

经济管理学院

课 程 报 告

(复杂网络与社会计算)

题 目: week3 课程作业

课程教师: 赵吉昌

学院/专业: 经济管理学院/信管

学生姓名: 范春

学 号: 21377061

2024 年 3 月 13 日



北京航空航天大学
BEIHANG UNIVERSITY

题目要求:

- 2. 从Index of Complex Networks (<https://icon.colorado.edu>) 下载技术网络 **Gnutella p2p networks**, 并进行恶意攻击实验, 观察最大连通分支占网络规模 (P_∞) 随攻击比例 (f) 的变化。参照1中资料的思路, 比较不同节点重要性度量方法在攻击中的差异。假定 $P_\infty < 0.8$ 时, 对应的 f 为 f_c , 请比较本周给出的几类重要性指标中, **哪个指标的攻击性更强 (即 f_c 最小)**。注意, 可以在攻击时提前计算好所有节点的重要性并排序, 实验时分别尝试不同的攻击比例, 如0.01, 0.02, 0.05等。另外, 注意 `nx.Graph` 的 `copy` 方法, 因为网络结构在攻击中会被破坏, 需要保存初始副本, 以用于其他 f 。
- 3. 在2中, 同时观察第二大连通分支占网络规模的比例随 f 的变化规律, 并观察有无稳定的现象出现。
- 4. 在社交媒体中一些场景中, 会有社交关系的“失效”, 如取消关注、拉黑等。思考并讨论这类行为如何与复杂网络的鲁棒性关联及其可能导致的影响。

问题 2 和问题 3, 完整代码如下:

```
import networkx as nx
import matplotlib.pyplot as plt

def network(filePath):
    """
    生成网络
    """
    G = nx.Graph()
    with open(filePath, 'r') as f:
        lines = f.readlines()
        for line in lines[4:]:
            row = line.strip().split('\t')
            G.add_edge(int(row[0]), int(row[1]))
    return G

def giant_component_ratio(G):
    """
    计算最大连通分支和第二连通分支占网络规模的比值
    """
    results = []
    ccs = sorted(nx.connected_components(G), key=len, reverse=True)
    largest_cc = ccs[0]
    giant_component_size = len(largest_cc)
    network_size = len(G.nodes())
    giant_component_ratio = giant_component_size / network_size
    results.append(giant_component_ratio)
    if (len(ccs) > 1):
        second_cc = ccs[1]
        second_component_size = len(second_cc) / network_size
        results.append(second_component_size)
```

```

else:
    print("该网络连通，无第二大连通分支！")
    return results

def importanceIndex(G):
    """
    计算节点的重要性的各个指标
    """
    degree Centrality = nx.degree_centrality(G) #度中心性
    nbc = nx.betweenness_centrality(G) #介数
    Cc = nx.closeness_centrality(G, u=None, distance=None,
wf_improved=True) #接近中心性
    core = nx.core_number(G) #核数
    return [degree_centrality,nbc,Cc,core]

def attack_network(G, importance_measure,px):
    nodes_sorted = sorted(importance_measure, key=importance_measure.get,
reverse=True)
    num_nodes_to_remove = int(len(nodes_sorted)*px)
    nodes_to_remove = nodes_sorted[:num_nodes_to_remove]
    G_copy = G.copy()
    G_copy.remove_nodes_from(nodes_to_remove)
    ratio = giant_component_ratio(G_copy)
    ratio1 = ratio[0]
    ratio2 = ratio[1]
    return ratio1,ratio2

def main():
    filePath = "C:\\Users\\范春
\\Desktop\\week3\\p2p-Gnutella04.txt\\p2p-Gnutella04.txt"
    G = network(filePath)
    importanceindex = importanceIndex(G)
    item = ["degree_centrality","nbc","Cc","core"]
    for j in range(4):
        resultPath = f"C:\\Users\\范春\\Desktop\\{item[j]}.txt"
        with open(resultPath , 'w') as result_file:
            for i in range(1,81):
                px = i*0.01
                ratio1, ratio2 = attack_network(G,importanceindex[j],px)
                result_file.write(f"{px}\\t{ratio1}\\t{ratio2}\\n")
            print(f"{item[j]}成功\\n")

if __name__=="__main__":
    main()

```

下述代码用于绘制最大连通分支和第二连通分支占网络规模的比值随着攻击比例变化的曲线图。

```
import matplotlib.pyplot as plt

def read_txt(file_name):
    with open(file_name, 'r') as file:
        lines = file.readlines()
        data = [[float(x) for x in line.strip().split()] for line in lines]
    return data

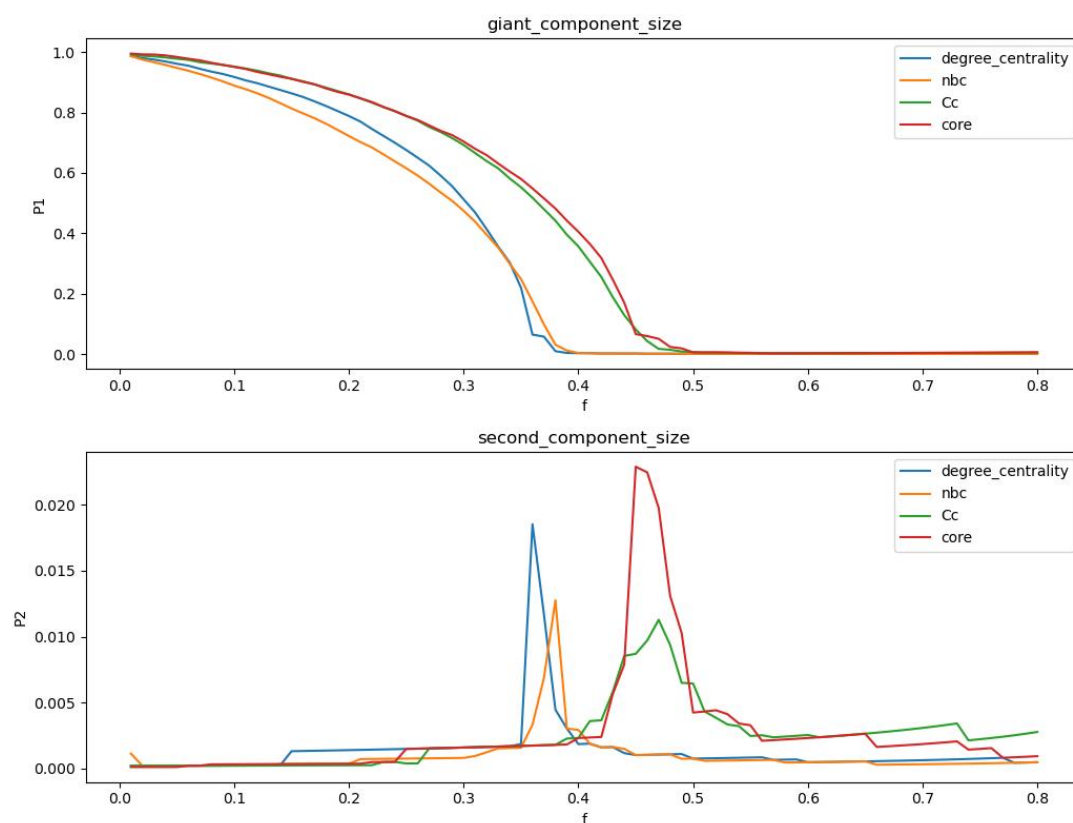
def plot_graph(file_names, names):
    fig, axs = plt.subplots(2, figsize=(10, 8))
    for i, file_name in enumerate(file_names):
        data = read_txt(file_name)
        f = [row[0] for row in data]
        P1 = [row[1] for row in data]
        P2 = [row[2] for row in data]
        axs[0].plot(f, P1, label=names[i])
        axs[1].plot(f, P2, label=names[i])

    axs[0].set_title('giant_component_size')
    axs[0].set_xlabel('f')
    axs[0].set_ylabel('P1')
    axs[0].legend()
    axs[1].set_title('second_component_size')
    axs[1].set_xlabel('f')
    axs[1].set_ylabel('P2')
    axs[1].legend()

    plt.tight_layout()
    plt.show()

if __name__ == "__main__":
    file_names = ["C:\\Users\\范春\\Desktop\\degree_centrality.txt",
                  "C:\\Users\\范春\\Desktop\\nbc.txt", "C:\\Users\\范春\\Desktop\\Cc.txt",
                  "C:\\Users\\范春\\Desktop\\core.txt"]
    names = ["degree_centrality", "nbc", "Cc", "core"]
    plot_graph(file_names, names)
```

运行结果如下：



在本次实验中，我选取了度中心性（degree centrality）、介数（nbc）、接近中心性（Cc）、核数（core）四个指标作为节点的重要性度量，分别绘制了四种指标下最大连通分支和第二大连通分支分别占网络规模随攻击比例的变化。假定 $P_\infty < 0.8$ 时对应的 f 为 f_c ，由图可知介数指标的 f_c 值最小，即该指标的攻击性更强，然后为度中心性，接近中心性和核数的 f_c 值基本相同。当攻击比例 $f < 0.3$ 或 $f > 0.6$ 时，第二大连通分支占网络规模的比例变化较小，比较平稳；当攻击比例 $0.3 < f < 0.6$ 时，第二大连通分支占网络规模的比例变动较大，出现峰值。

问题 4:

社交媒体网络一般可以看作无标度网络，少数节点具有极高的度数，而大多数节点具有较低的度数，这就好比社交媒体上的知名人物，他们一般拥有大量的关注者，从而度数较高，但绝大多数普通网民不被人所熟知，由于交际圈有限，所以充当的是度数较小的点。因此社交媒体网络对随机故障是鲁棒的，而对于恶意攻击是极端脆弱的。

社交关系的“失效”，如取消关注、拉黑等行为在社交媒体中是比较常见的行为，对于普通用户，即度数较小的点来说，其变化并不会对整个媒体网络带来

实质性的影响，但如果该事件是针对知名人物的，例如对名人的恶意造谣、抹黑，从而造成大量粉丝取关、拉黑等，那么就会对该媒体网络造成巨大影响，例如影响网络结构，导致网络中节点度分布不均匀，出现更多的孤立结点；降低网络的鲁棒性；影响信息的传播。