

TP3 - AEDS 3

Paulo César Moraes de Menezes - 2022.1.08.024

Rafael Silva Fortuna - 2022.1.08.026

Bruno Martins Cordeiro - 2020.1.08.006

Mateus Henrique Martins - 2020.1.08.022

Vitor Reis Rubatino - 2019.1.08.017

1 Introdução

A centralidade em grafos é um conceito importante na análise de redes, que mede a importância ou influência relativa de um vértice em um grafo. A centralidade de um vértice é uma medida quantitativa que reflete sua posição estrutural em relação aos demais vértices da rede.

Existem várias medidas de centralidade em grafos, sendo as mais comuns:

Centralidade de grau (Degree centrality): Mede o número de arestas que estão conectadas a um vértice. Quanto maior o grau de um vértice, maior é sua centralidade.

Centralidade de proximidade (Closeness centrality): Mede a proximidade de um vértice em relação a todos os outros vértices do grafo. Um vértice com alta centralidade de proximidade está mais próximo de todos os outros vértices, em termos de distância percorrida ao longo do grafo.

Centralidade de intermediação (Betweenness centrality): Mede o número de vezes que um vértice atua como intermediário no caminho mais curto entre dois outros vértices. Um vértice com alta centralidade de intermediação tem um papel importante na comunicação e fluxo de informações dentro do grafo.

Centralidade de vetor próprio (Eigenvector centrality): Mede a influência de um vértice com base em sua conectividade com outros vértices importantes. Vértices que estão conectados a outros vértices com alta centralidade têm maior centralidade de vetor próprio.

O problema de centralidade em grafos consiste em determinar a centralidade de cada vértice em um dado grafo. O objetivo é identificar os vértices mais importantes ou influentes, que desempenham um papel crucial na estrutura e dinâmica da rede. A centralidade em grafos é amplamente utilizada em várias aplicações, como identificação de líderes, detecção de comunidades, análise de redes sociais, roteamento de redes de comunicação e muito mais.

É importante ressaltar que a escolha da medida de centralidade adequada depende do contexto e dos objetivos específicos da análise. Cada medida fornece uma perspectiva diferente sobre a importância dos vértices no grafo, e diferentes medidas podem

A centralidade em grafos é um conceito importante na análise de redes, que mede a importância ou influência relativa de um vértice em um grafo. A centralidade de um vértice é uma medida quantitativa que reflete sua posição estrutural em relação aos demais vértices da rede.

Existem várias medidas de centralidade em grafos, sendo as mais comuns:

Centralidade de grau (Degree centrality): Mede o número de arestas que estão conectadas a um vértice. Quanto maior o grau de um vértice, maior é sua centralidade.

Centralidade de proximidade (Closeness centrality): Mede a proximidade de um vértice em relação a todos os outros vértices do grafo. Um vértice com alta centralidade de proximidade está mais próximo de todos os outros vértices, em termos de distância percorrida ao longo do grafo.

Centralidade de intermediação (Betweenness centrality): Mede o número de vezes que um vértice atua como intermediário no caminho mais curto entre dois outros vértices. Um vértice com alta centralidade de intermediação tem um papel importante na comunicação e fluxo de informações dentro do grafo.

Centralidade de vetor próprio (Eigenvector centrality): mede a influência de um vértice com base em sua conectividade com outros vértices importantes. Vértices que estão conectadas a outros vértices com alta centralidade têm maior centralidade de vetor próprio.

O problema de centralidade em grafos consiste em determinar a centralidade de cada vértice em um dado grafo. O objetivo é identificar os vértices mais importantes ou influentes, que desempenham um papel crucial na estrutura e dinâmica da rede. A centralidade em grafos é amplamente utilizada em várias aplicações, como identificação de líderes, detecção de comunidades, análise de redes sociais, roteamento de redes de comunicação e muito mais.

É importante ressaltar que a escolha da medida de centralidade adequada depende do contexto e dos objetivos específicos da análise. Cada medida fornece uma perspectiva diferente sobre a importância dos vértices no grafo, e diferentes medidas podem

2 - Algoritmos

Os algoritmos utilizados no código são os seguintes:

2.1 - Algoritmo de Geração de Grafos Aleatórios:

O algoritmo gera um número aleatório de vértices dentro do intervalo [25, 75] usando random.randint. Gera uma densidade aleatória dentro do intervalo [0.35, 1] usando random.uniform. Cria um objeto grafo do tipo nx.Graph() para representar o grafo. Adiciona os vértices ao grafo usando grafo.add_nodes_from. Para cada par de vértices (i, j), em que $i < j$, sorteia aleatoriamente se uma aresta será adicionada com base na densidade desejada. Se a aresta for adicionada, é sorteado um peso aleatório entre 1 e 10 e adicionada ao grafo usando grafo.add_edge. Verifica se o grafo é conexo usando nx.is_connected. Se o grafo for desconexo, adiciona arestas para conectar os componentes conectados do grafo. Retorna o grafo gerado.

2.2 - Algoritmo de Cálculo da Centralidade de Betweenness:

O algoritmo recebe um grafo como entrada. Utiliza a função nx.betweenness_centrality do NetworkX para calcular a centralidade de betweenness para cada vértice do grafo. O resultado é um dicionário com o índice de betweenness para cada vértice. Retorna o dicionário de betweenness.

2.3 - Algoritmo de Coloração dos Vértices por Betweenness:

O algoritmo recebe o grafo e o dicionário de betweenness como entrada. Calcula o valor máximo de betweenness usando max. Atribui uma cor a cada vértice com base no índice de betweenness normalizado (entre 0 e 1). Retorna uma lista de cores para os vértices.

A complexidade dos algoritmos utilizados é a seguinte:

Algoritmo de Geração de Grafos Aleatórios: A geração do número de vértices e da densidade é feita em tempo constante. O loop aninhado para adicionar as arestas é executado no pior caso $O(n^2)$, onde n é o número de

vértices. A verificação de conexidade e a adição de arestas para conectar componentes desconexos também têm complexidade $O(n^2)$. Portanto, a complexidade total do algoritmo de geração do grafo é $O(n^2)$. Algoritmo de Cálculo da Centralidade de Betweenness: A função `nx.betweenness_centrality` tem complexidade $O(n^3)$ no pior caso, onde n é o número de vértices. Portanto, a complexidade total do algoritmo de cálculo da centralidade de betweenness é $O(n^3)$. Algoritmo de Coloração dos Vértices por Betweenness: A normalização do betweenness para cada vértice é feita em tempo constante. A criação da lista de cores tem complexidade $O(n)$, onde n é o número de vértices. Portanto, a complexidade total do algoritmo de coloração dos vértices é $O(n)$.

3 - Resultados

O código apresentado gera uma série de grafos aleatórios e calcula a centralidade de betweenness para cada um deles. Em seguida, os vértices são coloridos de acordo com o índice de betweenness e os gráficos resultantes são plotados e salvos como imagens PNG. Os resultados obtidos podem variar de execução para execução, devido à natureza aleatória da geração dos grafos. No entanto, os resultados geralmente mostram a distribuição de betweenness nos vértices, destacando aqueles que são mais centrais dentro do grafo. Além disso, os gráficos coloridos proporcionam uma visualização visualmente atraente da distribuição da centralidade de betweenness nos vértices, ajudando a identificar os vértices mais importantes ou influentes em cada grafo. Os resultados podem ser interpretados da seguinte maneira: A medida de betweenness indica a importância de um vértice como ponto de passagem para a comunicação entre outros vértices no grafo. Quanto maior o índice de betweenness, mais central é o vértice na rede. A coloração dos vértices de acordo com o betweenness permite visualizar visualmente os vértices mais centrais em cada grafo. Vértices com cores mais escuras (mais próximas do roxo) indicam um maior índice de betweenness e, portanto, uma maior importância no fluxo de comunicação da rede. Esses resultados podem ser úteis em diversas aplicações, como a identificação de vértices-chave em redes sociais, sistemas de transporte ou fluxos de informação, por exemplo. Através da análise da centralidade de betweenness, é possível identificar elementos cruciais e pontos de estrangulamento em uma rede, auxiliando na tomada de decisões e no entendimento do comportamento do sistema representado pelo grafo.









