

PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS
INSTITUTO DE CIÊNCIAS EXATAS E INFORMÁTICA
UNIDADE EDUCACIONAL SÃO GABRIEL
Bacharelado em Sistemas de Informação

João Dias

Maycon Felipe Lourenço

Octavio Augusto Pereira Martins

Pedro Henrique Damasceno Dias

TRABALHO PRÁTICO

Especificação de um sistema de informação

Belo Horizonte

2021

SUMÁRIO

1. Introdução

Este trabalho busca colocar em prática os conhecimentos adquiridos na disciplina de Algoritmos em Grafos, desenvolvendo um sistema de gestão de TCC, que irá agrupar alunos que desenvolverão trabalhos com temas semelhantes. Será trabalhado o Algoritmo de Kruskal, além de conceitos como Árvore Geradora Mínima e Clusterização.

2. Objetivo e Proposta

A proposta desenvolvida propõe a utilização de Grafos para organizar a estrutura de um Sistema de Gestão de TCC. O objetivo é otimizar a divisão de grupos de alunos que estão desenvolvendo trabalhos com temas semelhantes.

Para isso, utilizamos o Algoritmo de Kruskal, para alcançar uma Árvore Geradora Mínima, para através dela, fazer a divisão dos clusters, que são equivalentes aos grupos de alunos.

3. Metodologias Adotadas

3.1. Algoritmo de Kruskal

A proposta do Algoritmo de Kruskal é encontrar uma árvore geradora mínima a partir de um grafo conexo, não dirigido e com pesos nas arestas, ou seja, o objetivo é encontrar um sub-grafo contendo todos os vértices ligados, e com o peso total (soma de todos os pesos das arestas) sendo o menor possível.

As condições para formação de uma árvore geradora mínima (MST) são:

- Para qualquer par de vértices distintos, sempre tem que haver um caminho que os une.
- Como todos os vértices estão conectados entre si, um vértice inicial não é necessário.

A partir do momento que conseguimos formar a MST, basta remover algumas arestas, até que restem n sub-grafos. Este processo de divisão da árvore é chamado de Clusterização.

3.2. Clusterização

3.2.1. Conceito

A palavra “Cluster” que está em inglês, significa “grupo”. Portanto, clusterizar nada mais é do que agrupar.

Este determinado agrupamento se diz tanto de separações conjunto de dados, de clientes, de computadores a fim de características em comum entre estes comparados.

Considera-se para o trabalho que a clusterização um algoritmo que tem como a sua regra ser obedecida sempre que um objeto tiver que ser enquadrado em uma categoria qualquer. São agrupados conforme a maior ou menor distância em relação a um outro com que tenham características comuns.

3.2.2. Objetivo para o trabalho

A separação dos clusters representa o agrupamento dos alunos a serem orientados por cada professor, que lecionam disciplinas com grau de dissimilaridade baixo, em outras palavras cada sub-grafo representa um grupo de alunos formado, que um ou mais professores iram lecionar estas disciplinas semelhantes, assim indica o fim do algoritmo.

3.2.3. Como será feita a formação dos Clusters

A formação dos clusters/grupo ou a separação de matérias semelhantes para um grupo de ensino será feita partir da Árvore Geradora Mínima, um sub-grafo com todos vértices ligados gerada pelo algoritmo de Kruskal.

A quantidade de arestas a serem removidas é equivalente a quantidade de professores, usado a fórmula:

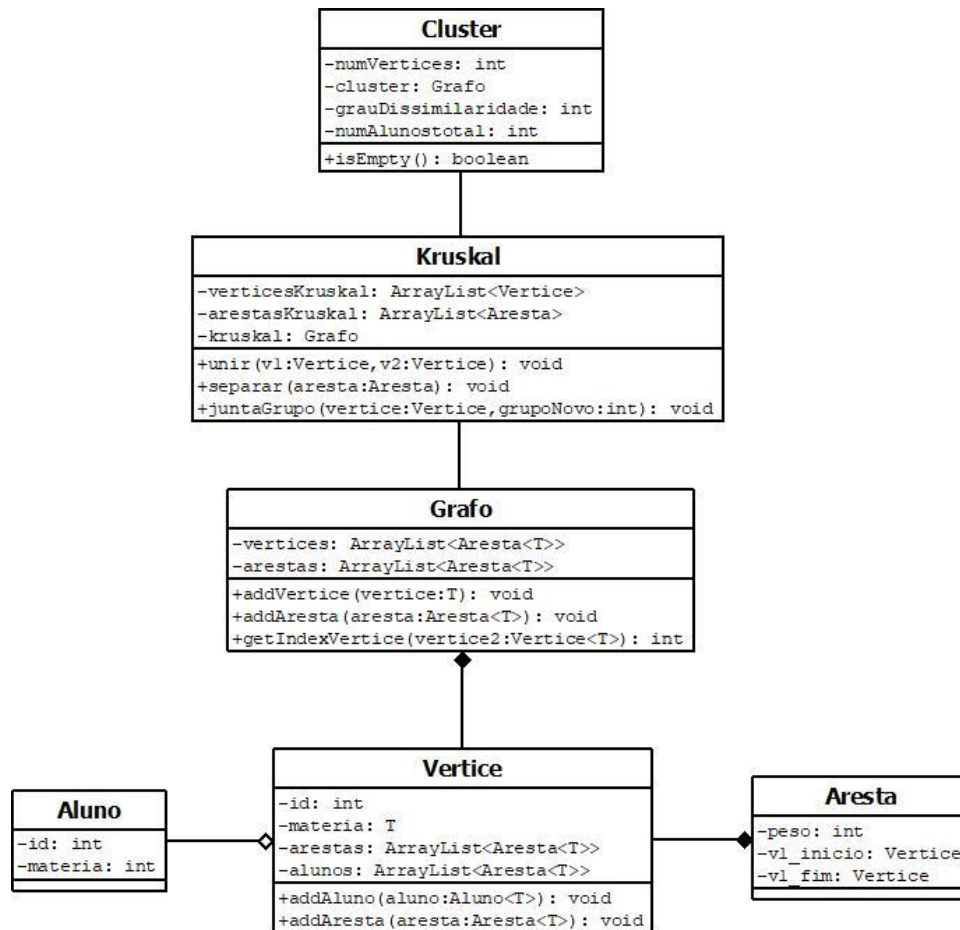
- Quantidade cortes = $n - 1$

A fórmula significa que a quantidade de arestas a serem cortadas será a quantidade totais de professores menos um. Em contrapartida, pode acontecer de terem mais professores do que as próprias arestas (o que é difícil), nesse caso o algoritmo irá cortar todas as arestas e identificará a sub-grafo que conter mais alunos para estudo, assim, alocar o professor para este sub-grafo.

A partir desse contexto o algoritmo de clusterização irá identificar os vértices com maiores graus e assim, identificará a aresta de maior peso entre o sub-grafo escolhido, em outras palavras irá remover as arestas que ligam duas matérias com maior grau de dissimilaridade, isto tudo a fim de formar n sub-grafos. Caso haja um “suposto” tipo de empate, em que entre as arestas identificadas para cortes

tenham o mesmo peso, o algoritmo irá mudar o vértice que foi identificado para o próximo de maior grau e irá refazer o processo.

4. Modelo UML



5. Resultados

Até certo ponto, apesar de algumas peculiaridades advindas do algoritmo de Kruskal, os resultados foram bem satisfatórios. Os pequenos pontos de insatisfação se dão devido a um problema intrínseco à utilização dessa lógica. O problema é o seguinte, quanto menor o número de professores, ou seja, quanto menor o número de clusters, maior será o grau de dissimilaridade em cada cluster. Além disso, quanto menor o número de clusters, sempre terá um cluster com um grau de dissimilaridade muito mais alto que os demais.

Teste com 2 clusters:

```
Quantos serão os professores?
2

Grupo: 1
Número de profs: 1 | Número de vértices: 14 | Grau Dissimilaridade: 133 | Número total alunos: 38

Grupo: 3
Número de profs: 1 | Número de vértices: 6 | Grau Dissimilaridade: 39 | Número total alunos: 12
```

Repare que o “Grupo 1” tem grau de dissimilaridade muito maior que o “Grupo 3”.

Teste com 4 clusters:

```
Quantos serão os professores?
4

Grupo: 0
Número de profs: 1 | Número de vértices: 2 | Grau Dissimilaridade: 10 | Número total alunos: 2

Grupo: 1
Número de profs: 1 | Número de vértices: 9 | Grau Dissimilaridade: 83 | Número total alunos: 27

Grupo: 3
Número de profs: 1 | Número de vértices: 6 | Grau Dissimilaridade: 39 | Número total alunos: 12

Grupo: 9
Número de profs: 1 | Número de vértices: 3 | Grau Dissimilaridade: 20 | Número total alunos: 9
```

Repare que o “Grupo 1” se mantém com o grau de dissimilaridade muito divergente dos demais.

Teste com 10 clusters:

```
Quantos serão os professores?
10

Grupo: 0
Número de profs: 1 | Número de vértices: 2 | Grau Dissimilaridade: 10 | Número total alunos: 2

Grupo: 4
Número de profs: 1 | Número de vértices: 3 | Grau Dissimilaridade: 8 | Número total alunos: 8

Grupo: 2
Número de profs: 1 | Número de vértices: 2 | Grau Dissimilaridade: 7 | Número total alunos: 3

Grupo: 3
Número de profs: 1 | Número de vértices: 3 | Grau Dissimilaridade: 12 | Número total alunos: 5

Grupo: 7
Número de profs: 1 | Número de vértices: 2 | Grau Dissimilaridade: 10 | Número total alunos: 4

Grupo: 9
Número de profs: 1 | Número de vértices: 3 | Grau Dissimilaridade: 20 | Número total alunos: 9

Grupo: 14
Número de profs: 1 | Número de vértices: 2 | Grau Dissimilaridade: 15 | Número total alunos: 8

Grupo: 12
Número de profs: 1 | Número de vértices: 1 | Grau Dissimilaridade: 0 | Número total alunos: 3

Grupo: 16
Número de profs: 1 | Número de vértices: 1 | Grau Dissimilaridade: 0 | Número total alunos: 4

Grupo: 18
Número de profs: 1 | Número de vértices: 1 | Grau Dissimilaridade: 0 | Número total alunos: 4
```

Essa divergência só se nivela a partir de 10 clusters.

Apesar de ser intrínseco ao tipo de algoritmo utilizado, não deixa de ser um problema.

Por fim, ao testarmos com 20 clusters, obtemos 20 diferentes grupos com Grau de dissimilaridade 0.

```
Quantos serão os professores?
20

Grupo: 0
Número de profs: 1 | Número de vértices: 1 | Grau Dissimilaridade: 0 | Número total alunos: 2

Grupo: 1
Número de profs: 1 | Número de vértices: 1 | Grau Dissimilaridade: 0 | Número total alunos: 3

Grupo: 2
Número de profs: 1 | Número de vértices: 1 | Grau Dissimilaridade: 0 | Número total alunos: 3

Grupo: 3
Número de profs: 1 | Número de vértices: 1 | Grau Dissimilaridade: 0 | Número total alunos: 2

Grupo: 4
Número de profs: 1 | Número de vértices: 1 | Grau Dissimilaridade: 0 | Número total alunos: 2

Grupo: 5
Número de profs: 1 | Número de vértices: 1 | Grau Dissimilaridade: 0 | Número total alunos: 0

Grupo: 6
Número de profs: 1 | Número de vértices: 2 | Grau Dissimilaridade: 0 | Número total alunos: 3

Grupo: 7
Número de profs: 1 | Número de vértices: 1 | Grau Dissimilaridade: 0 | Número total alunos: 3

Grupo: 8
Número de profs: 1 | Número de vértices: 1 | Grau Dissimilaridade: 0 | Número total alunos: 2

Grupo: 9
Número de profs: 1 | Número de vértices: 1 | Grau Dissimilaridade: 0 | Número total alunos: 1

Grupo: 10
Número de profs: 1 | Número de vértices: 1 | Grau Dissimilaridade: 0 | Número total alunos: 2

Grupo: 11
Número de profs: 1 | Número de vértices: 1 | Grau Dissimilaridade: 0 | Número total alunos: 5

Grupo: 12
Número de profs: 1 | Número de vértices: 1 | Grau Dissimilaridade: 0 | Número total alunos: 3

Grupo: 14
Número de profs: 1 | Número de vértices: 1 | Grau Dissimilaridade: 0 | Número total alunos: 3

Grupo: 15
Número de profs: 1 | Número de vértices: 1 | Grau Dissimilaridade: 0 | Número total alunos: 4

Grupo: 16
Número de profs: 1 | Número de vértices: 1 | Grau Dissimilaridade: 0 | Número total alunos: 4
```

6. Conclusão

Os resultados obtidos foram satisfatórios, visto que o uso de Grafos facilitou a alocação de professores e alunos para a determinada realização do TCC do curso Sistemas de Informação, mas que visa implementação em todo o Sistema de Gestão de TCC. O objetivo da separação/alocação de grupos foi realizado com sucesso pelo algoritmo de Kruskal, que visa selecionar um sub-grafo contendo todos os vértices ligados que foram separados pela clusterização.

Com isso, para trabalhos futuros, o modo de separação por dissimilaridade pode ser atualizado, visto que pode acontecer de professores estarem com matérias opostas dependendo da quantidade de professores, isto é algo bem mais

raro, mas que pode acontecer. Por fim, está aberto “novos ares” para trabalhos futuros, desde implementação de novos algoritmos caso no futuro seja descoberto um mais rápido e objetivo ao que se pede, verificação de outras propriedades, e também, não tão necessário, o incremento de visualização gráfica dos grafos para auxílio tanto de desenvolvedores como de usuários comuns.