidade de utilizar no projeto, algoritmos de comparação por similaridade que calculam a semelhança entre frases, foram selecionados aqueles que de fato possam ser utilizados na comparação de conceitos. A princípio, foram levantados algoritmos de comparação por similaridade sintática. Por lidarem com a sintaxe das palavras, esses algoritmos comparam a maneira que os conceitos foram escritos, ou seja, a relação lógica entre eles. Entretanto, nem todos foram usados. Alguns apresentam condições muito específicas o que os tornam inviáveis para a aplicação neste trabalho, cujo foco são os conceitos.

A seguir estão listados os algoritmos encontrados seguidos de suas definições e características.

LCS - *Longest Common Substring* (Maior *Substring* em Comum**)**

Este algoritmo procura semelhanças entre strings[1] por meio da busca da maior *substring* contida entre as *strings* comparadas. Uma *substring* comum a duas *strings* é uma *substring* das suas *strings* simultaneamente (XAVIER; BATISTA, 2018). Por exemplo, ao comparar as *strings* “cinto” e “sinto” tem como resultado a *substring* “into” (4 caracteres).

Levenshtein *Edit Distance*

[1]*String*: cadeia de caracteres normalmente utilizada para retratar palavras e/ou frases.

O algoritmo compara duas *strings* por meio do número mínimo de operações para transformar uma *string* na outra. As operações usadas por esse algoritmo são: inserção, remoção e substituição. Um resultado igual a 0 quer dizer que as duas *strings* comparadas são iguais (FREEMAN; CONDON; ACKERMAN, 2006).

Jaro Winkler

Este algoritmo é uma extensão da "*Jaro Distance Metric*" que é um método que compara os caracteres comuns existentes entre duas *strings* desde que os caracteres de B sejam encontrados na mesma ordem que os de A. Utiliza das mesmas métricas da distância de Jaro e a diferença entre eles é que a Jaro Winkler tem uma bonificação em seu resultado porque valoriza as *strings* que tem prefixo em comum (SILVA, 2007). A métrica da comparação é 0 quando as *strings* comparadas forem totalmente diferentes e 1 quando as *strings* forem iguais.

Algoritmo de Hamming

O algoritmo de Hamming mede o número de substituições necessárias para transformar uma *string* na outra. Um requisito desse algoritmo é que as *strings* comparadas têm que possuir o mesmo tamanho (NOROUZI; FLEET; SALAKHUTDINOV, 2012). Se algum dos conceitos comparados for de maior que o outro, o conceito com maior número de caracteres é cortado para que os dois conceitos fiquem com o mesmo número de caracteres.

Damerau-Levenshtein *Distance*

A distância de Damerau-Levenshtein é uma extensão da distância de Levenshtein. Ela também é definida como o número mínimo de operações de edição simples na sequência para alterara uma *string* em outra, mas a lista de operações permitidas é estendida, além das 3 operações (inserção, exclusão e substituição) ela também contém a operação de transposição de dois caracteres adjacentes (MIRA; FEIJÃO; MEIDANIS; DUQUEESTRADA; JOLY, 2013).

2.3.2 Algoritmos de Comparação por Similaridade Semântica

Neste ponto, com os algoritmos de comparação por similaridade sintática encontrados, iniciou-se a procura na literatura por algoritmos de comparação por similaridade que tratam a semântica das palavras. Foram encontradas a similaridade de

Jaccard, a similaridade do Cosseno e uma adaptação da similaridade de Jaccard que aborda sinônimos.

A similaridade de Jaccard é uma métrica baseada no número de elementos em comum entre dois conjuntos de palavras (X e Y). A similaridade é dada pela equação (1) e varia entre [0,1], sendo 0 total dissimilaridade e 1 quando os dois conceitos são iguais (PONTES, 2015).

$$Jaccard(X,Y) = \frac{|X \cap Y|}{|X \cup Y|} \quad (1)$$

A similaridade do Cosseno consiste no cálculo do ângulo do cosseno formado entre dois vetores de palavras (X e Y). Cada palavra representa uma posição no vetor. Essa métrica é dada pela equação (2) e varia entre [0,1], sendo 1 quando os conceitos são iguais (PONTES, 2015).

$$Cosseno(X,Y) = \frac{\sum Xn \times Yn}{\sqrt{\sum X * \sum Y}} \quad (2)$$

A similaridade de Jaccard adaptada é uma extensão da similaridade de Jaccard que além de considerar o número de palavras iguais entre os conceitos, também considera o número de sinônimos presentes na comparação. A similaridade é apresentada pela equação (3) e varia entre [0,1], sendo 0 quando não há igualdade e 1 quando os conceitos comparados são equivalentes (MOREIRA; HAYASHI; COELHO; SILVA, 2015).

$$JaccardAdaptado = \frac{QtdIG + QtdSIN}{QtdT} \quad (3)$$

Sendo:

QtdIG o número de palavras iguais entre os conceitos comparados.

QtdSIN o número de sinônimos entre os conceitos comparados.

QtdT o número total de palavras dos conceitos comparados.

3 DESENVOLVIMENTO

Este capítulo apresenta a análise e a seleção dos algoritmos encontrados na literatura que mais se adequam à comparação de conceitos. Em seguida, será mostrada a implementação dos algoritmos em fluxogramas com exemplos práticos de seus funcionamentos.

3.1 Análise e seleção dos Algoritmos

Para analisar os algoritmos encontrados na literatura, a etapa de teste foi dividida em duas partes. Na primeira parte, foram testados os algoritmos de comparação por similaridade sintática, a fim de verificar como os algoritmos selecionados se comportam na comparação de conceitos de Mapas Conceituais. Os algoritmos LCS, Levenshtein, Hamming e Damerau foram normalizados para que os resultados das comparações entre os conceitos A e B retornassem valores entre o intervalor [0,1], onde 1 significa total similaridade e 0 total dissimilaridade. A Tabela 2 mostra o resultado dos primeiros testes com os algoritmos.

Tabela 2: Resultado da comparação por similaridade entre dois conceitos

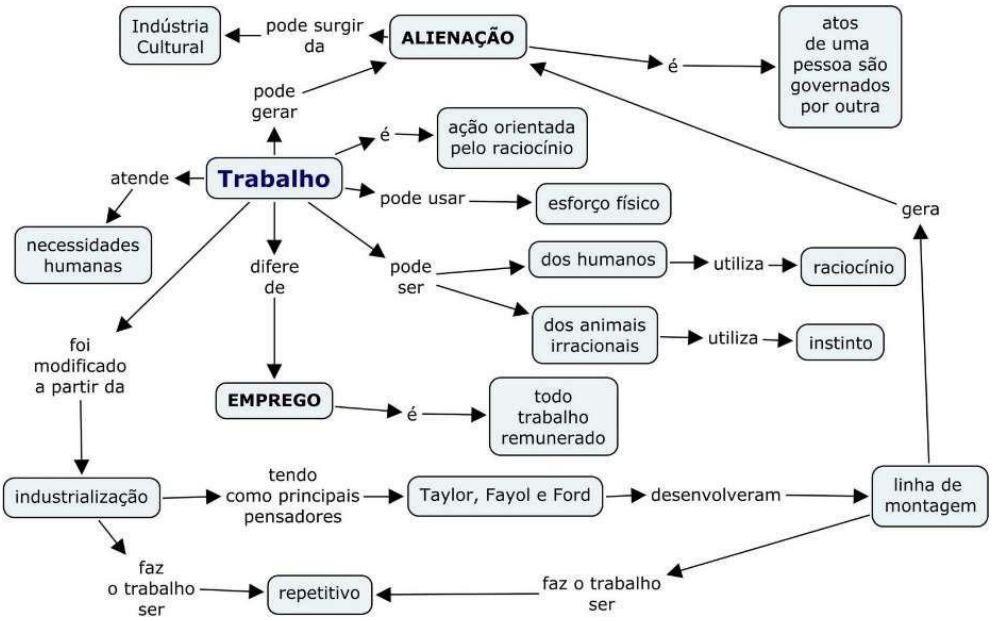
Conceito A	Conceito B	LCS	Levenshtein	Jaro Winkler	Hamming	Damerau
“Manga”	“manga”	1,00	1,00	1,00	1,00	1,00
“Lar”	“Casa”	0,25	0,25	0,53	Falha	0,25
“Sorte”	“Corte”	0,80	0,80	0,87	0,80	0,80
“Facebook”	“Instagram”	0,11	0	0,41	Falha	0
“Celular”	“Telefone”	0,25	0,25	0,51	Falha	0,25

Fonte: Próprio Autor

Pelos resultados obtidos (Tabela 2), pôde-se perceber que o algoritmo de Hamming não é adequado já que os conceitos comparados teriam que ter o mesmo número de caracteres limitando a maneira de expressar o conhecimento e assim falhando até mesmo com a própria definição de conceitos em Mapas Conceituais.

Com os algoritmos selecionados, foi feita uma formalização de um teste usando conceitos de dois Mapas Conceituais diferentes, apresentados nas Figura 1 (exposta na seção 2.1) e a Figura 2, a serem comparados por similaridade.

Figura 2: Mapa Conceitual – Relação do Homem com o Trabalho



Fonte: http://fernandoscpimentel.blogspot.com/2010/06/ciencias-sociais.html

Com os conceitos separados, pôde-se realizar a comparação cujo resultado é apresentado na Tabela 3.

Tabela 3: Resultado da comparação dos conceitos das figuras 1 e 2

Conceitos da Figura 1	Conceitos da Figura 2	LCS	Levenshtein	Jaro Winkler	Damerau
Trabalho	Trabalho	1,00	1,00	1,00	1,00
Emprego	Emprego	1,00	1,00	1,00	1,00
Todo trabalho remunerado	todo trabalho remunerado	1,00	1,00	1,00	1,00
Quando os atos de uma pessoa são governados por outra	atos de uma pessoa são governados por outra	0,81	0,81	0,79	0,81
linha de montagem	linha de montagem	1,00	1,00	1,00	1,00
Taylor, Fayol e Ford	Taylor, Fayol e Ford	1,00	1,00	1,00	1,00

Continua

Continuação

Industrialização	Industrialização	1,00	1,00	1,00	1,00
Tudo o que o ser humano faz para obter ou transformar algo	ação orientada pelo raciocínio	0,05	0,26	0,51	0,26
dos animais irracionais	dos animais irracionais	1,00	1,00	1,00	1,00
humano	dos humanos	0,55	0,55	0,68	0,55
força e raciocínio	Raciocínio	0,55	0,55	0,59	0,55
força e instinto	Instinto	0,5	0,5	0,53	0,5
repetitivo	Repetitivo	1,00	1,00	1,00	1,00
consumismo	necessidades humanas	0,25	0,20	0,45	0,20
Alienação	Alienação	1,00	1,00	1,00	1,00

Fonte: Próprio Autor

Com esse teste foi possível concluir que o algoritmo Jaro Winkler apresenta valores falso-positivos (linha 8), fazendo com que ele não seja adequado para o escopo deste projeto. Também foi possível compreender que, pelo algoritmo Damerau ser uma extensão do Levenshtein, eles apresentam valores iguais ou muito parecidos acarretando numa comparação análoga o que o torna irrelevante para esta pesquisa. Portanto, no caso da similaridade sintática, foram selecionados os algoritmos LCS e Levenshtein. No caso da similaridade semântica, foram considerados todos os algoritmos estudados, pois eram apenas três.

3.2 Implementação dos Algoritmos de Comparação por Similaridade

A Figura 3 apresenta o fluxograma que contém a entrada dos conceitos, o pré-processamento feito e a apresentação dos resultados.

Para que os conceitos fossem manipuláveis pelos algoritmos, houve a necessidade de pré-processamento dos mesmos. Porém, houve necessidades diferentes

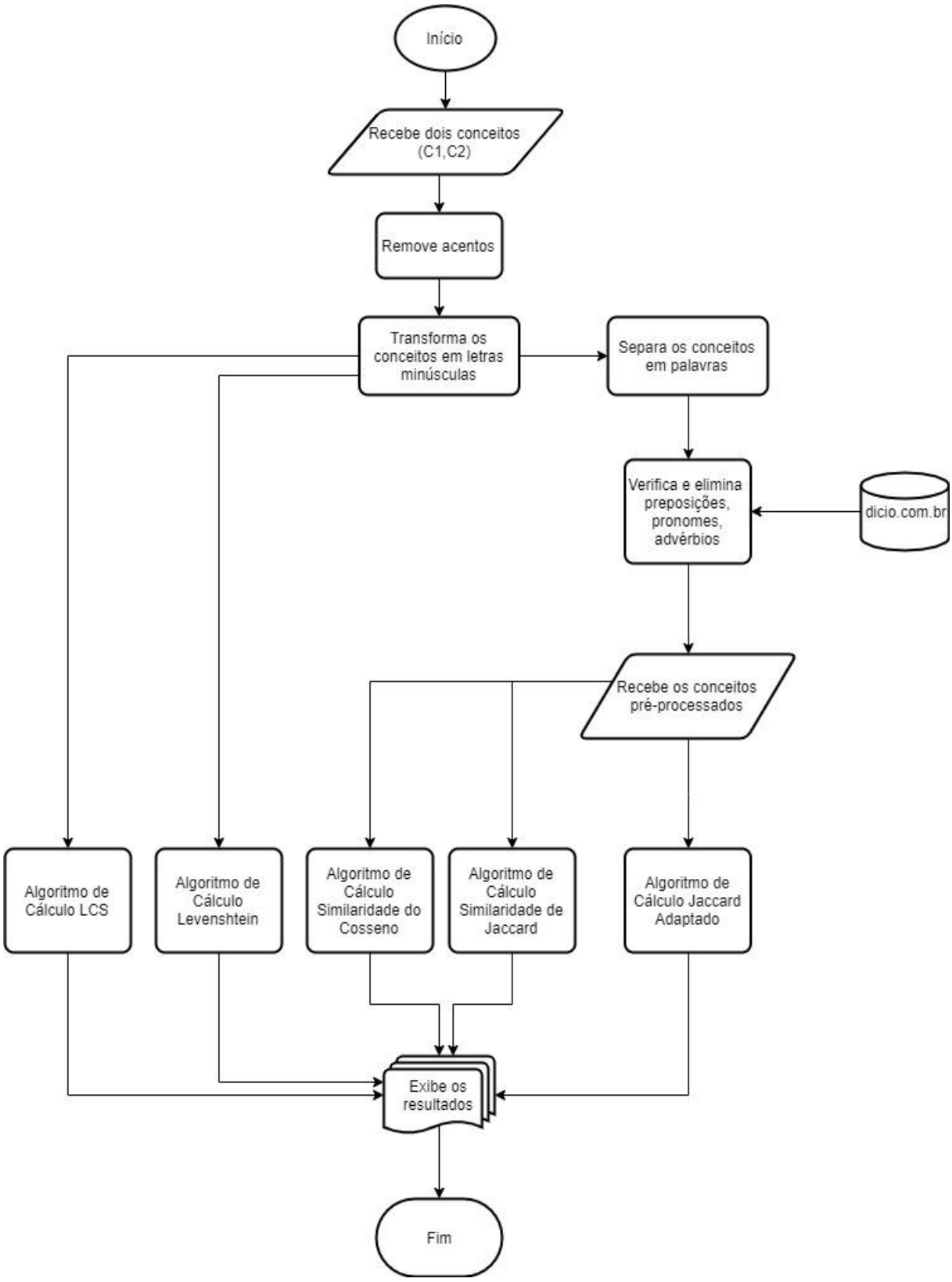
para os algoritmos sintáticos (LCS, Levenshtein) e os algoritmos semânticos (Cosseno, Jaccard, Jaccard Adaptado). As seguintes etapas de pré-processamento foram implementadas neste trabalho:

1- Remoção dos acentos.
2- Transformar os conceitos em letras minúsculas, visto que na análise humana isso não faz diferença.
3- Segmentação dos conceitos para identificar as palavras que os compõem.
4- Remoção de artigos, pronomes e advérbios que sozinhos não tem relevância para a comparação (REZENDE; MARCACINI; MOURA, 2011).

Para os algoritmos sintáticos devem ser realizadas as etapas 1 e 2 de pré-processamento. Para os algoritmos semânticos devem ser realizadas todas as etapas de pré-processamento.

Para realizar a etapa 4 do pré-processamento para os algoritmos semânticos, houve a necessidade de implementar uma função que faz uma verificação em uma base online "https://www.dicio.com.br/" no momento em que o programa verifica e elimina preposições, pronomes e advérbios. A base online utilizada, é um dicionário online da Língua Portuguesa, em que os usuários digitam palavras que querem saber os significados. Por se tratar de um dicionário, além de retornar o significado da palavra pesquisada, também retorna a estrutura morfológica da mesma. Também possui outras funcionalidades, como por exemplo, frases e exemplos com a palavra procurada. No caso do programa implementado, são enviadas as palavras dos conceitos para o site e como resposta, são retornadas as estruturas morfológicas delas. Por exemplo, ao pesquisar a palavra "Estudante" no site, tem como retorno seu significado e sua estrutura morfológica, no caso substantivo ou adjetivo.

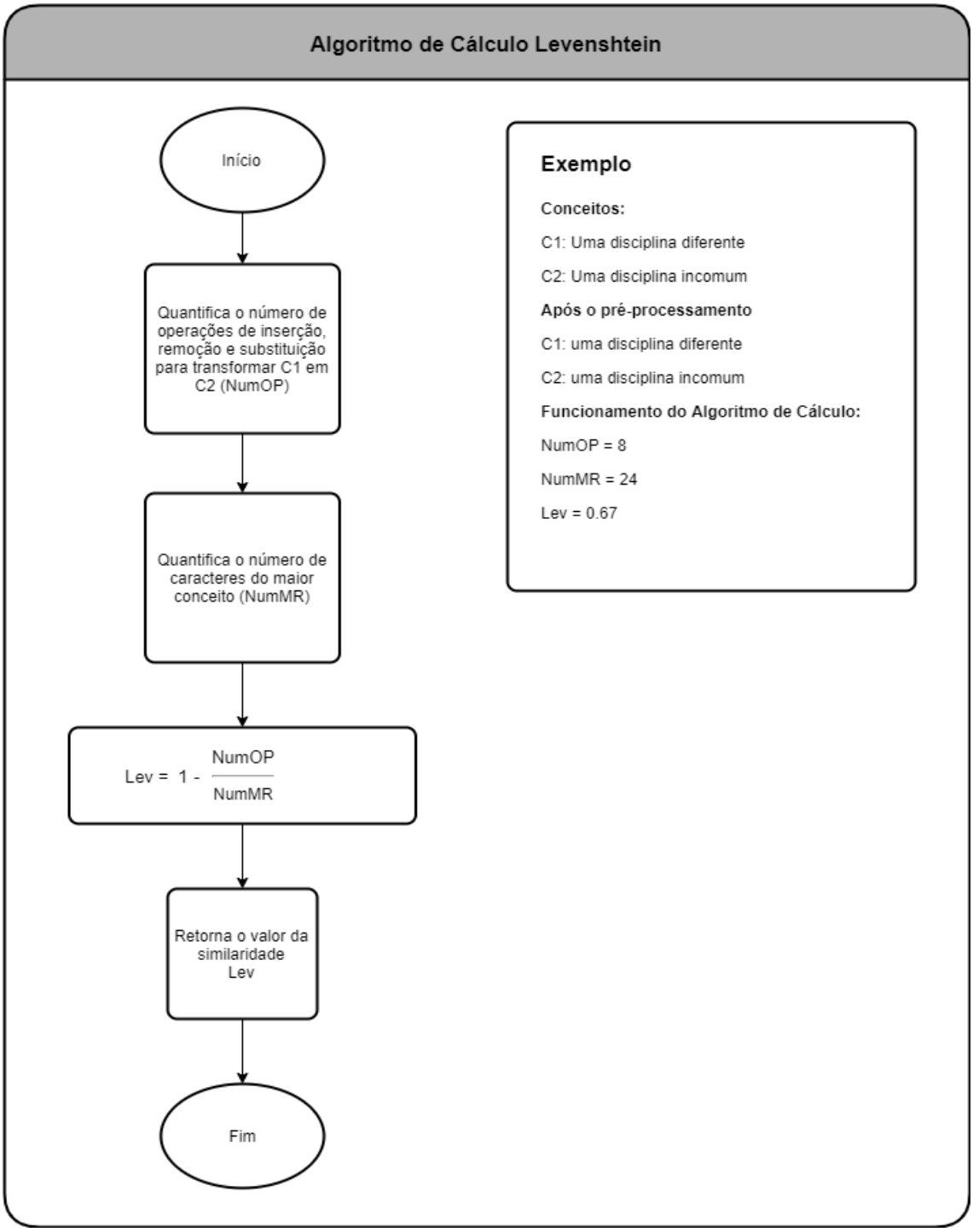
Figura 3: Fluxograma sobre o funcionamento do programa



Fonte: Próprio Autor

As Figuras 4,5,6,7,8 apresentam os fluxogramas que compreendem o funcionamento dos Algoritmos de Similaridade usados para calcular a similaridade dos conceitos. Cada fluxograma apresenta um exemplo da comparação dos conceitos “Uma disciplina diferente” e “Uma disciplina incomum” ao serem comparados. A Figura 4 apresenta o Algoritmo de Cálculo Levenshtein.

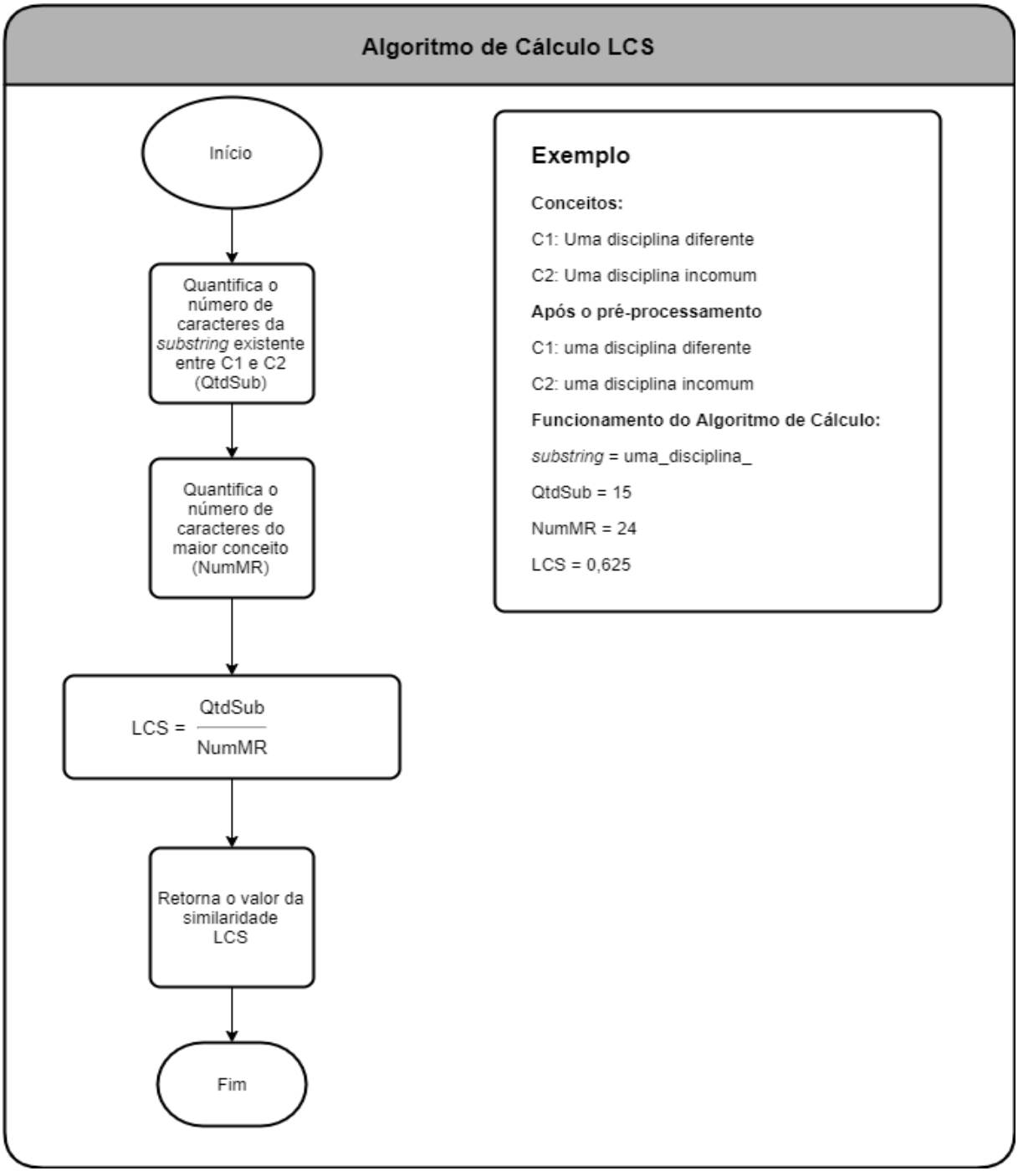
Figura 4: Fluxograma sobre o funcionamento Algoritmo de Cálculo Levenshtein



Fonte: Próprio Autor

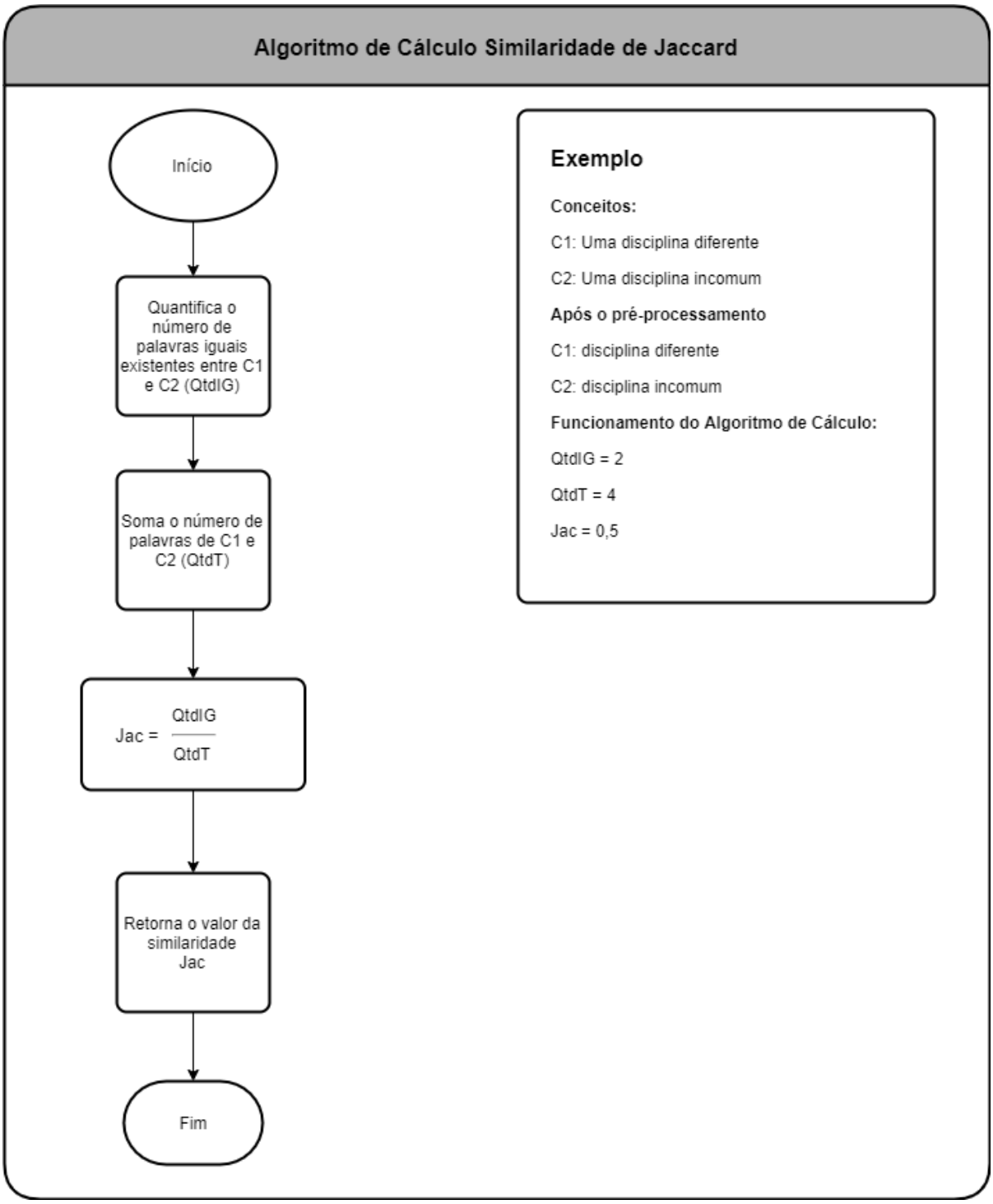
No Algoritmo LCS exemplificado pela Figura 5, é possível perceber que os espaços entre uma palavra e outra dos conceitos, também contam como caracteres, fazendo com que as *substrings* tenham um número maior de caracteres do que o número de caracteres real dos conceitos.

Figura 5: Fluxograma sobre o funcionamento Algoritmo de Cálculo LCS



Fonte: Próprio Autor

Figura 6: Fluxograma sobre o funcionamento Algoritmo de Cálculo Similaridade de Jaccard

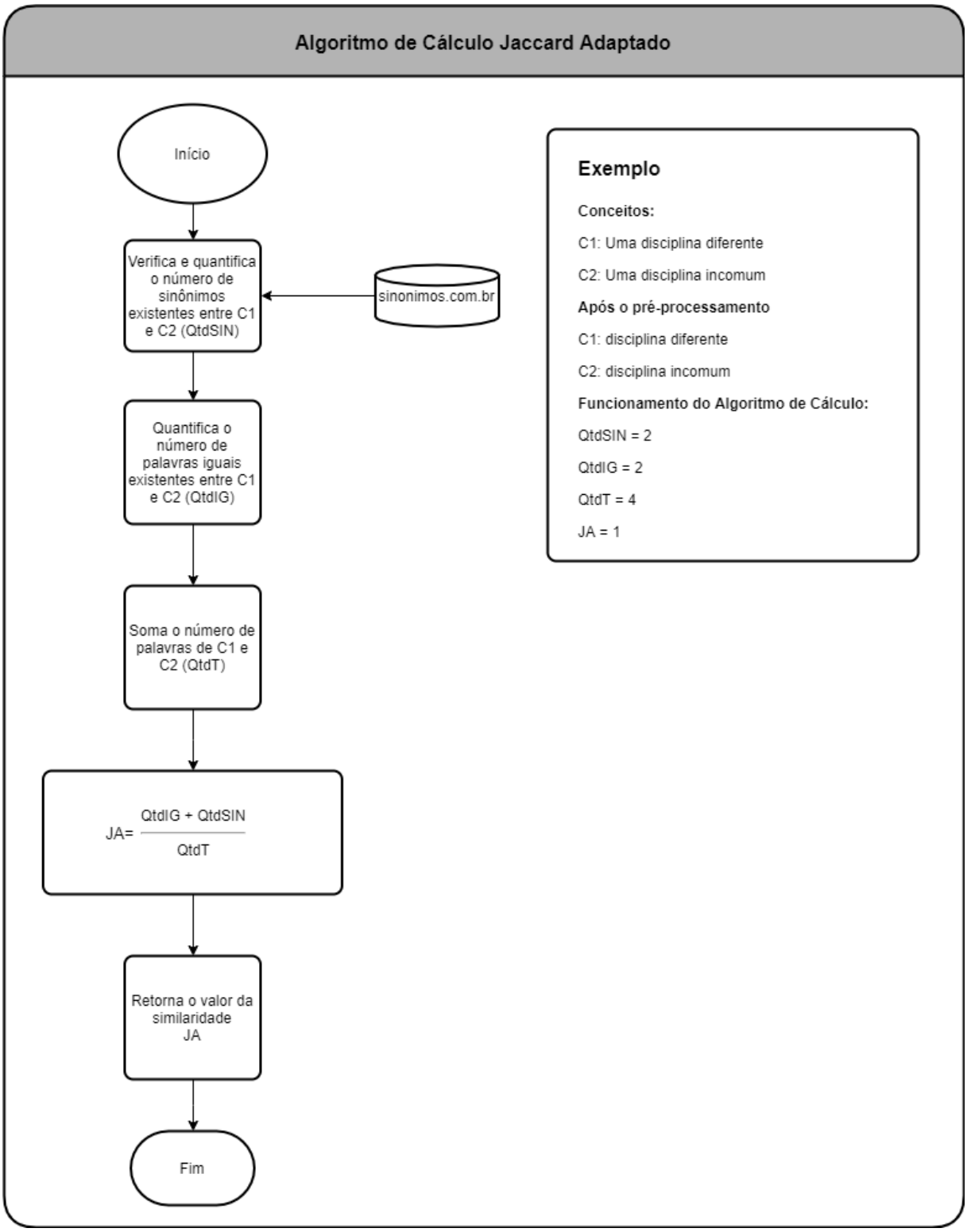


Fonte: Próprio Autor

A Figura 7 apresenta o fluxograma sobre o funcionamento do Algoritmo de Cálculo de Jaccard Adaptado. Nesse algoritmo há necessidade de utilizar uma base para a verificação de sinônimos presentes na comparação. No caso do programa, foi utilizada uma base de sinônimos online “https://www.sinonimos.com.br/”. O programa

envia todas as palavras dos conceitos para o site, e como retorno obtém todos os possíveis sinônimos da palavra procurada. Em seguida, faz a verificação da existência de sinônimos entre os conceitos comparados.

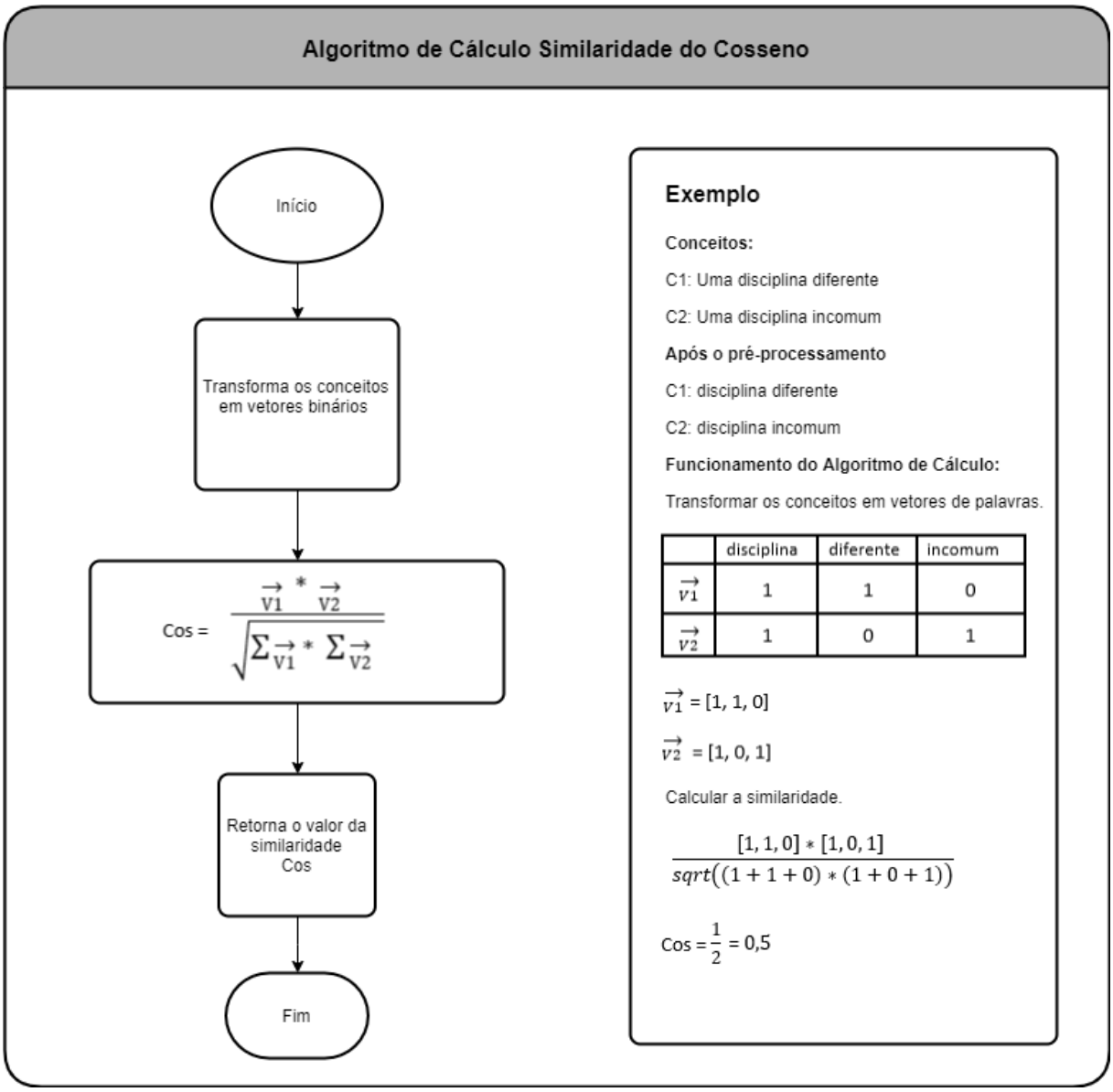
Figura 7: Fluxograma sobre o funcionamento Algoritmo de Cálculo Jaccard Adaptado



Fonte: Próprio Autor

A Figura 8 apresenta o fluxograma sobre o funcionamento do Algoritmo de Cálculo Similaridade do Cosseno. Nesse algoritmo os conceitos são transformados em vetores binários para que seja possível calcular a similaridade.

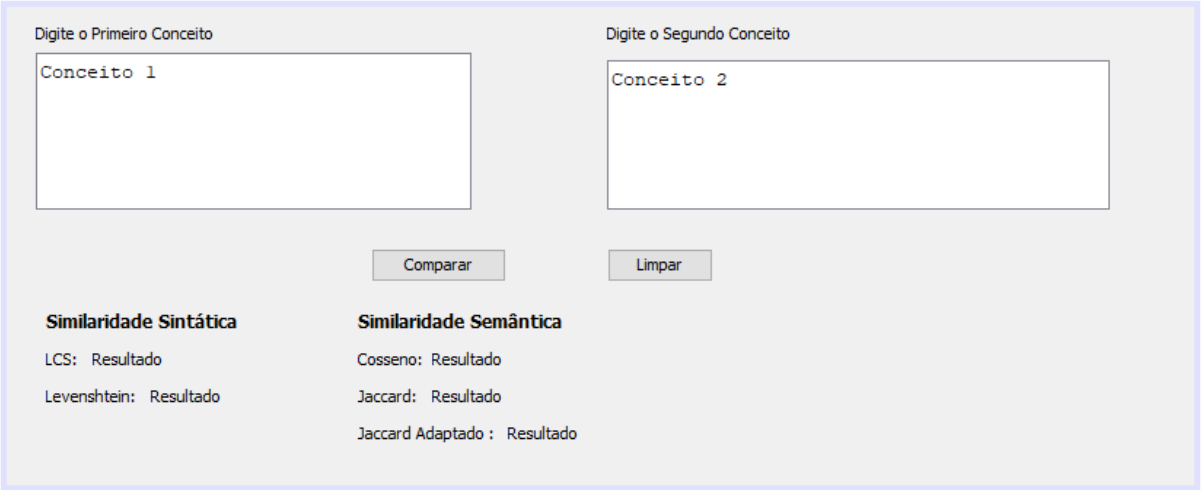
Figura 8: Fluxograma sobre o funcionamento Algoritmo de Cálculo Similaridade do Cosseno



Fonte: Próprio Autor

Para a execução dos testes com os algoritmos de comparação por similaridade selecionados foi feito um programa com interface gráfica com a finalidade de receber os dois conceitos que serão comparados e após a execução, exibir o resultado. O desenvolvimento do programa foi feito na linguagem Java por meio da IDE Net Beans 8.2. A Figura 9 apresenta a Tela Principal do programa desenvolvido.

Figura 9: Tela Principal para execução dos testes



Fonte: Próprio Autor

4 RESULTADOS

Para os testes dos algoritmos foram usados conceitos coletados da base de dados do grupo GEICon – Grupo de Engenharia da Informação e Conhecimento da Faculdade de Tecnologia da UNICAMP. A Tabela 4 mostra os resultados dos testes, sendo LCS = Algoritmo de Cálculo *Longest Common Substring,* Lev = Algoritmo de Cálculo Levenshtein, Cos = Similaridade do Cosseno, Jac = Similaridade de Jaccard, JA = Algoritmo de Cálculo Jaccard Adaptado.

Tabela 4: Resultado da comparação dos conceitos do Grupo GEICon

Conceito 1	**Conceito 2**	**LCS**	**Lev**	**Cos**	**Jac**	**JA**
experiências	Com experiências vividas até o momento	0,31	0,31	0,71	0,66	0,66
experiências	Experiência vivida por amigo e familiares	0,27	0,29	0	0	0
experiências	Boa experiência	0,73	0,67	0	0	0
experiências	Experiência de aprendizado	0,42	0,42	0	0	0
experiências	Experiência real de administração do tempo do aluno	0,21	0,23	0	0	0
experiências	Deveras a experiência de entrar na Universidade	0,23	0,25	0	0	0
experiências	Experiências	1	1	1	1	1
Com experiências vividas até o momento	Experiência vivida por amigo e familiares	0,27	0,43	0	0	0
Com experiências vividas até o momento	Boa experiência	0,31	0,34	0	0	0
Com experiências vividas até o momento	Experiência de aprendizado	0,29	0,42	0	0	0
Com experiências vividas até o momento	Experiência real de administração do tempo do aluno	0,21	0,35	0	0	0,22
Com experiências vividas até o momento	Forma de ansiedade pelas novas experiências	0,30	0,25	0,31	0,28	0,28
Experiência vivida por amigo e familiares	Boa experiência	0,27	0,19	0,32	0,28	0,28
Experiência vivida por amigo e familiares	Experiência de aprendizado	0,29	0,44	0,31	0,28	0,28

Continua

Continuação

Conceito 1	Conceito 2	LCS	Lev	Cos	Jac	JA
Experiência vivida por amigo e familiares	Experiência real de administração do tempo do aluno	0,23	0,39	0,15	0,17	0,17
Experiência vivida por amigo e familiares	Novas experiências	0,27	0,22	0	0	0
Experiência vivida por amigo e familiares	Deveras a experiência de entrar na Universidade	0,25	0,30	0,26	0,25	0,25
Experiência vivida por amigo e familiares	Forma de ansiedade pelas novas experiências	0,25	0,21	0	0	0
Experiência vivida por amigo e familiares	Experiências	0,27	0,29	0	0	0
Boa experiência	Experiência de aprendizado	0,42	0,27	0,5	0,5	0,5
Boa experiência	Experiência real de administração do tempo do aluno	0,21	0,14	0,23	0,22	0,22
Boa experiência	Novas experiências	0,67	0,78	0	0	0
Boa experiência	Forma de ansiedade pelas novas experiências	0,28	0,32	0	0	0
Boa experiência	Experiências	0,73	0,67	0	0	0
Experiência de aprendizado	Experiência real de administração do tempo do aluno	0,23	0,41	0,23	0,22	0,22
Experiência de aprendizado	Novas experiências	0,42	0,27	0	0	0
Experiência de aprendizado	Deveras a experiência de entrar na Universidade	0,32	0,42	0,41	0,40	0,40
Experiência de aprendizado	Forma de ansiedade pelas novas experiências	0,25	0,28	0	0	0
Experiência de aprendizado	Experiências	0,42	0,42	0	0	0
Experiência real de administração do tempo do aluno	Novas experiências	0,21	0,14	0	0	0
Experiência real de administração do tempo do aluno	Deveras a experiência de entrar na Universidade	0,23	0,23	0,19	0,20	0,20
Experiência real de administração do tempo do aluno	Forma de ansiedade pelas novas experiências	0,21	0,21	0	0	0
Experiência real de administração do tempo do aluno	Experiências	0,21	0,23	0	0	0
Novas experiências	Forma de ansiedade pelas novas experiências	0,42	0,42	0,63	0,57	0,57

Continua

Continuação

Conceito 1	Conceito 2	LCS	Lev	Cos	Jac	JA
Deveras a experiência de entrar na Universidade	Forma de ansiedade pelas novas experiências	0,25	0,25	0	0	0
Deveras a experiência de entrar na Universidade	Experiências	0,23	0,25	0	0	0
Forma de ansiedade pelas novas experiências	Experiências	0,28	0,28	0,45	0,33	0,33
além de saber a matéria	Matérias mais recorrentes no vestibular	0,18	0,26	0	0	0
além de saber a matéria	Com várias matérias	0.35	0.47	0.47	0	0
Bastante para a prova	Provas anteriores da Unicamp	0,18	0,39	0	0	0
Bastante para a prova	Medo de reprovar a disciplina	0,17	0,27	0	0	0
Bastante para a prova	Teorias dos assuntos da prova	0,24	0,34	0,45	0,33	0,33
Bastante para a prova	Lidar com o tempo de prova	0,23	0,35	0,58	0,5	0,5
Provas anteriores da Unicamp	Medo de reprovar a disciplina	0,17	0,17	0	0	0
Provas anteriores da Unicamp	Teorias dos assuntos da prova	0,17	0,27	0,22	0,22	0,22
Provas anteriores da Unicamp	Lidar com o tempo de prova	0,18	0,21	0	0	0
Medo de reprovar a disciplina	Teorias dos assuntos da prova	0,17	0,14	0	0	0
Teorias dos assuntos da prova	Lidar com o tempo de prova	0,21	0,45	0,26	0,25	0,25
Mudanças	Mudanças	1	1	1	1	1
antes de experimentar	Fase experimental	0,62	0,67	0	0	0
com situações que já conhece	Com situações que já conhece	1	1	1	1	1
Aluno	Aluno	1	1	1	1	1
Aluno	Aluno passar no vestibular	0,19	0,19	0,58	0,5	0,5
Aluno	Estudante	0,11	0,22	0	0	1
Aluno passar no vestibular	Se depender do aluno	0,19	0,23	0,33	0,33	0,33
Grupo	Turma	0,2	0	0	0	1
Disciplinas	Disciplina viva	0,66	0,66	0	0	0
Disciplinas	Disciplina diferente	0,5	0,5	0	0	0
Disciplinas	Métodos diferentes	0,11	0,27	0	0	0,66
Disciplinas	Disciplina com metodologia incomum	0,29	0,29	0	0	0
Disciplina sobre administração empresarial	Disciplina com metodologia incomum	0,26	0,38	0,33	0.33	0.33

Continua

Continuação

Conceito 1	Conceito 2	LCS	Lev	Cos	Jac	JA
Disciplina viva	Disciplina diferente	0,55	0,6	0,5	0,55	0,5
Disciplina diferente	Disciplina diferente	1	1	1	1	1
Grupos	Mesmo Grupo	0,45	0,36	0	0	0
Grupo	Mesmo Grupo	0,45	0,45	0,45	1	1
Interação com o grupo	Grupo	0,24	0,24	0,71	0,66	0,66

Fonte: Próprio Autor

Os resultados apresentados na Tabela 4 comprovam que os algoritmos usados para a similaridade semântica apresentam melhores resultados em relação a acurácia da comparação. Por exemplo, a comparação por similaridade entre os conceitos “Boa experiência” e “Experiências” retornou para os algoritmos de Jaccard, Cosseno e Jaccard Adaptado o valor 0 (zero). Em contrapartida, para os algoritmos LCS e Levenshtein a comparação retornou valores 0,73 e 0,67, respectivamente. Esses resultados se devem porque os dois conceitos apresentam a palavra “experiência”, fazendo com que os algoritmos de similaridade sintática considerem que os conceitos comparados são similares, quando na verdade eles não apresentam nenhuma relação de equivalência. Outro caso é quando os conceitos “experiências” e “Com experiências vividas até o momento” são comparados. Os algoritmos usados para a comparação por similaridade semântica retornaram os valores 0,71(Cosseno) e 0,66(Jaccard, Jaccard Adaptado), enquanto os usados para a comparação por similaridade sintática o valor 0,31. Apesar dos conceitos não serem iguais em relação a sintaxe, considerando uma análise humana eles podem ser usados para dizer a mesma coisa, provando que se acrescentar algumas palavras, os algoritmos sintáticos não são eficazes em relação a comparações dos conceitos. A adaptação de Jaccard é válida quando relacionamos conceitos que possuem sinônimos, como no caso da comparação dos conceitos “Aluno” e “Estudante”, em que o algoritmo retornou 1 e os demais, valores baixos para a comparação. Apesar dos algoritmos de Jaccard, Jaccard Adaptado e do Cosseno apresentarem melhores resultados no que se refere a comparação dos conceitos, os mesmos apresentaram maiores tempos de execução, pela necessidade de ter mais pré-processamento e a geração de vetores binários (Cosseno) para serem manipuláveis pelos algoritmos. Quanto a complexidade de implementação dos códigos, os algoritmos LCS e Levenshtein são mais fáceis de implementar porque só trabalham com *substrings* e técnicas de adição, remoção e edição de caracteres.

5 CONCLUSÕES

A comparação por similaridade é uma possível solução para identificar conceitos equivalentes em diferentes Mapas Conceituais, o que contribui para a comparação de Mapas Conceituais como um todo.

A compreensão da definição de conceito e de como ele é formado ajuda na pesquisa de técnicas e métodos adequados para a comparação por similaridade entre conceitos.

Os resultados encontrados comprovam que algoritmos de similaridade baseados na sintaxe dos conceitos são menos precisos que os algoritmos que realizam a similaridade semântica.

Os algoritmos de similaridade baseados na semântica dos conceitos são eficazes, porém menos eficientes que os algoritmos que abrangem a comparação sintática, já que são necessárias mais etapas de pré-processamento dos conceitos antes de calcular o valor da similaridade, elevando o tempo de processamento.

Entretanto, existem medidas e recursos que podem ser tomados para diminuir esse tempo de processamento. Por exemplo, o pré-processamento (limpeza e a adequação dos conceitos para que eles sejam capazes de serem manipulados pelos algoritmos) poderia ser feito de maneira separada ao programa que controla o algoritmo de cálculo, minimizando assim o tempo de execução da comparação.

Uma dificuldade encontrada foi que existem muitos trabalhos na literatura de comparação por similaridade entre palavras em inglês, que utilizam bancos de dados léxicos em inglês. Entretanto, não existem muitas técnicas para a comparação de palavras por similaridade em português.

De modo geral, a acurácia da comparação por similaridade dos conceitos em Mapas Conceituais depende da sintaxe e da semântica dos conceitos. Entretanto, resultados mais precisos aumentam o tempo de execução e a complexidade da implementação do algoritmo. Assim, este trabalho teve como finalidade auxiliar futuros trabalhos que necessitem da comparação de Mapas Conceituais na escolha do algoritmo mais adequado para comparar conceitos de acordo com os requisitos de cada aplicação. Para trabalhos futuros, existem outros problemas semânticos que podem ser tratados a fim de melhorar a acurácia da comparação. Por exemplo, existem problemas semânticos em relação a homônimos, parônimos e a polissemia das palavras.

6 REFERÊNCIAS BIBLIOGRÁFICAS

BARIONI, M. C. N. **Operações de consulta por similaridade em grandes bases de dados complexos**. Tese, São Carlos: Universidade de São Paulo. Instituto de Ciências Matemáticas e de Computação, 2006.

FREEMAN, Andrew; CONDON, Sherri; ACKERMAN, Christopher. Cross **Linguistic Name Matching in English and Arabic: a "one to many mapping" extension of the Levenshtein edit distance algorithm**. 2006. Proceedings of HLT.

MIRA, Cleber; FEIJÃO, Pedro; MEIDANIS, João; DUQUE-ESTRADA, Tiago; JOLY, Carlos. **Tradução taxonômica: o caso do SinBiota**. Relatório técnico, Instituto de Computação, Universidade de Campinas, 2013.

MOREIRA, A. L. M.; HAYASHI, T. W. N.; COELHO, G. P.; SILVA, A. E. A. **A Clustering Method for Weak Signals to Support Anticipative Intelligence**. International Journal of Artificial Intelligence and Expert Systems, v. 6, p. 1-14, 2015.

MOREIRA, M. A. **Mapas Conceituais e Aprendizagem Significativa**. Instituto de Física: Universidade Federal do Rio Grande do Sul, 2012

NOROUZI, M; FLEET, D.J; SALAKHUTDINOV, R; **Hamming Distance Metric Learning**. Departments of Computer Science and Statistics University of Toronto, 2012.

NOVAK, J. D.; CAÑAS, A. J. **The Theory Underlying Concept Maps and How to Construct and Use Them**. IHMC CmapTools, p. 1–36, 2008.

PÉREZ, Cláudia C. C; VIEIRA, Renata. **Mapas Conceituais: geração e avaliação.** III Workshop em Tecnologia da Informação e da Linguagem Humana – TIL,2005.

PONTES, Elvys Linhares. **Utilização de grafos e matriz de similaridade na sumarização automática de documentos baseada em extração de frases.** Dissertação de mestrado. UFC, 2015

REZENDE, S. O., MARCACINI, R. M. e MOURA, M. F. **O uso da Mineração de Textos para Extração e Organização Não Supervisionada de Conhecimento**. Revista de Sistemas de Informação da FSMA n. 7 (2011) p. 7-21, 2011.

SILVA, Maria Estela Vieira da. **XSimilarity: Uma ferramenta para consultas por similaridade embutidas na linguagem XQuery**. Trabalho de graduação. UFRG,

2007.

XAVIER, Eduardo Semkiw; BATISTA, Jonathan da Silva. **Criação de um banco de dados não relacional a partir de informação extraída de textos**. 2018. 39 f. Trabalho de Conclusão de Curso (Tecnologia em Análise e Desenvolvimento de Sistemas) - Universidade Tecnológica Federal do Paraná, Ponta Grossa, 2018.