

Projeto 1

Inácio Leal do Rosário - 232011331
João Marcelo Costa de Santana - 232023790
Carlos Caua Rocha da Silva - 231034304
Prof. Díbio — Teoria e Aplicação de Grafos

Resumo

Este relatório analisa uma rede social baseada nos dados de *Facebook Ego-Networks*, utilizando métricas de grafos para identificar usuários mais influentes e importantes na disseminação de informações. Foram construídos grafos com 2000 nós amostrados, aplicando o algoritmo de Louvain para detectar comunidades e calculando centralidades de grau, intermediação, proximidade e autovetor. Os resultados visuais e numéricos são discutidos a partir dos padrões observados no subgrafo.

1 Introdução

O estudo de redes sociais sob a perspectiva da Teoria dos Grafos permite compreender estruturas complexas de interação entre indivíduos. Este projeto visa aplicar técnicas computacionais de análise de grafos ao conjunto de dados *Facebook Ego-Networks*, disponibilizado pela Stanford Network Analysis Project (SNAP). O objetivo é identificar padrões de conectividade, agrupamentos naturais (comunidades) e nós de maior relevância na rede.

2 Metodologia

2.1 Coleta e Construção do Grafo

Os dados utilizados são provenientes dos arquivos `.edges` do conjunto de *ego-networks* do Facebook, onde cada arquivo representa as conexões entre amigos de um usuário específico (ego). Utilizando a biblioteca `NetworkX`, foi construído um grafo completo conectando:

- As arestas entre amigos (arestas internas);
- O nó ego e seus respectivos amigos.

Após a construção do grafo completo, uma amostra aleatória de 2000 nós foi extraída para reduzir a complexidade e permitir análises mais detalhadas.

2.2 Análise de Comunidades

O algoritmo de Louvain foi aplicado para detectar comunidades, maximizando a modularidade. Cada nó foi associado a uma comunidade, e as cores nos gráficos representam a partição obtida.

2.3 Cálculo das Centralidades

Foram calculadas quatro medidas clássicas de centralidade:

1. **Grau:** mede o número de conexões diretas de cada nó.
2. **Intermediação (Betweenness):** quantifica o papel de ponte de um nó entre diferentes partes da rede.
3. **Proximidade (Closeness):** avalia quão próximo um nó está dos demais na rede.
4. **Autovetor (Eigenvector):** mede a influência de um nó com base na importância de seus vizinhos.

2.4 Visualização

Os grafos foram plotados com o layout de mola (`spring_layout`), destacando as comunidades e os nós com maiores valores de centralidade.

2.5 Descrição do Código

A implementação foi realizada em Python 3, utilizando as bibliotecas `pandas`, `networkx`, `matplotlib`, `glob` e `community-louvain`. O código é dividido em cinco etapas principais:

1. **Construção do grafo completo:** cada arquivo `.edges` é lido e interpretado como um conjunto de conexões entre amigos. O identificador numérico do arquivo representa o nó *ego*, que é conectado a todos os amigos listados no arquivo. O resultado é um grafo não direcionado com milhares de nós e arestas.
2. **Amostragem e subgrafo:** para reduzir a complexidade computacional, é realizada uma amostragem aleatória de 2000 nós a partir do grafo completo, gerando um subgrafo (`subgraph()`) com densidade e estrutura similares à rede original.
3. **Visualização inicial:** a função `spring_layout()` gera uma disposição espacial baseada em forças, aproximando nós conectados e afastando os desconectados. O grafo é exibido com `matplotlib`.
4. **Detecção de comunidades:** o algoritmo de Louvain (`community_louvain.best_partition`) é aplicado sobre o subgrafo, retornando a atribuição de cada nó a uma comunidade, maximizando a modularidade global.
5. **Cálculo e visualização das centralidades:** são computadas as métricas de grau, intermediação, proximidade e autovetor usando funções do `NetworkX`. Em seguida, são gerados gráficos que destacam os 10 nós mais centrais em cada medida, com tamanhos e cores proporcionais à importância de cada vértice.

O fluxo do código segue uma lógica modular e reproduzível, permitindo adaptar o mesmo pipeline para diferentes conjuntos de dados ou tamanhos de amostra.

3 Resultados e Discussão

3.1 Visão Geral

Parâmetro	Valor
Nós no grafo completo	3963
Arestas no grafo completo	88156
Nós na amostra (subgrafo)	2000
Arestas no subgrafo	23042

A rede amostrada apresentou certa estrutura modular, com grupos de nós mais conectados entre si, mas ainda mantendo diversas conexões externas. Esse padrão é comum em redes sociais, onde os usuários formam círculos de amizade ou interesse, mas continuam ligados a outros grupos por conexões intermediárias.

3.2 Análise das Centralidades

Os nós com maior grau tendem a atuar como hubs locais, enquanto os de alta intermediação conectam diferentes comunidades. Já os de alta centralidade de autovetor representam indivíduos influentes, conectados a outros igualmente influentes.

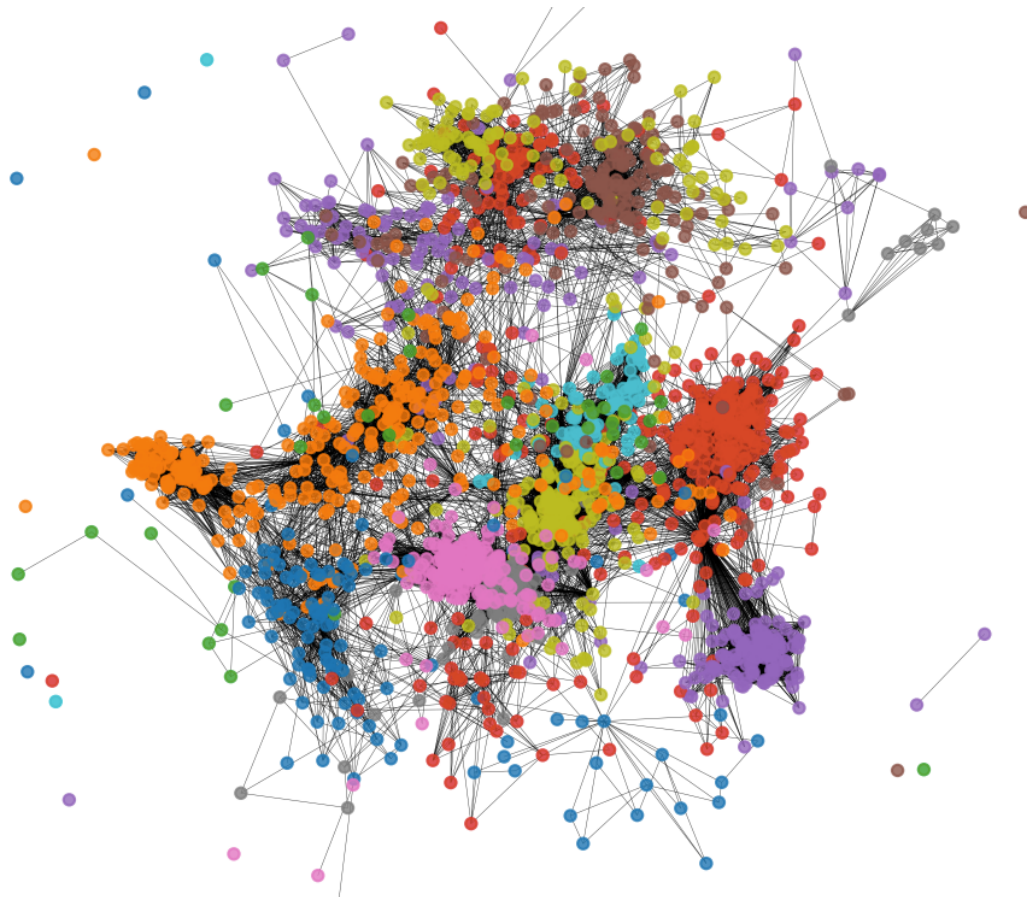


Figura 1: Visualização das comunidades detectadas pelo algoritmo de Louvain.

4 Conclusão

A análise revelou uma rede social heterogênea, composta por múltiplas comunidades interligadas por nós de alta intermediação. Esses nós desempenham papel central na difusão de informações e na coesão estrutural da rede. As métricas calculadas (grau, intermediação, proximidade e autovetor) demonstraram coerência entre si, validando o método aplicado. O algoritmo de Louvain mostrou-se eficiente em identificar comunidades coesas e de alta modularidade.