

UNIVERSIDADE DE PERNAMBUCO - UPE  
ESCOLA POLITÉCNICA DE PERNAMBUCO - POLI

# Modelagem de redes sociais com grafos

Uma abordagem baseada na interação de usuários

Mikael Silva

Professor: Cleyton Rodrigues

Recife – PE  
Julho de 2025

UNIVERSIDADE DE PERNAMBUCO - UPE  
ESCOLA POLITÉCNICA DE PERNAMBUCO - POLI

# Modelagem de redes sociais com grafos

Uma abordagem baseada na interação de usuários

Mikael Silva

Artigo apresentado à disciplina de  
Estrutura de Dados  
do curso de Engenharia de Computação da  
Escola Politécnica de Pernambuco – UPE.  
Professor: Cleyton Rodrigues

Recife – PE  
Julho de 2025

## 1 - Introdução

O presente artigo tem por objetivo apresentar os conceitos aprendidos na disciplina de Estrutura de Dados através do uso de grafos para simular interações entre usuários de uma rede social, de modo a entender a importância dos grafos na representação e análise de sistemas complexos. A base de dados escolhida para a modelagem foi o Instagram, devido a sua popularidade e também ao fato de a maioria das pessoas estarem familiarizadas com a plataforma, facilitando o entendimento do projeto aqui apresentado. O código foi escrito na linguagem Python e está disponível no github do autor.<sup>1</sup>

## 2 - Classes

Há cinco classes no projeto, cada uma com sua função específica. A classe *User* (os vértices do grafo) representa os usuários da nossa rede, que daqui para frente será chamada de UniPoli. A classe *Conection*(as arestas do grafo) representa as conexões entre os usuários da UniPoli. As arestas são ponderadas e direcionadas, isto é, as conexões têm peso que representam o quanto um usuário A curtiu, comentou ou compartilhou posts do usuário B e direcionadas pois o usuário A pode seguir o usuário B, mas B pode não seguir A e vice-versa, tornando essa aresta(conexão) direcionada. A classe *Post* representa os posts que os usuários postam na rede. A *UniPoli*, classe principal, representa o nosso app. É nela que está contida praticamente toda a lógica do projeto. E por fim, a classe *Gui*, que gera uma interface gráfica para o projeto.

## 3- Análise de algoritmos

Foi implementado dois algoritmos clássicos de grafos, DFS(Depth-First Search) e BFS(Breadth-First Search). O BFS explora o grafo em camadas, visitando todos os vizinhos

---

<sup>1</sup>[Github do autor](#)

de um nó antes de avançar. Ele é ideal para encontrar o caminho mais curto em termos de conexões. Na UniPoli, ele é usado para recomendar amigos para um determinado usuário(amigo de amigo), pessoas que estão a 2 nós de distância do usuário. Já o DFS explora o grafo profundamente, visitando um caminho até o fim antes de retroceder. Isso pode ser útil para encontrar conexões profundas, mas pode consumir mais memória em grafos grandes.

## 4 - Métricas

**4.1- Complexidade temporal.** Análise do tempo de execução do algoritmo BFS em diferentes tamanhos do grafo (número de usuários e conexões). Para um grafo com 20 usuários o tempo de execução do BFS é mostrado abaixo:

```
In [50]: # Tempo de execução

import time
start = time.time()
_ = rede.recommend('Mikael') # Coloca o output em _ Lambda
end = time.time()
print(f'Time taken: {end - start:.5f} seconds')

--- Gerando recomendações para: Mikael ---

--- Iniciando BFS a partir de: Mikael ---
Visitando: Mikael (Distância: 0)
Adicionado à fila: Madalena
Visitando: Madalena (Distância: 1)
Adicionado à fila: Ana
Visitando: Ana (Distância: 2)
--- BFS Concluída. Ordem: ['Mikael', 'Madalena', 'Ana'] ---
--- Recomendações para Mikael: ['Ana'] ---
Time taken: 0.00000 seconds
```

**4.2 - Grau dos Nós (Degree).** O grau de um nó (usuário) mede o número de conexões que ele possui. Em grafos direcionados como a UniPoli, tem-se:

- **Grau de Entrada (In-degree):** Representa o número de usuários que seguem um determinado usuário. Um alto grau de entrada indica popularidade ou que o usuário é um receptor significativo de informações.
- **Grau de Saída (Out-degree):** Indica o número de usuários que um determinado usuário segue. Um alto grau de saída pode caracterizar um usuário que consome bastante conteúdo ou que é muito ativo em seguir outros.
- **Grau Total:** A soma do grau de entrada e de saída, refletindo a atividade geral de conexão do usuário na rede.

**4.3 - Centralidade (Centrality).** As métricas de centralidade são utilizadas para identificar os nós mais "importantes" ou "influentes" dentro da rede:

- **Centralidade de Grau (Degree Centrality):** Uma versão normalizada do grau. Um usuário com alta centralidade de grau está fortemente conectado à rede, seja por seguir ou ser seguido por muitos outros.
- **Centralidade de Intermediação (Betweenness Centrality):** Mede a frequência com que um usuário está nos caminhos mais curtos entre outros pares de usuários. Usuários com alta centralidade de intermediação atuam como "pontes" ou "conectores" entre diferentes partes da rede, controlando o fluxo de informação.
- **Centralidade de Proximidade (Closeness Centrality):** Avalia a proximidade de um usuário a todos os outros na rede. Usuários com alta centralidade de proximidade podem alcançar outros usuários rapidamente, tornando-os eficientes na disseminação de informações.

**4.4 - Componentes Conectados (Weakly Connected Components).** Em um grafo direcionado, um componente fracamente conectado é um subconjunto de nós onde cada nó pode ser alcançado a partir de qualquer outro nó no mesmo subconjunto, ignorando a direção das arestas. Essa métrica ajuda a identificar comunidades ou clusters dentro da UniPolí, revelando grupos de usuários que estão interligados, mesmo que indiretamente. O número de componentes indica o nível de fragmentação da rede.

**4.5 - Diâmetro da Rede (Diameter).** O diâmetro da rede é a maior distância (número de conexões) entre quaisquer dois usuários no grafo (ou no maior componente conectado, caso a rede seja fragmentada). Um diâmetro pequeno sugere que a informação pode se espalhar rapidamente por toda a rede, indicando uma estrutura mais compacta.

**4.6 - Densidade do Grafo (Density).** A densidade do grafo mede o quão "completa" a rede é, calculando a proporção de conexões existentes em relação ao número total de conexões que seriam possíveis. Uma densidade próxima de 1 indica uma rede muito interligada, enquanto uma densidade próxima de 0 sugere uma rede esparsa. Em redes sociais, a densidade geralmente é baixa, pois é incomum que todos os usuários sigam todos os outros.

---

--- Métricas da Rede UniPoli ---

```
1. Grau dos Nós:
- Grau Total (In + Out):
  Mikael: 3
  Madalena: 0
  Ana: 1
  Elon Musk: 1
  Maria: 0
  Caio: 0
  Otavio: 0
  Julia: 0
  Vitoria: 0
  Bruno: 1
  Jose: 0
  Trump: 0
  Hanry: 0
  Chris: 0
  Bob: 0
  Robert: 0
  Cleiton: 0
  Nilton: 0
  Zé de manga: 0
  Zé do rádio: 0
- Grau de Entrada (In-degree):
  Mikael: 3
  Madalena: 0
  Ana: 0
  Elon Musk: 0
  Maria: 0
  Caio: 0
  Otavio: 0
  Julia: 0
  Vitoria: 0
  Bruno: 0
  Jose: 0
  Trump: 0
```

## 5 - Conclusão

O desenvolvimento deste projeto foi uma experiência enriquecedora. Permitiu-me não apenas consolidar o conhecimento sobre estruturas de dados, mas também aprofundar a compreensão da fundamental importância dos grafos na representação e análise de sistemas complexos, como as redes sociais. A capacidade de aplicar algoritmos e métricas para extrair informações significativas de dados conectados é uma habilidade crucial na ciência da computação e na análise de dados, e este projeto serviu como um excelente ponto de partida para explorar essas áreas.