МІНІСТЕРСТВО НАУКИ І ОСВІТИ УКРАЇНИ

КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

ІМЕНІ ТАРАСА ШЕВЧЕНКА

ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

**ПРАКТИЧНЕ ЗАНЯТТЯ № 4 ЗА ТЕМОЮ:**

**Методи аналізу моделей соціальних мереж з використанням теорії графів. Google web search**

МІНІСТЕРСТВО НАУКИ І ОСВІТИ УКРАЇНИ

КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

ІМЕНІ ТАРАСА ШЕВЧЕНКА

ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

ЗВІТ З ПРАКТИЧНОЇ РОБОТИ № 4ЗА ТЕМОЮ:

**Методи аналізу моделей соціальних мереж з використанням теорії графів. Google web search**

Група \_\_\_\_\_11\_\_\_\_\_\_

Курс \_\_\_\_\_1\_\_\_\_\_\_\_\_\_

Студент (ка) \_\_\_\_Супруненко М. І. \_\_\_\_

Дата оформлення \_\_\_\_\_\_28.02.2025\_\_\_\_\_\_\_

Перевірив \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Дата \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

МЕТА: Метою практичної роботи є формування професійних вмінь та навичок щодо використання інструментів та методів аналізу моделей соціальних мереж з використанням теорії графів, концепцію жорстко пов’язаних компонентів мережі, основи web пошуку та ранжування сторінок PageRank, параметри мертвих вузлів та пасток у web пошуку та вміння застосовувати отримані знання на практиці в практичних задачах аналізу великих даних.

ОСНАЩЕННЯ:

ПЛАН ПРОВЕДЕННЯ ПРАКТИЧНОГО ЗАНЯТТЯ:

1. Відповіді на вхідні питання.
2. Проведення досліджень невеликих моделей соціальних мереж. Виявлення особливості мережі (наявність пасток, мертвих вузлів).
3. Підготовка звіту за результатами дослідження.
4. Захист отриманих результатів досліджень.

ПИТАННЯ ДЛЯ ВХІДНОГО КОНТРОЛЮ СТУДЕНТІВ:

1. Соціальна мережа як граф.
2. Графи з кількома типами вузлів.
3. Small-world Effect. Small world graph.
4. Кластеризація соціальних мереж.
5. Нерівномірний (перекошений) розподіл ступенів.
6. Моделювання зв’язків між об’єктами в соціальній мережі .
7. Random networks. Scale-free network.
8. Степеневий закон (закон масштабування).
9. Community detection in social networks. Node Betweenness. Edge betweenness.
10. Виявлення спільноти в соціальних мережах.
11. Структура інтернету. Strongly Connected Components (SCC).
12. Introduction to web search.
13. PageRank. Мертві вузли (Dead nodes). Пастка павука (Spider traps).

ХІД ПРОВЕДЕННЯ ПРАКТИЧНОГО ЗАНЯТТЯ:

1. Проаналізуйте граф соціальної мережі (згідно з додатком 1). В результаті необхідно вияснити чи є мертві вузли та пастки в графі мережі та визначитися з методами розрахунку показника PageRank.
2. Створіть матрицю суміжності.
3. Створіть матрицю переходів.
4. Розрахуйте початковий вектор.
5. Розрахуйте значення показника PageRank для кожного вузла.
6. У разі наявності мертвих вузлів або пасток, використовуйте відповідні методи, розглянуті в лекційному матеріалі.
7. Оформіть розрахунки пп.3-5 у вигляді функцій та циклів для уніфікації та можливості моделювання різних матриць.

ДОМАШНЄ ЗАВДАННЯ: Підготовка звіту та захисту отриманих результатів.

Додаток 1

|  |  |
| --- | --- |
| 9 | C:\Users\it_admin\Documents\Наука_Книги_Уроки\Університет_Шевченко\Моделі_та_методи_обробки_великих_даних\Лабораторні роботи\Рис\PR_9.jpg |

|  |
| --- |
| import numpy as np  import networkx as nx  import matplotlib.pyplot as plt  # 1. Створення графа (визначення вузлів та ребер)  G = nx.DiGraph()  edges = [(1, 2), (1, 3), (2,1), (2, 4), (2,6), (3, 1),(3,7), (4, 5), (4, 6), (5, 7), (6, 6), (7, 5), ]  G.add\_edges\_from(edges)  # 2. Встановлення кольорів для вузлів згідно зображення  node\_colors = {1: 'blue', 2: 'red', 3: 'yellow', 4: 'purple', 5: 'purple', 6: 'orange', 7: 'orange'}  colors = [node\_colors[node] for node in G.nodes()]  # 3. Фіксоване розташування вузлів, щоб відповідало зображенню  pos = {      1: (1, 0),      2: (2, 1),      3: (0, -1),      4: (1.5, 2),      5: (-0.5, 1.5),      6: (1, 1),      7: (-1, 0)  }  # 4. Візуалізація графа  plt.figure(figsize=(6, 6))  nx.draw(G, pos, with\_labels=True, node\_color=colors, edge\_color='black', node\_size=2000, font\_size=14, arrows=True)  plt.show()  # 5. Створення матриці суміжності  adj\_matrix = nx.to\_numpy\_array(G, nodelist=sorted(G.nodes()))  print("Матриця суміжності:")  print(adj\_matrix)  # 6. Побудова матриці переходів (нормалізація)  def transition\_matrix(adj\_matrix):      P = adj\_matrix.copy()      for i in range(len(P)):          if P[i].sum() > 0:              P[i] /= P[i].sum()      return P  P = transition\_matrix(adj\_matrix)  print("Матриця переходів:")  print(P)  # 7. Обчислення PageRank  alpha = 0.85  # Фактор загасання  n = len(P)  r = np.ones(n) / n  # Початковий вектор  def pagerank(P, alpha=0.85, tol=1.0e-6):      n = P.shape[0]      r = np.ones(n) / n  # Початковий вектор      delta = 1  # Початкове значення для циклу      while delta > tol:          r\_new = alpha \* P.T @ r + (1 - alpha) / n          delta = np.linalg.norm(r\_new - r, 1)          r = r\_new      return r  page\_rank\_values = pagerank(P, alpha)  print("PageRank значення для вузлів:")  for i, rank in enumerate(page\_rank\_values, start=1):      print(f"Вузол {i}: {rank:.4f}")  # 8. Виявлення мертвих вузлів (без вихідних ребер) та пасток  dead\_nodes = [node for node in G.nodes() if G.out\_degree(node) == 0]  print("Мертві вузли:", dead\_nodes)  # 9. Виявлення пасток  spider\_traps = [node for node in G.nodes() if G.in\_degree(node) > 0 and G.out\_degree(node) == 1 and list(G.successors(node))[0] == node]  print("Пастки павука:", spider\_traps) |
|  |
| Матриця суміжності:  [[0. 1. 1. 0. 0. 0. 0.]  [1. 0. 0. 1. 0. 1. 0.]  [1. 0. 0. 0. 0. 0. 1.]  [0. 0. 0. 0. 1. 1. 0.]  [0. 0. 0. 0. 0. 0. 1.]  [0. 0. 0. 0. 0. 1. 0.]  [0. 0. 0. 0. 1. 0. 0.]]  Матриця переходів:  [[0. 0.5 0.5 0. 0. 0. 0. ]  [0.33 0. 0. 0.33 0. 0.33 0.]  [0.5 0. 0. 0. 0. 0.0.5 ]  [0. 0. 0. 0. 0.5 0.50. ]  [0. 0. 0. 0. 0. 0. 1. ]  [0. 0. 0. 0. 0. 1.0. ]  [0. 0. 0. 0. 1. 0.0. ]]  PageRank значення для вузлів:  Вузол 1: 0.0524  Вузол 2: 0.0437  Вузол 3: 0.0437  Вузол 4: 0.0338  Вузол 5: 0.2515  Вузол 6: 0.3212  Вузол 7: 0.2538  Мертві вузли: []  Пастки павука: [6] |

З наданих даних можна зробити кілька висновків:

Матриця суміжності показує з'єднання між вузлами графу. Якщо в матриці стоїть одиничне значення, це означає, що між відповідними вузлами існує зв'язок (з'єднання), якщо нуль — зв'язку немає.

Матриця переходів демонструє ймовірності переходу між вузлами, де кожен елемент матриці представляє ймовірність переходу з одного вузла в інший. Важливо, що ймовірності для кожного вузла сумуються до одиниці, що вказує на нормалізацію цих значень.

PageRank значення для кожного вузла вказують на "важливість" кожного з вузлів у графі. Чим вище значення PageRank для вузла, тим більша ймовірність, що цей вузол є важливим для структури графа. Вузол 6 має найбільше значення (0.3212), що свідчить про його високу важливість, а вузол 4 — найменше (0.0338), що свідчить про його меншу роль.

Мертві вузли — це вузли, які не можуть бути досягнуті з інших вузлів, і в даному випадку таких вузлів немає.

Пастки павука — це вузли, в яких "застрягли" переходи, тобто, з яких не можна вийти, або вони мають властивість вести лише до себе. У цьому випадку пастка павука знаходиться на вузлі 6, що може вказувати на те, що цей вузол має цикл, з якого не виходить.

**Висновок:**

В результаті виконання практичної роботи було проаналізовано граф з використанням методу PageRank для оцінки важливості вузлів. Результати показали, що вузол 6 є найбільш важливим, а вузол 4 — найменш важливим. Також було виявлено, що у графі є пастка павука (вузол 6), що може вказувати на структуру графа з циклозалежними вузлами.