Part III

Advanced GNN Techniques

Lecture 07

GraphSAGE and Scalable Learning

- □ Understanding GraphSAGE for large-scale graphs.
- □ Neighbor sampling and aggregation techniques.
- □ Activity: Implement GraphSAGE for node classification on a large dataset.

Challenges and Limitations of PREVIOUS GNNs

$$h_i^{(k+1)} = \Psi\left(\sum_{j\in\aleph_i} \alpha_{ij}^{(k)} W^{(k)} h_i^{(k)}\right)$$

GCN

$$\alpha_{ij}^{(k)} = \frac{1}{\sqrt{Deg_{(i)}^{(k)}}\sqrt{Deg_{(j)}^{(k)}}}$$

- GCNs aim to tackle the issues of vanilla GNNs through **normalization**.
- Attention Scores are Explicitly Fixed in GCNs.

GAT

$$\alpha_{ij}^{(k)} = SoftMax\left(e_{ij}^{(k)}\right)$$

- GATs use dynamic weighting through self-attention instead of fixed normalization coefficients.

Challenges and Limitations of GATs

Computational Complexity:

GATs, especially with multiple attention heads, can be computationally expensive due to the need for pairwise attention scores between nodes.

Sensitivity to Graph Structure:

GATs may be sensitive to changes in the graph structure, and their performance can be affected when applied to graphs with varying sizes or irregular structures.

Attention Mechanism Challenges:

While attention mechanisms in GATs are beneficial, designing an effective attention mechanism that captures complex relationships in diverse graphs can be challenging.

Transductive Nature:

GATs are transductive, meaning they cannot generalize to unseen nodes during training. When the graph evolves, and new nodes are introduced, retraining the entire model becomes necessary.

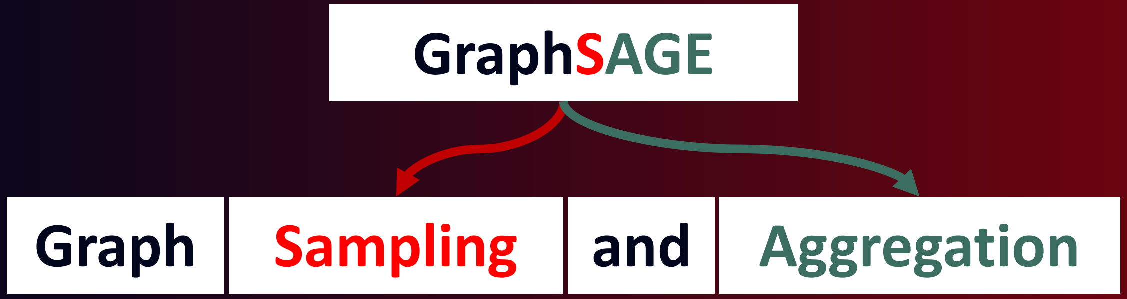
Challenges and Limitations of GATs

GraphSAGE's Resolution:		
	Inductive Learning:	A learning approach, enabling generalization to unseen nodes without the need for full retraining. It allows to efficiently generate representations for nodes not present during training.
	Efficient Handling of Graph Evolution:	Well-suited for ever-evolving graphs, addressing the inefficiency of transductive approaches.
	Overcoming Scalability Challenges:	**GraphSAGE** is suitable for scalable learning on diverse and evolving graph structures.

Introducing GraphSAGE

Introduced by Hamilton et al. in 2017 as a framework for inductive representation learning on large graphs.

GraphSAGE addresses the challenge of scaling to large graphs, particularly those with over 100,000 nodes.



The working process of **GraphSage** is mainly divided into two steps, the first is performing **neighbourhood sampling** of an input graph and the second one **learning aggregation functions** at each search depth.

Hamilton, W., Ying, Z., & Leskovec, J. (2017). Inductive representation learning on large graphs. Advances in neural information processing systems, 30.

Introducing GraphSAGE

01 Neighbourhood Sampling

02 Learning Aggregation Functions

Traditional neural networks

Introduction to Mini-Batching:

- Essential in traditional neural networks.
- involves dividing the dataset into batches.

Types of Gradient Descent:

- **Batch GD:** Updates after processing the entire dataset, slow and memory-intensive.
- **Stochastic GD (SGD):** Updates for each training example, noisy but suitable for online training.
- **Mini-Batch GD:** Updates at the end of mini-batches, faster with stable convergence. Crucial for large graphs.

Advantages of Mini-Batch GD:

- Faster processing with GPU parallelization.
- More stable convergence.
- Essential for handling large graphs beyond available memory.

Tabular datasets:

Straightforward, involves selecting n samples (rows).

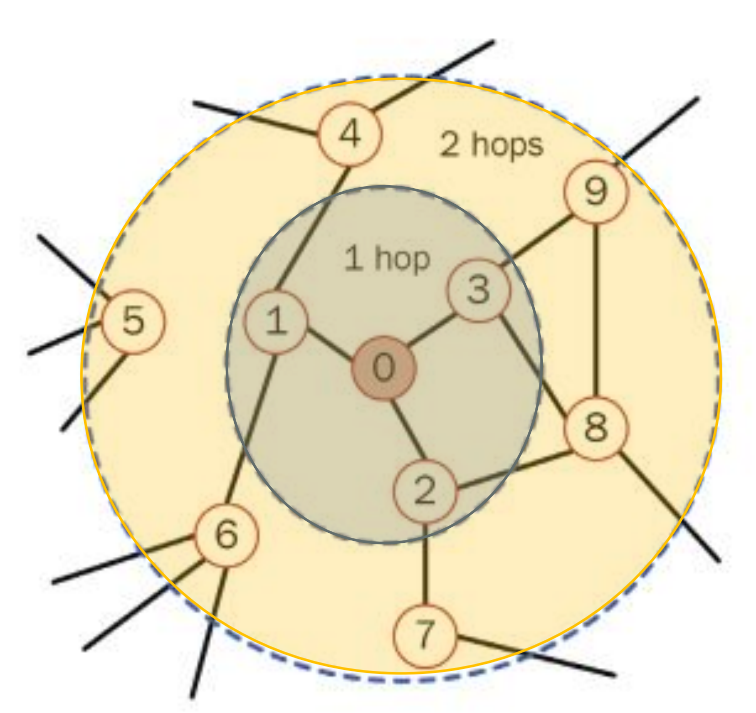
DR. B. KHALDI
Assoc. Prof.
email: b.khaldi@esi-sba.dz

Introducing GraphSAGE

For GNN

01 Neighbourhood Sampling

02 Learning Aggregation Functions



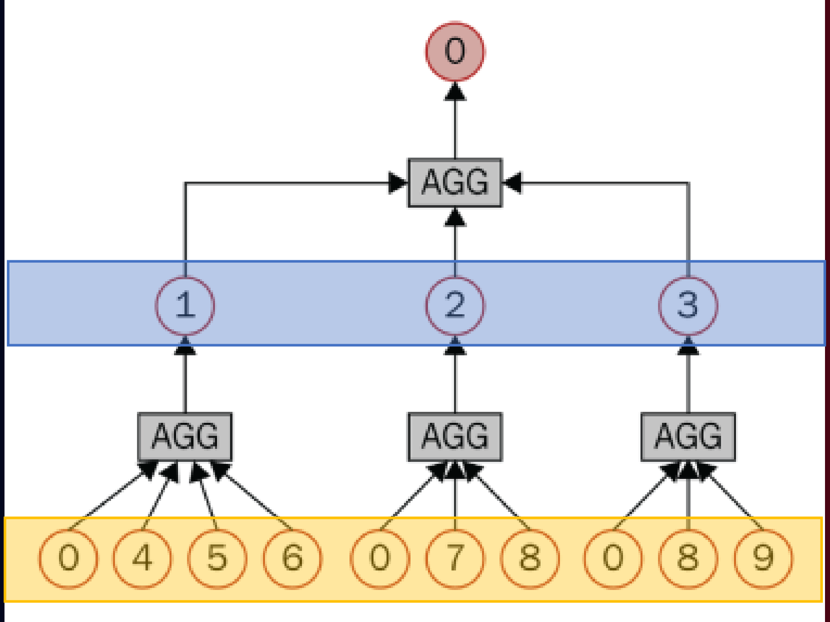
Graph Dataset Division:	**Graph datasets:** Challenge due to essential connections; selecting nodes requires careful consideration.
GNN's Perspective on Datasets:	**GNN layers** compute node embeddings based on their neighbors. Each GNN layer considers a specific **hop distance** (1 hop, 2 hops, etc.).
Importance of Direct Neighbors:	Computing an embedding requires only the **direct neighbors** (1 hop) in **GNN layers**. Additional layers involve extending to neighbors' neighbors (2 hops) and beyond.

Introducing GraphSAGE

For GNN

01 Neighbourhood Sampling

02 Learning Aggregation Functions



Graph Dataset Division:

Graph datasets: Challenge due to essential connections; selecting nodes requires careful consideration.

GNN's Perspective on Datasets:

GNN layers compute node embeddings based on their neighbors.

Each GNN layer considers a specific **hop distance** (1 hop, 2 hops, etc.).

Importance of Direct Neighbors:

Computing an embedding requires only the **direct neighbors** (1 hop) in **GNN layers**.

Additional layers involve extending to neighbors' neighbors (2 hops) and beyond.

Introducing GraphSAGE

For GNN

01 Neighbourhood Sampling

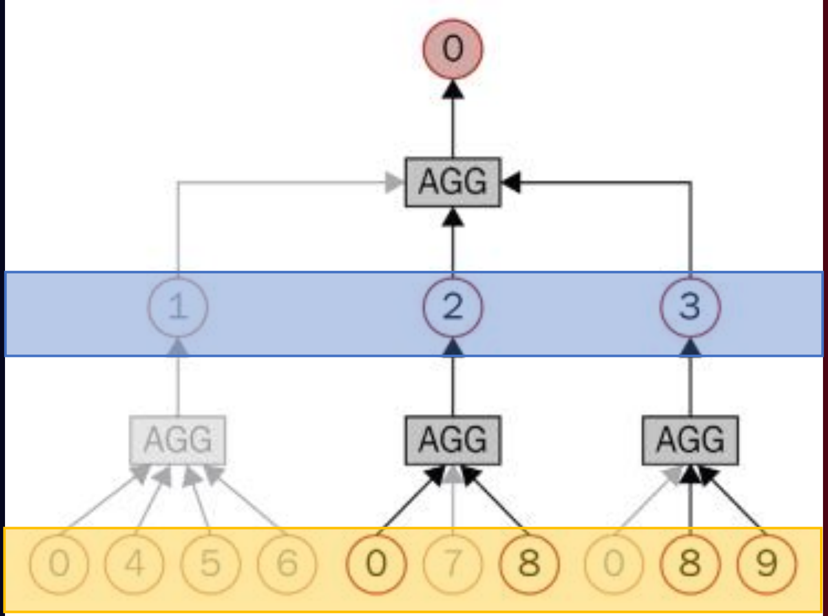
02 Learning Aggregation Functions

Challenges:

- **Exponential Growth:**
  - **Issue:** Computation graph size grows exponentially.
- **High-Degree Nodes Issue:**
  - **Problem:** Nodes with high degrees create large graphs.

GraphSAGE Solution:

- **Neighbor Sampling:**
  - **Technique:** Limits graph size by sampling neighbors.
  - **Example:** Keep a predefined number of neighbors per hop.
  - **Result:** Controls size, preventing exponential growth.



Example: we choose only to keep (at most) 3 neighbors during **the first hop** and **5** neighbors during **the second hop**. Hence, the computation graph cannot exceed **3 × 5 = 15** nodes in this case.

Introducing GraphSAGE

01 Neighbourhood Sampling

02 Learning Aggregation Functions

Sampling Techniques in GraphSAGE

Efficiency vs. Variance:	**Low sampling number:** More efficiency, but makes the training more random (higher variance)
	Balancing Act: Choose sampling number based on efficiency and training variance.
Managing GNN Layers:	Keep GNN layers (hops) low to avoid exponential computation graphs.
	Challenge: Balancing layer depth for efficiency and accuracy.
Neighbor Sampling Trade-Off:	**Efficient Handling:** Neighbor sampling manages large graphs.
	Trade-Off: Prunes information, impacting performance, e.g., accuracy.
Alternative Techniques - PinSAGE:	**Pinterest's Approach:** Implements random walks for sampling.
	Determines importance based on frequency.
	Efficiency: Proven more efficient by selecting critical nodes.

Introducing GraphSAGE

Aggregation Techniques in GraphSAGE

01 Neighbourhood Sampling

02 Learning Aggregation Functions

Aggregation Operator:

Essential for computing embeddings after selecting neighboring nodes.

GraphSAGE Proposed Aggregators:

Mean Aggregator: Averages embeddings of target nodes and sampled neighbors.

LSTM Aggregator: Based on LSTM architecture, capable of discriminating more graph structures.

Pooling Aggregator: Employs MLP and max operation for feature selection.

Introducing GraphSAGE

Aggregation Techniques in GraphSAGE

01 Neighbourhood Sampling

02 Learning Aggregation Functions

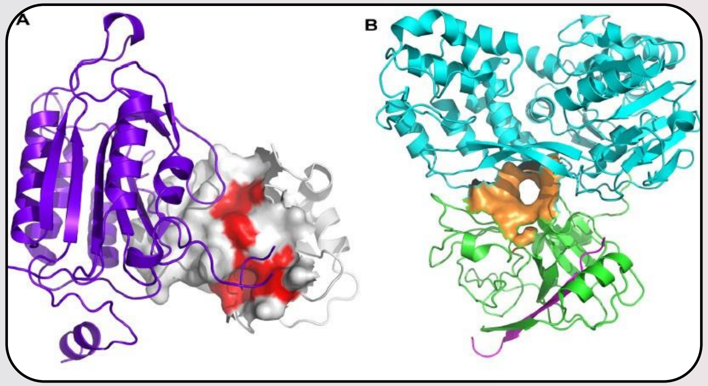
Global Transition Function (Message Passing)

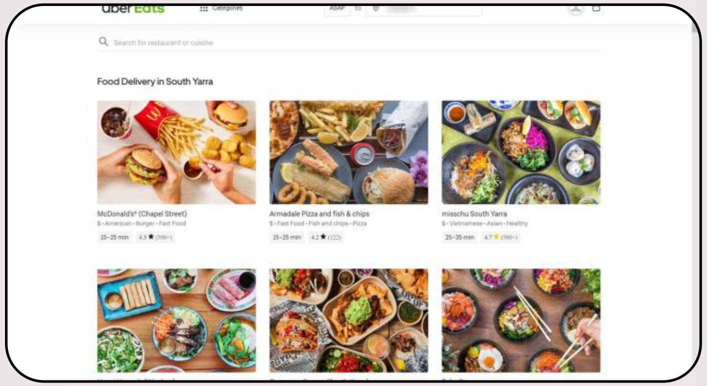
Mean

$$h_i^{(k+1)} = \Psi\left(\sum_{j\in\aleph_i} \frac{1}{|\aleph_i|} W^{(k)} h_i^{(k)}\right)$$

CHALLENGES & LIMITATIONS
GRAPHSAGE

Introducing GraphSAGE




Uber Eats
Categories
ASAP
to
Search for restaurant or cuisine
Food Delivery in South Yarra
McDonald's® (Chapel Street)
$ • American • Burger • Fast Food
15–25 min
4.3 ★ (500+)
Armadale Pizza and fish & chips
$ • Fast Food • Fish and chips • Pizza
15–25 min
4.2 ★ (122)
misschu South Yarra
$ • Vietnamese • Asian • Healthy
25–35 min
4.7 ★ (500+)

Protein-Protein Interaction Graphs:

- Predicts embeddings for **Protein-Protein** interaction graphs.
- Enables efficient prediction of embeddings for new **species/organisms**.

UberEats:

- Employs **GraphSAGE** in its **recommendation system**.
- Chosen for inductive capability and scalability to billions of nodes.
- Recommends dishes, restaurants, and cuisines based on user preferences.

Pinterest:

- Utilizes **PinSage**, a **GraphSAGE** variant, for **visual recommendations**.
- **PinSage** handles billions of nodes in web-scale graphs.
- Powers recommendations for visual bookmarks (**pins**) related to shopping.

THANK YOU