The University of New South Wales - COMP9312 - 24T2 - **Data Analytics for Graphs**

Assignment 2

Distributed Graph Processing, Feature Engineering, and Graph Neural Networks

Important updates:

Update @ 25 Jul 2024

- Fix the error or the weight matrix W and make the GAT layer formulation clearer in Q2.1.

Summary

Submission	Submit an electronic copy of all answers on **Moodle** (Only the last submission will be used).
Required Files	A `.pdf` file is accepted. The file name should be `ass2_zid.pdf`
Deadline	**9 pm Friday 2 August** (Sydney Time)
Marks	**20** marks (**10%** toward your total marks for this course)

Late penalty. 5% of max assessment marks will be deducted for each additional day (24 hours) after the specified submission time and date. No submission is accepted 5 days (120 hours) after the deadline.

START OF QUESTIONS

Q1 (3 marks)

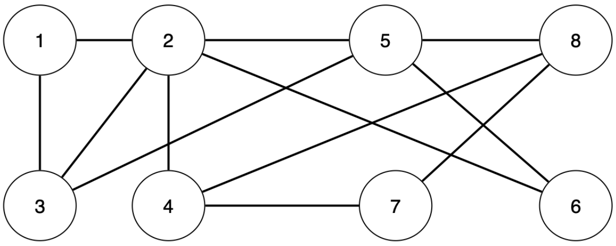
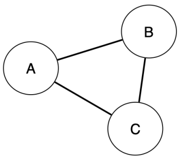


Figure 1



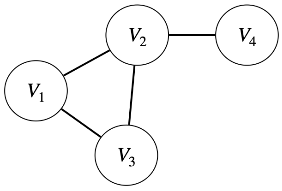


Figure 2

Figure 3

1.1 Are the graphs in Figure 1 and Figure 2 homomorphic? If so, demonstrate a matching instance. (1 mark)

1.2 Present all **unique subgraphs** in Figure 1 that are isomorphic to the graph in Figure 3. For example, $\{v_1 : 1, v_2 : 2, v_3 : 3\}$, $\{v_1 : 2, v_2 : 1, v_3 : 3\}$, and $\{v_1 : 3, v_2 : 1, v_3 : 2\}$ are all considered as the same subgraph 123. (2 marks)

Marking for Q1.1: 1 mark is given for the correct answer. 0 mark is given for all other cases.

Marking for Q1.2: 2 marks are given if the result subgraphs are correct, complete, and not redundant. Extra subgraphs and missing subgraphs will result in a loss of marks.

Q2 (5 marks)

2.1 Given an undirected graph as shown in Figure 4,

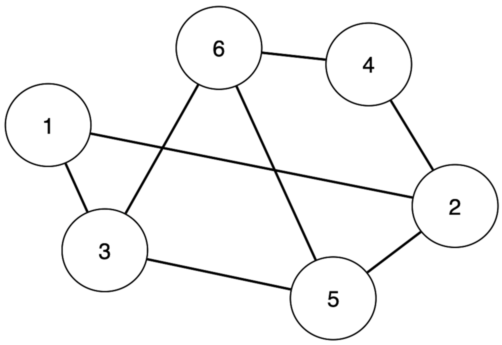


Figure 4

we aim to compute the output H^1 of the first graph convolutional layer with self-loops using the Graph Attention Network (GAT) model. The goal is to transform the initial node embeddings from a dimension of 4 to a dimension of 5 through this layer. The equation can be written as:

$$h_v^{(l)} = \sigma\left(\sum_{u \in N(v)\cup\{v\}} \alpha_{vu} W^{(l)} h_u^{(l-1)}\right),$$

where h_v^l indicates the d_l-dimensional embedding of node v in layer l, and $H^l = [h_{v1}^l, h_{v2}^l, h_{v3}^l, h_{v4}^l, h_{v5}^l, h_{v6}^l]^T$. $a_{vu} = \frac{1}{|N(v)\cup\{v\}|}$ is the weighting factor of node u's message to node v. $W^{(l)} \in R^{d_l * d_{l-1}}$ denotes the weight matrix for the neighbours of v in layer l, d_l denotes the dimension of the node embedding in layer l. $\sigma(\cdot)$ denotes the $ReLU$ non-linear function. The initial embedding for all nodes is stacked in H^0. W^1 is the weight matrices. Self-loops are included in the calculation to ensure that the node's information is retained. Therefore, the term v is added to its set of neighbors, which can be expressed as $\{v\} \cup N(v)$. Round the values to 2 decimal places (for example, 3.333 will be rounded to 3.33 and 3.7571 will be rounded to 3.76). (3 marks)

$$H^0 = \begin{pmatrix} 0.30 & -0.60 & 0.10 & -0.20 \\ 0.60 & 0.40 & 0.40 & -0.10 \\ 0.20 & 0.70 & -0.40 & 0.50 \\ -0.40 & 0.60 & 0.10 & 0.80 \\ 0.40 & 0.90 & -0.20 & 0.10 \\ 0.30 & -0.30 & 0.90 & 0.70 \end{pmatrix} \quad W^1 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ 1 & 0 \\ 1 & 1 \end{pmatrix}$$

2.2 Please determine whether the following statements are TRUE or FALSE. (2 marks)

a. Skip-connections is a good technique used to alleviate over-smoothing.
b. To design a model for predicting dropout on a course website, we represent it as a bipartite graph where nodes indicate students or courses. The task here is considered as node classification.
c. Graph Attention Network (GAT) has less expressive power compared to GCN, as it computes the attention score between each pair of neighbors, which introduces extra computational complexity.
d. The main difference between GraphSAGE and GCN is that GraphSAGE needs an activation function to add nonlinearity.

Marking for Q2.1: 3 marks are given for the correct result. Incorrect values will result in a deduction of marks. Providing a **detailed** and **correct** description of the calculation will earn marks for a valid attempt, even if there are major mistakes in the result.

Marking for Q2.2: 0.5 mark is given for each correct TRUE/FALSE answer.

Q3 (8 marks)

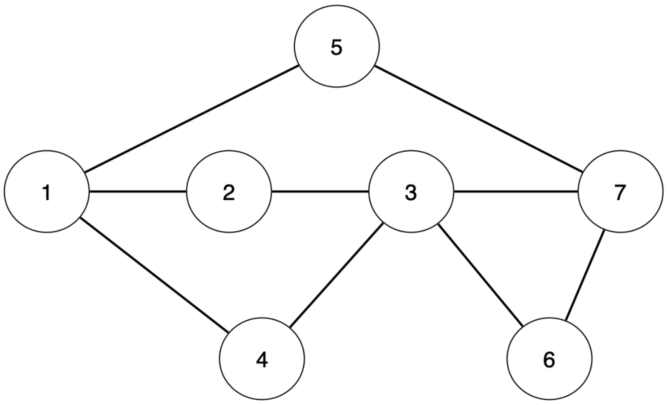


Figure 5

Suppose we aim to count the number of shortest paths from a source vertex to all other vertices in an undirected unweighted graph shown using Pregel.

3.1 Write the pseudocode for the compute implementation in Pregel. (3 marks)

3.2 Assume we run your algorithm with the source node 1 for the graph in Figure 5 on three workers. Vertices 1 and 5 are in worker X. Vertices 2 and 4 are in worker Y. Vertices 3, 6 and 7 are in worker Z. Please indicate the set of active vertices and how many messages are sent in **each** iteration. (3 marks)

3.3 Can the combiner optimization be used in this case? If yes, write the pseudocode for a combiner implementation. Calculate how many messages are sent in **each** iteration if the combiner is used in 3.2. If no, briefly discuss why a combiner cannot be used. (2 marks)

Marking for Q3.1: 3 marks are given for the correct answer. 0 mark is given for all other cases.

Marking for Q3.2: 2 marks are given for the correct answer. 0 mark is given for all other cases.

Marking for Q3.3: 3 marks are given for the correct answer. 0 mark is given for all other cases.

Q4 (4 marks)

Consider the graph in Figure 6,

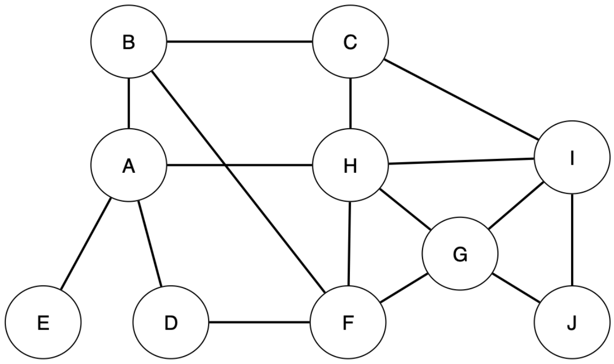


Figure 6

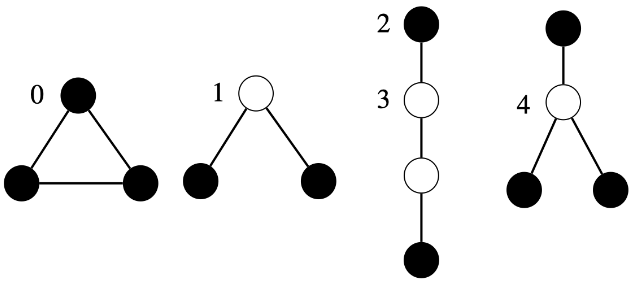


Figure 7

4.1 Compute the betweenness centrality and closeness centrality of nodes C and H in Figure 6. Round the values to 2 decimal places (for example, 3.333 will be rounded to 3.33 and 3.7571 will be rounded to 3.76). (2 marks)

4.2 Given the graphlets in Figure 7, derive the graphlet degree vector (GDV) for nodes A and G. Note that only the induced matching instances are considered in GDV. (2 marks)

Marking for Q4.1: 1 mark is given for correct betweenness centrality values. 1 mark is given for correct closeness centrality values.

Marking for Q4.2: 1 mark is given for each correct vector. Incorrect values in each vector will result in a deduction of marks.

END OF QUESTIONS