

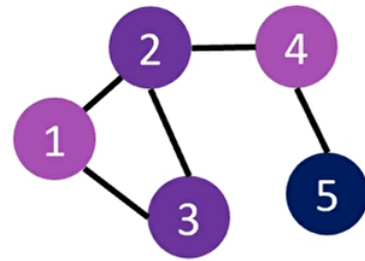
Graph Neural Network

Sadeq Elfergany

Faculty of computers and AI, BSU University

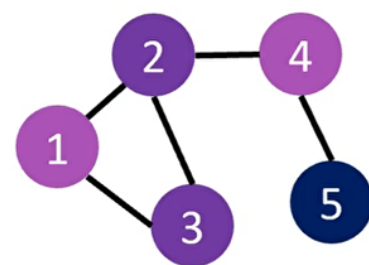
a.k.sadeq@eng.bsu.edu.eg

A Simple Graph



$$G = (V, E)$$

A Simple Graph

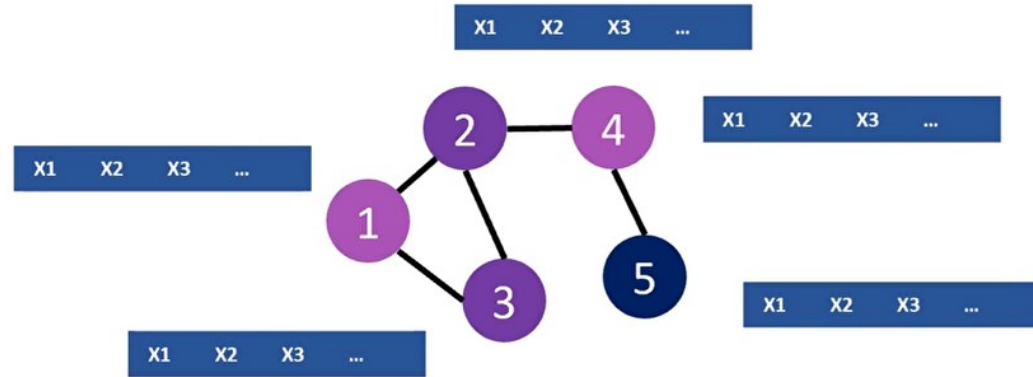


$$G = (V,E)$$

	v1	v2	...
v1	0	1	...
v2	1	0	...
v3	1	1	...
...

Adjacency matrix
 $V \times V$

A Simple Graph

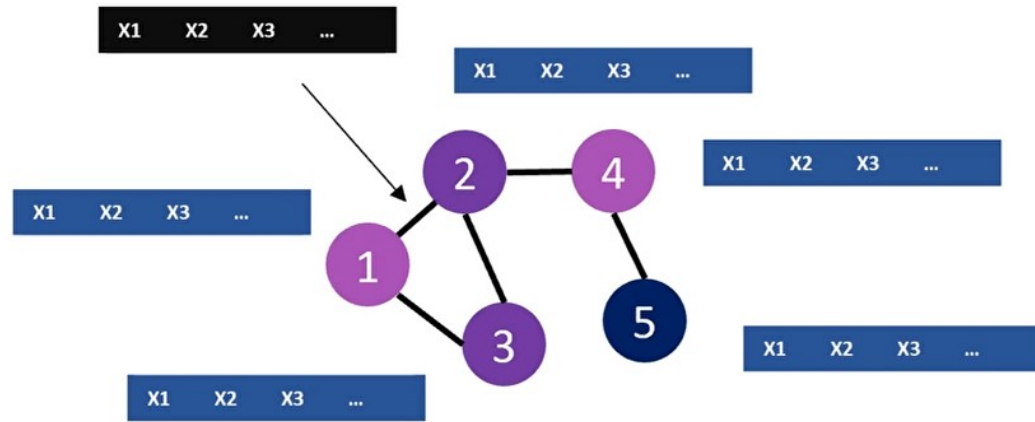


$G = (V, E)$

	V1	V2	...
V1	0	1	...
V2	1	0	...
V3	1	1	...
...

Adjacency matrix
 $V \times V$

A Simple Graph



$G = (V, E)$

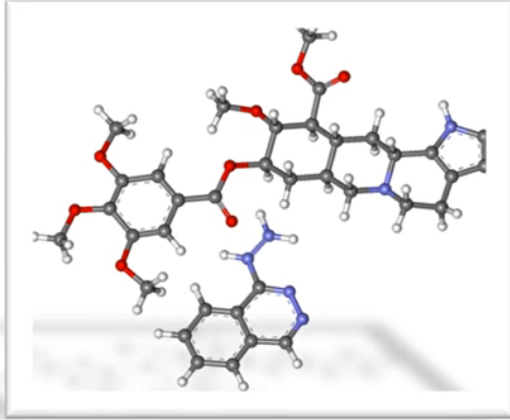
	V1	V2	...
V1	0	1	...
V2	1	0	...
V3	1	1	...
...

Adjacency matrix
 $V \times V$

Graph Data is everywhere

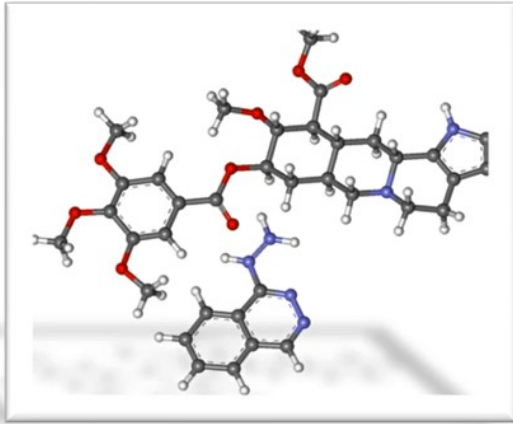


Graph Data is everywhere



Medicine / Pharmacy

Graph Data is everywhere

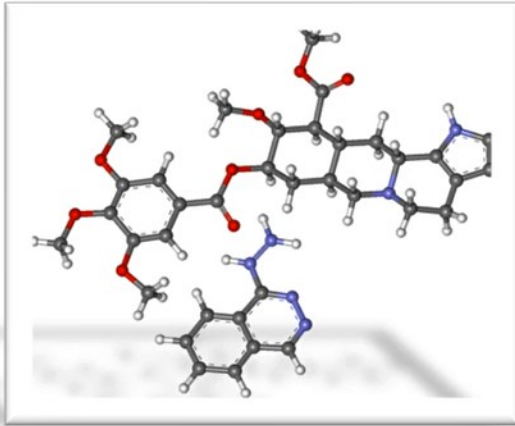


Medicine / Pharmacy



Recommender Systems

Graph Data is everywhere



Medicine / Pharmacy

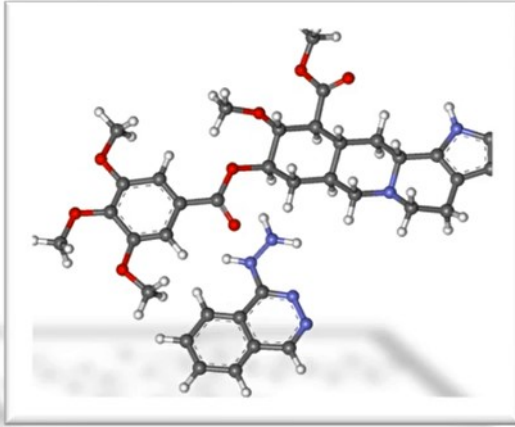


Recommender Systems



Social Networks

Graph Data is everywhere



Medicine / Pharmacy



Recommender Systems



Social Networks

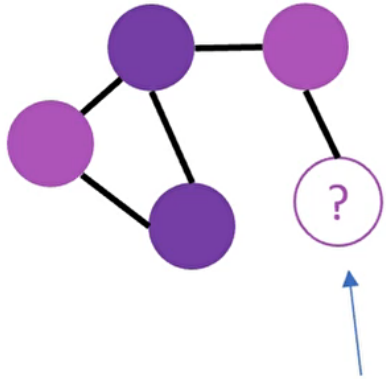


3D Games / Meshes

Examples for Machine Learning Problems with Graph Data

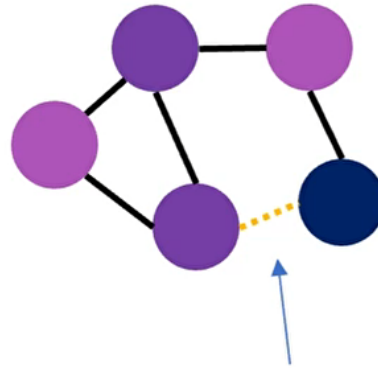
Examples for Machine Learning Problems with Graph Data

Node-level predictions



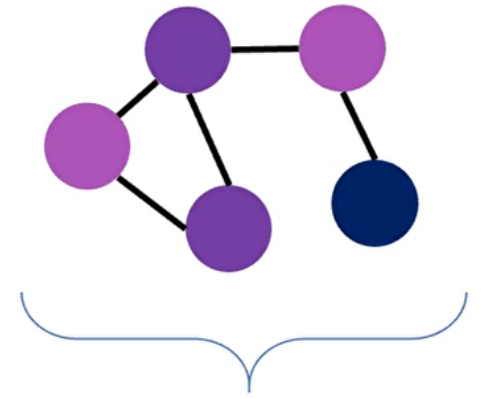
Does this person smoke?
(unlabeled node)

Edge-level predictions (Link prediction)



Next Netflix video?

Graph-level predictions



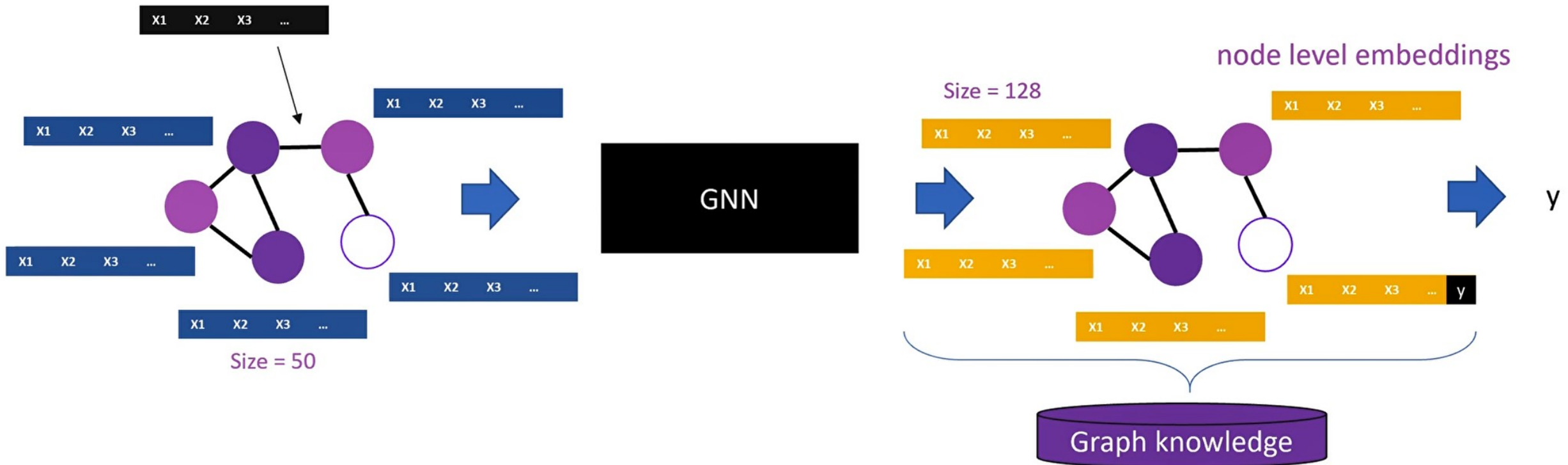
Is this molecule a suitable drug?

Fundamental Idea of GNNs

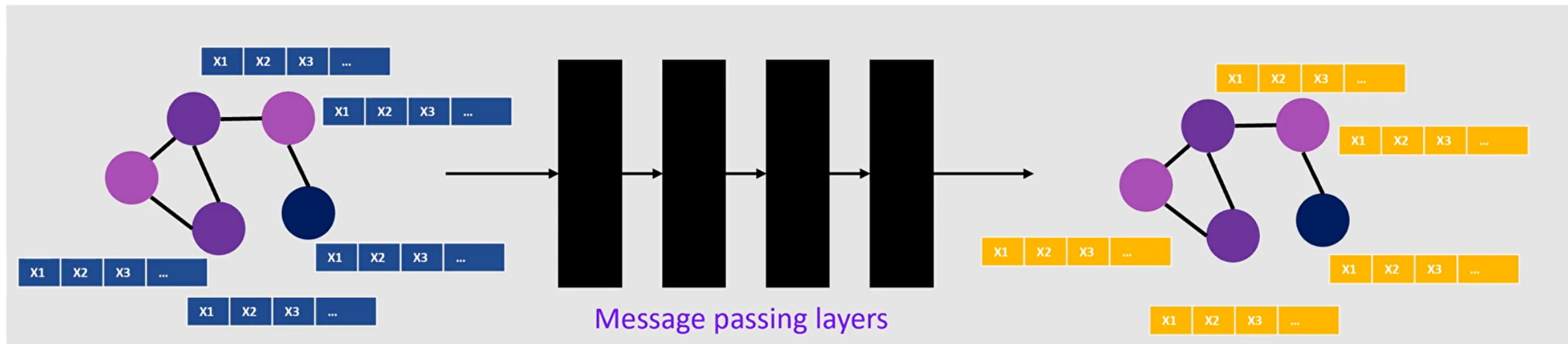


>> Learning a for neural networks suitable representation of graph data. <<

= Representation learning



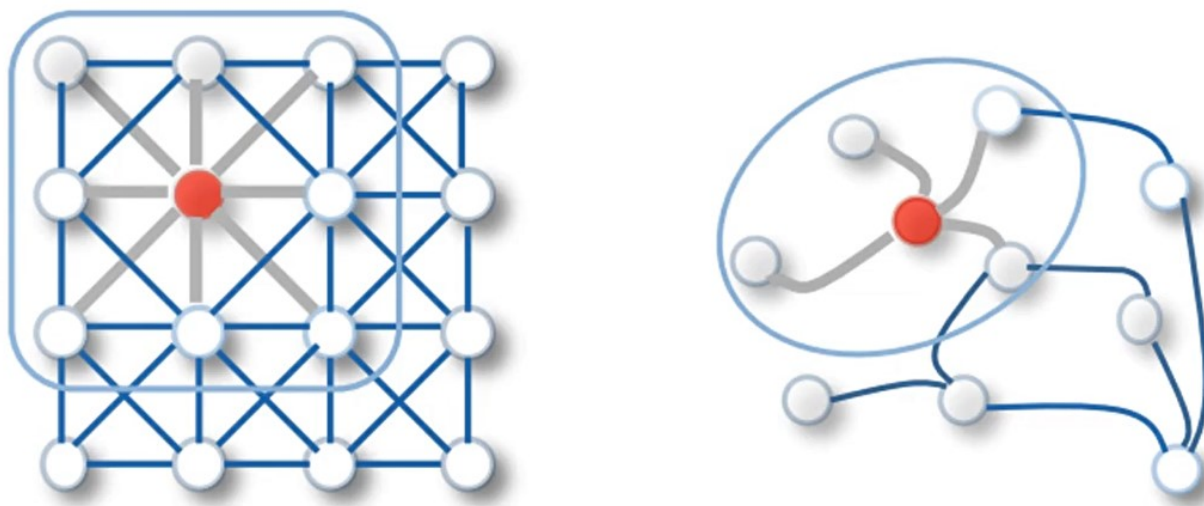
How do Graph Neural Networks work?



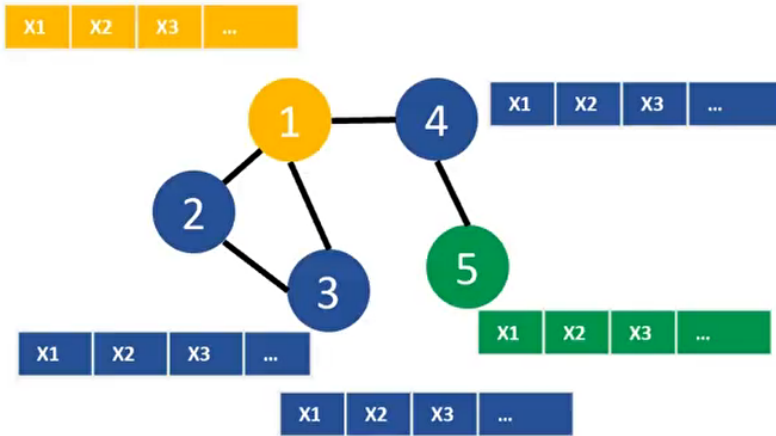
What is happening in the Message Passing Layers?



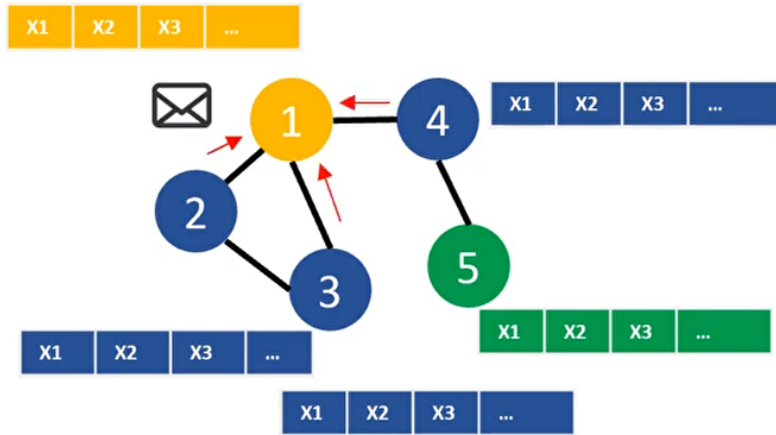
What is happening in the Message Passing Layers?



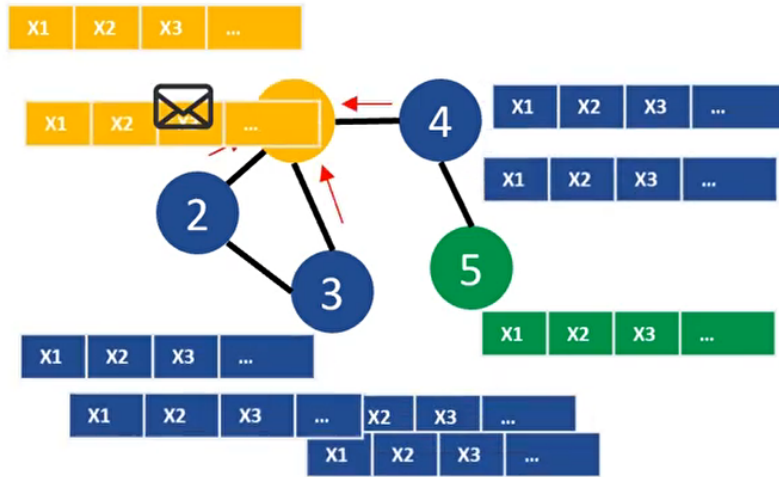
What is happening in the Message Passing Layers?



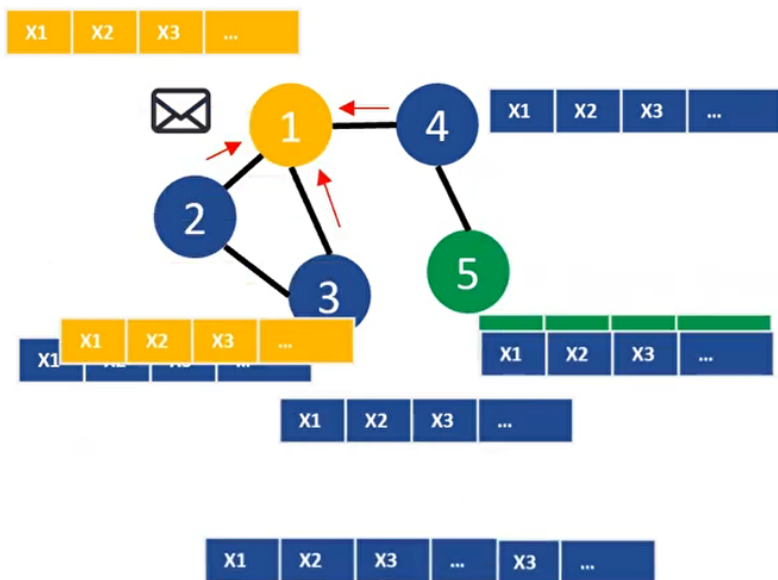
What is happening in the Message Passing Layers?



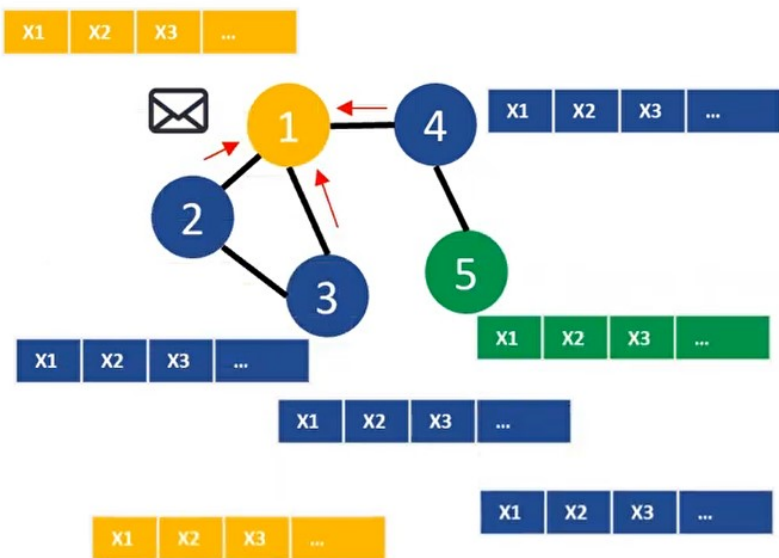
What is happening in the Message Passing Layers?



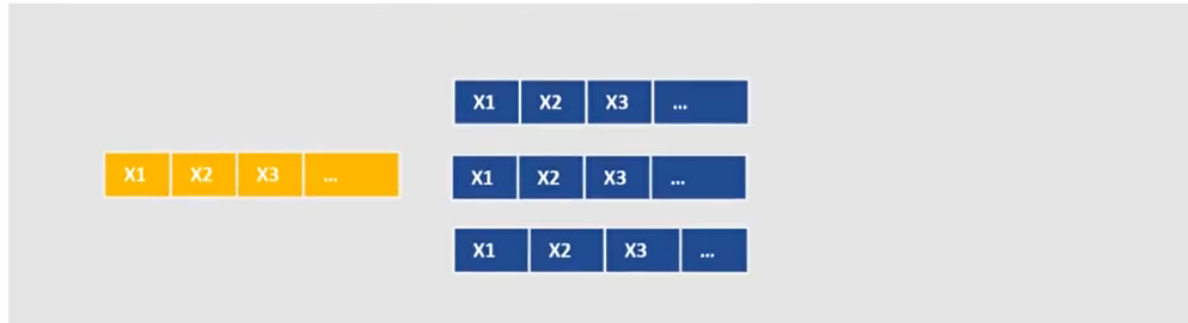
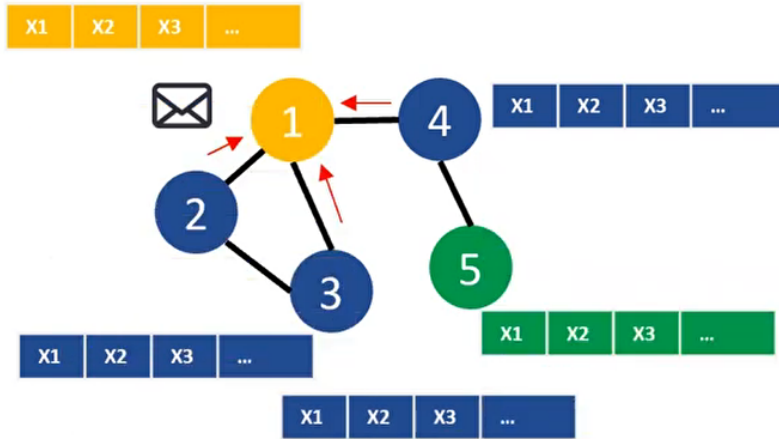
What is happening in the Message Passing Layers?



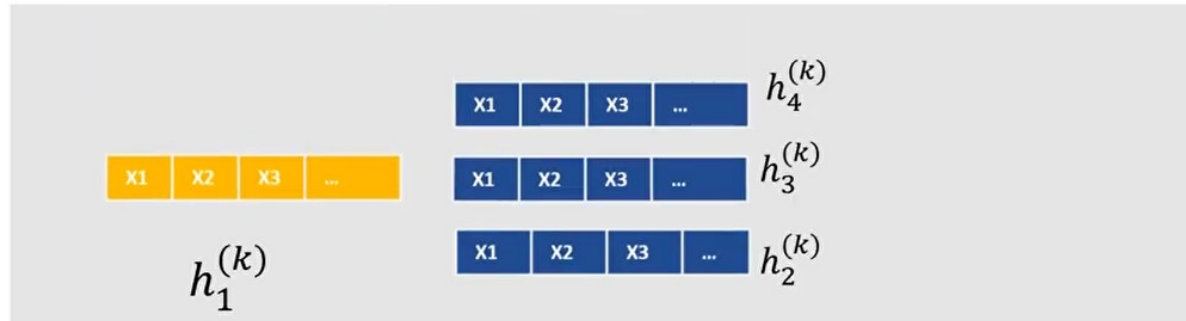
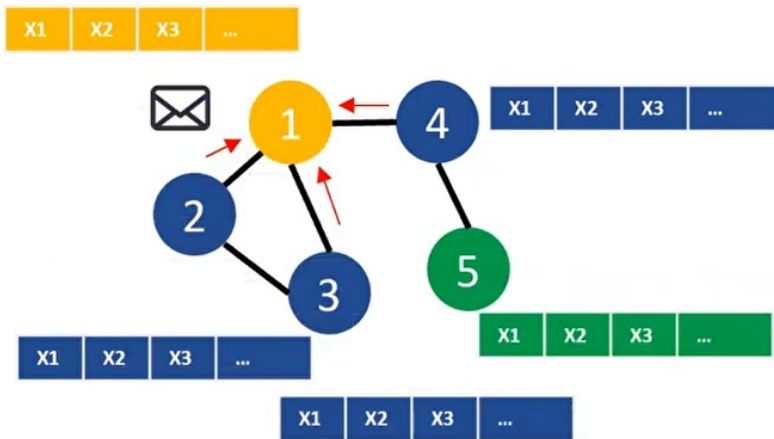
What is happening in the Message Passing Layers?



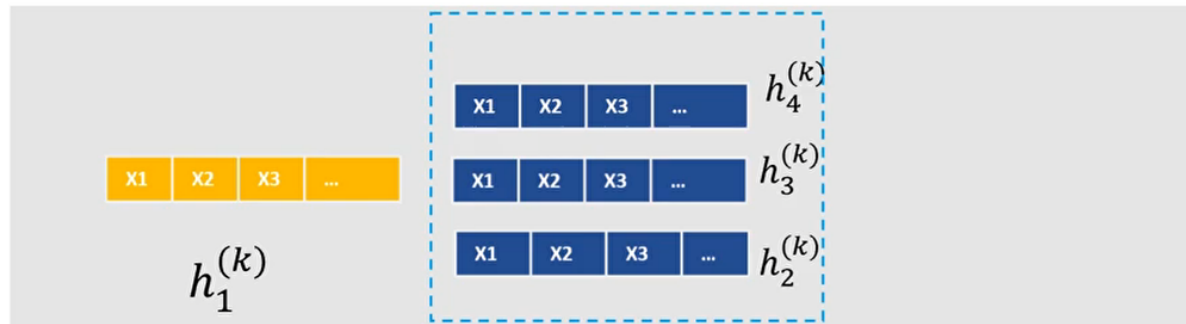
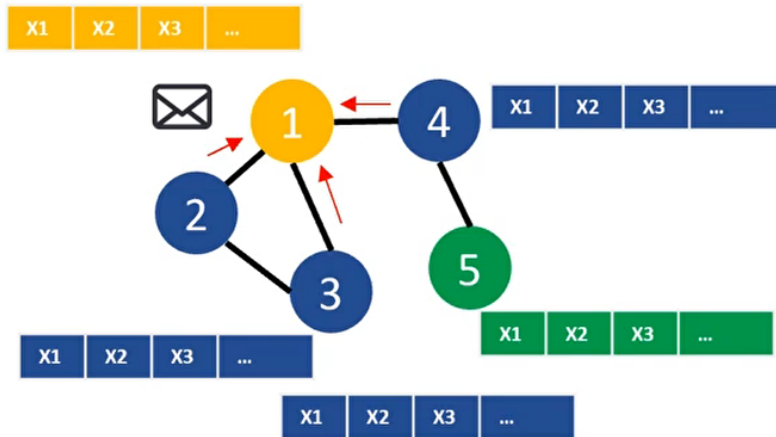
What is happening in the Message Passing Layers?



What is happening in the Message Passing Layers?

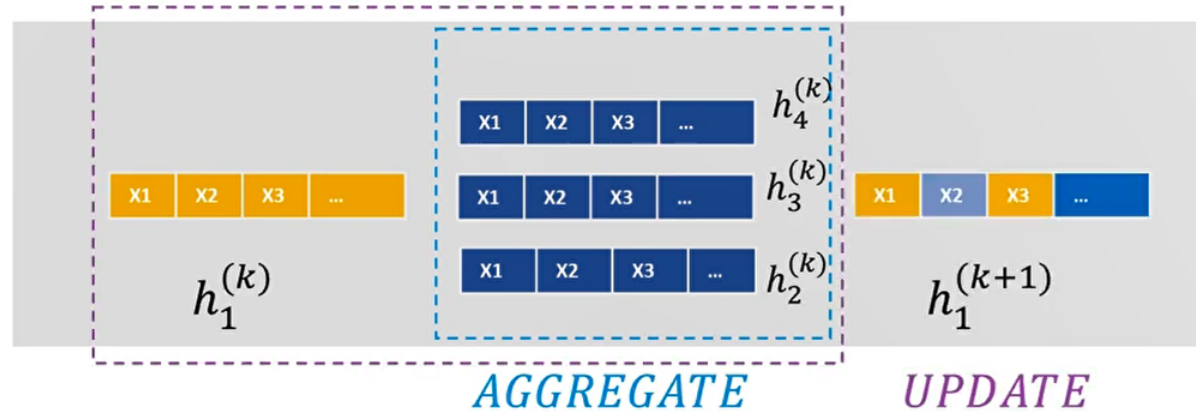
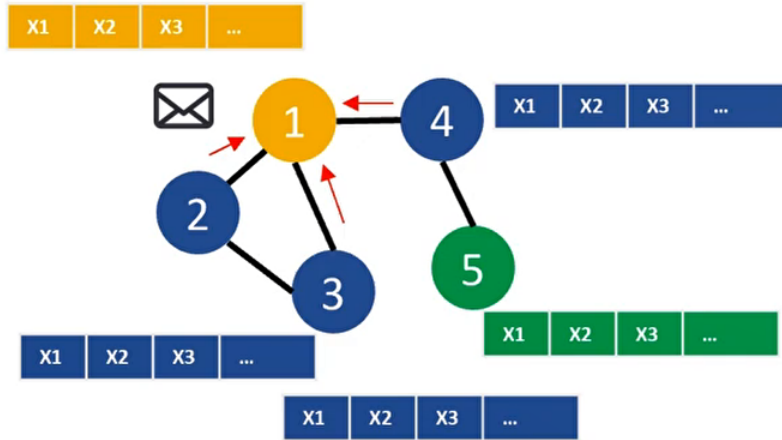


What is happening in the Message Passing Layers?

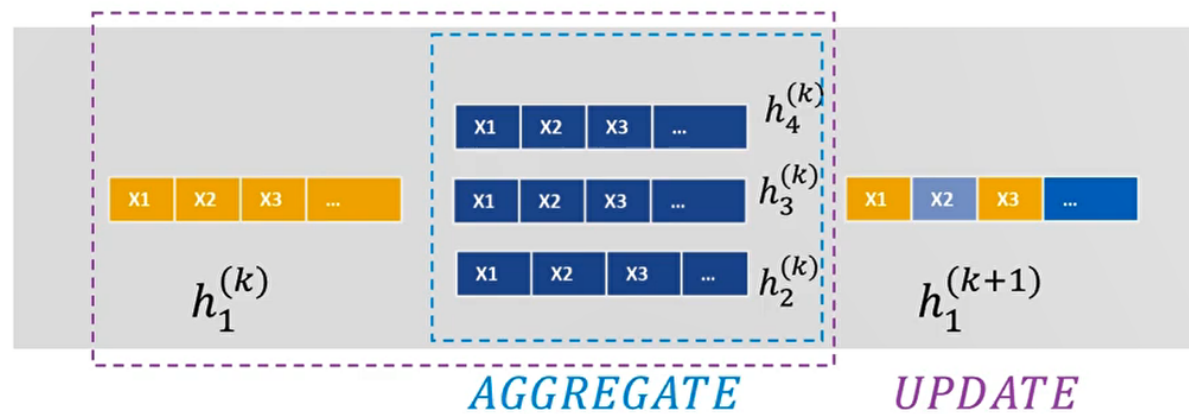
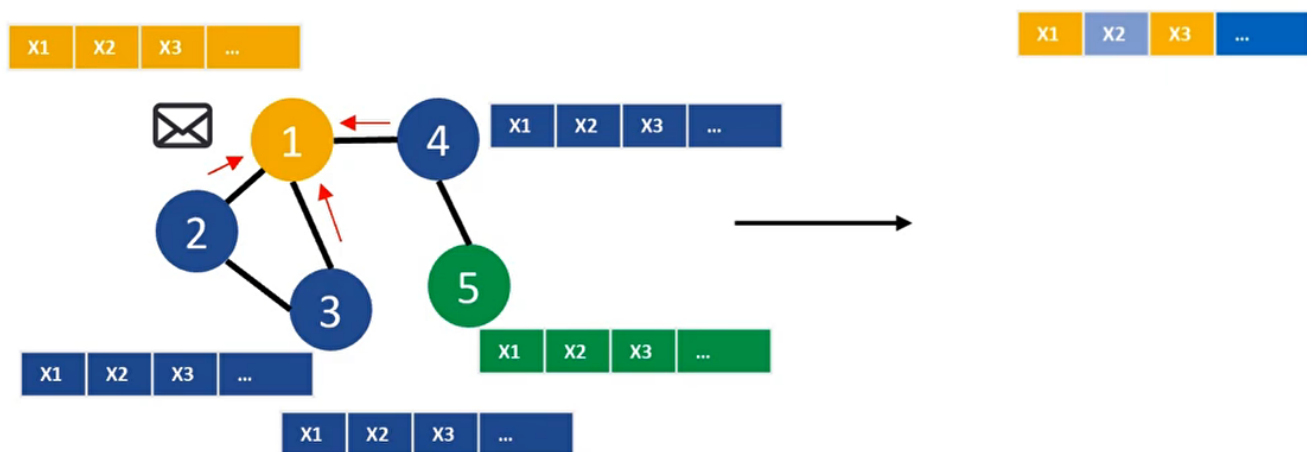


AGGREGATE

What is happening in the Message Passing Layers?

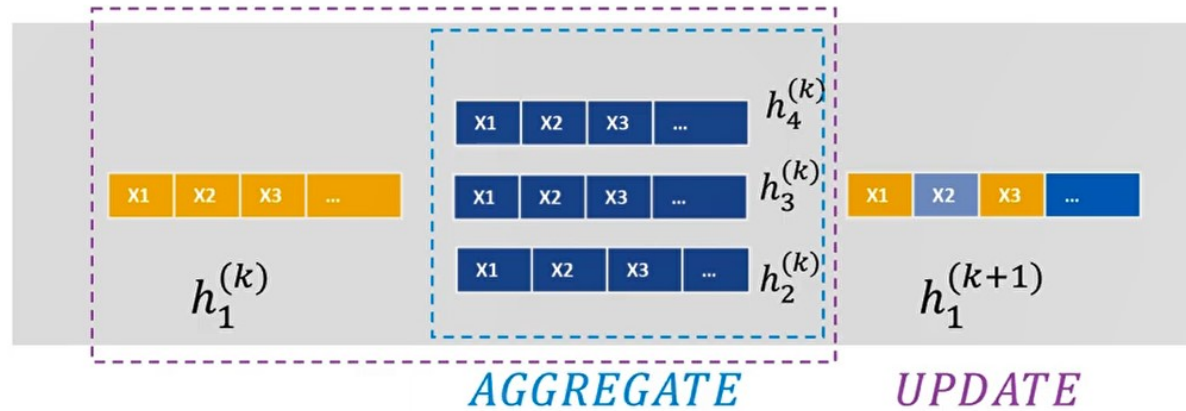
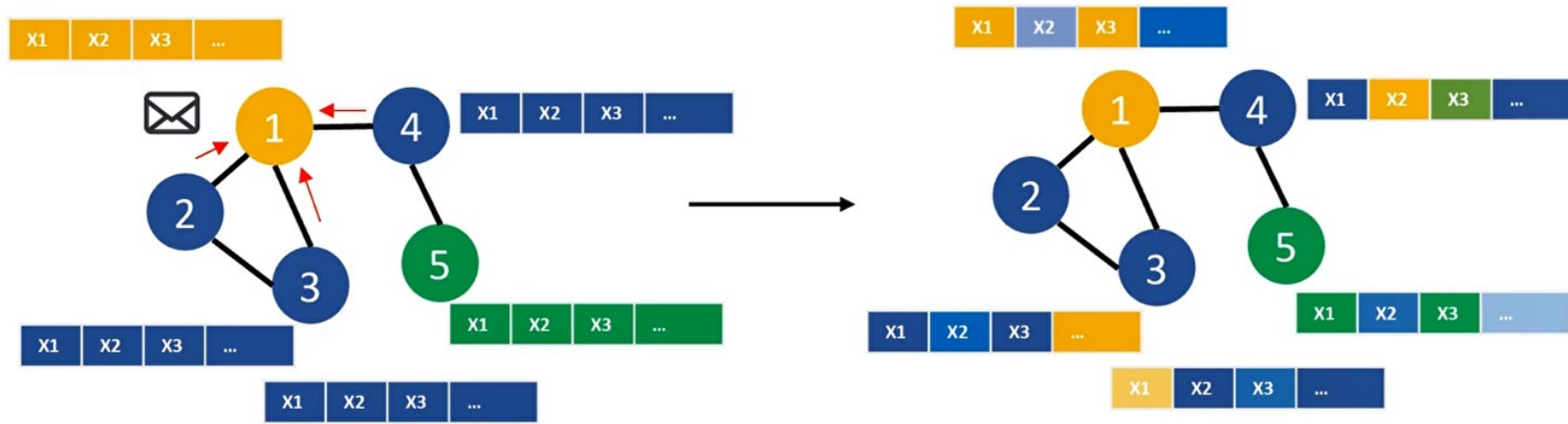


What is happening in the Message Passing Layers?

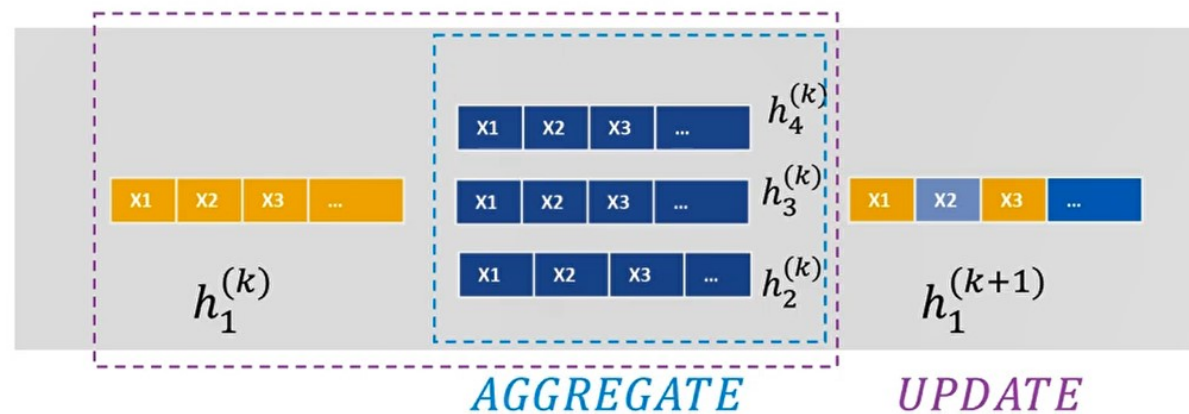
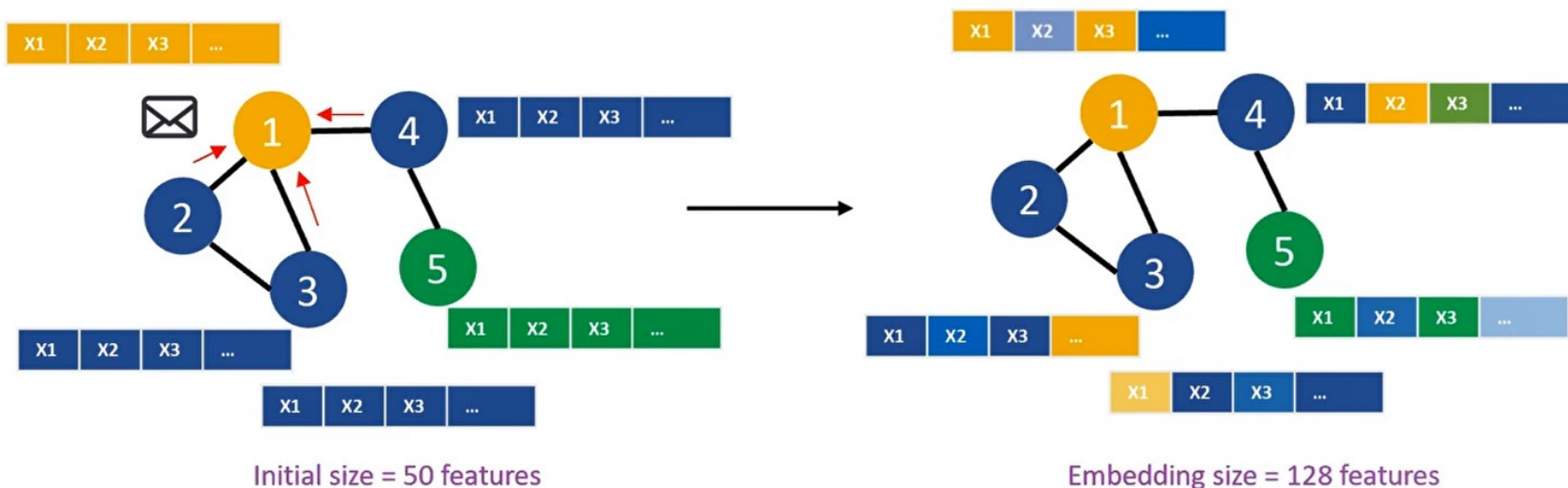




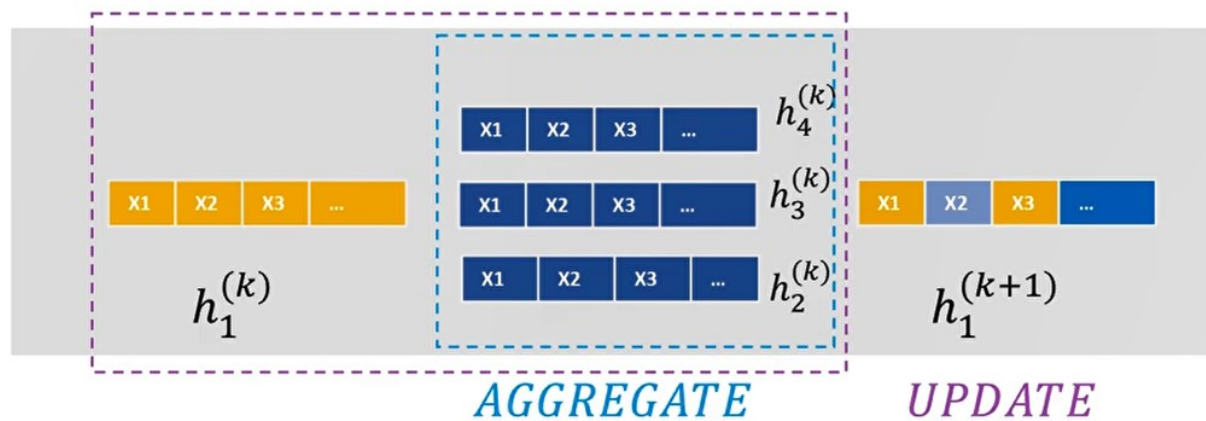
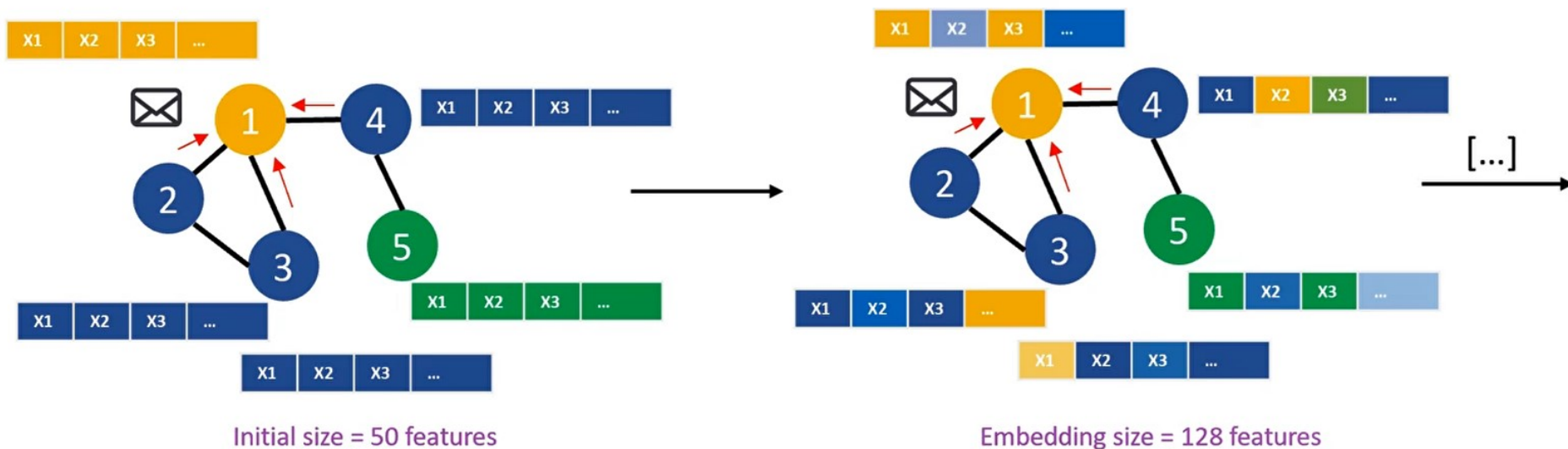
What is happening in the Message Passing Layers?



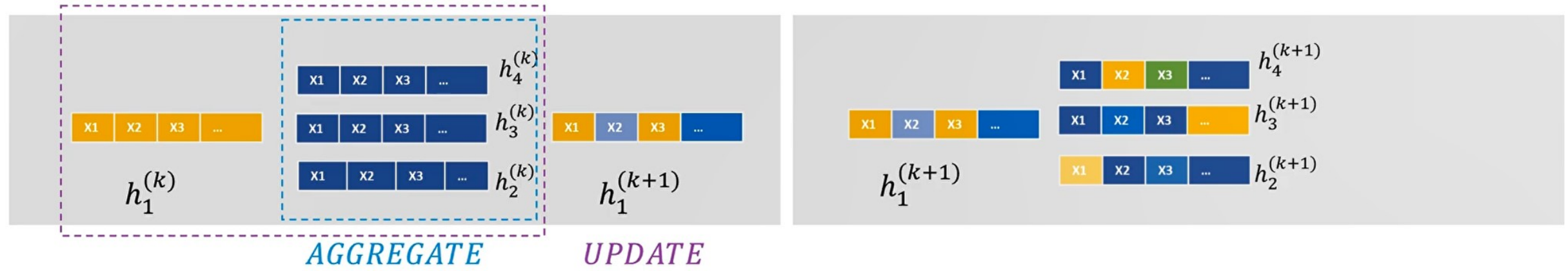
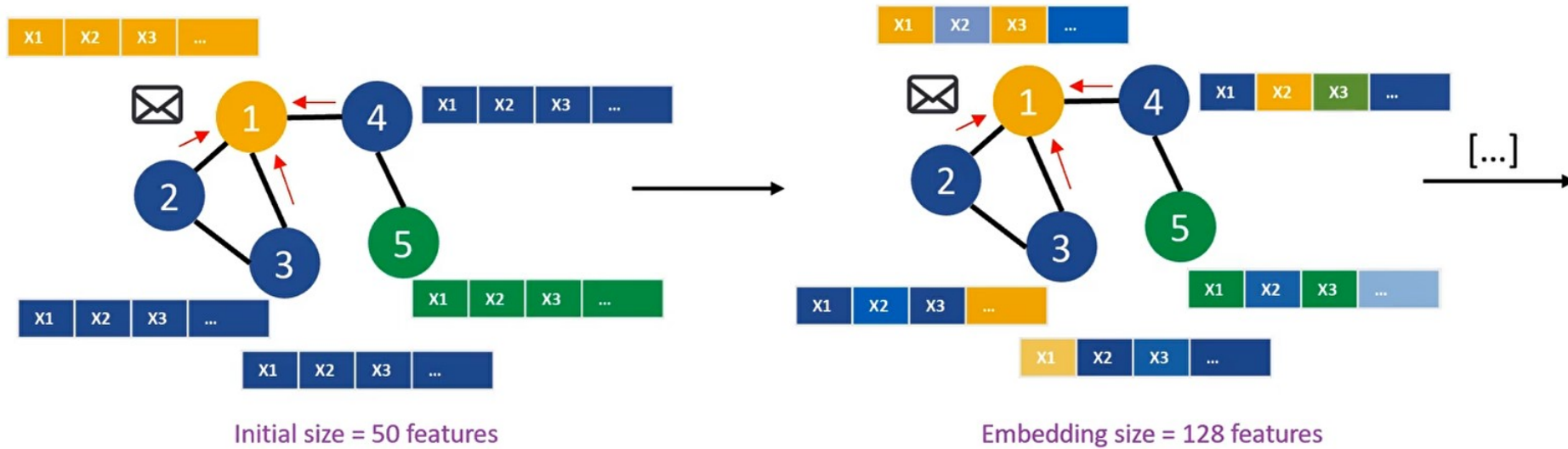
What is happening in the Message Passing Layers?



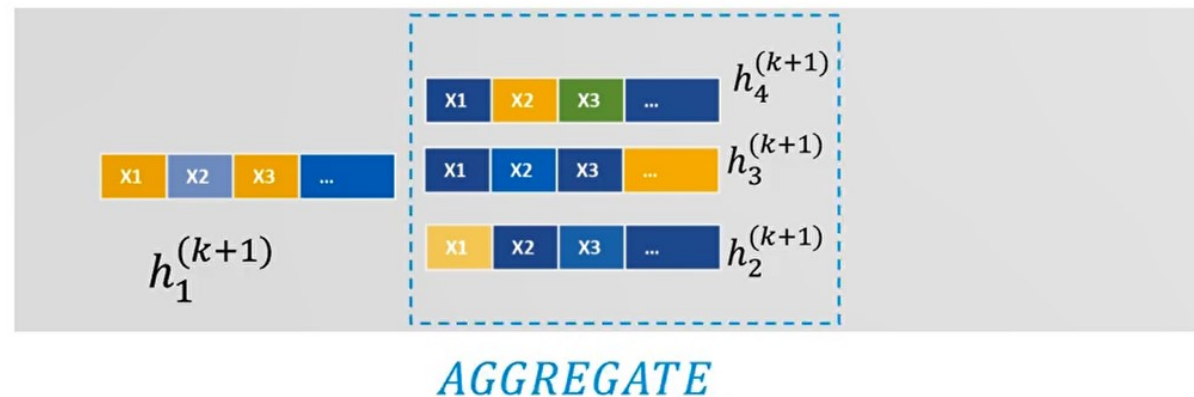
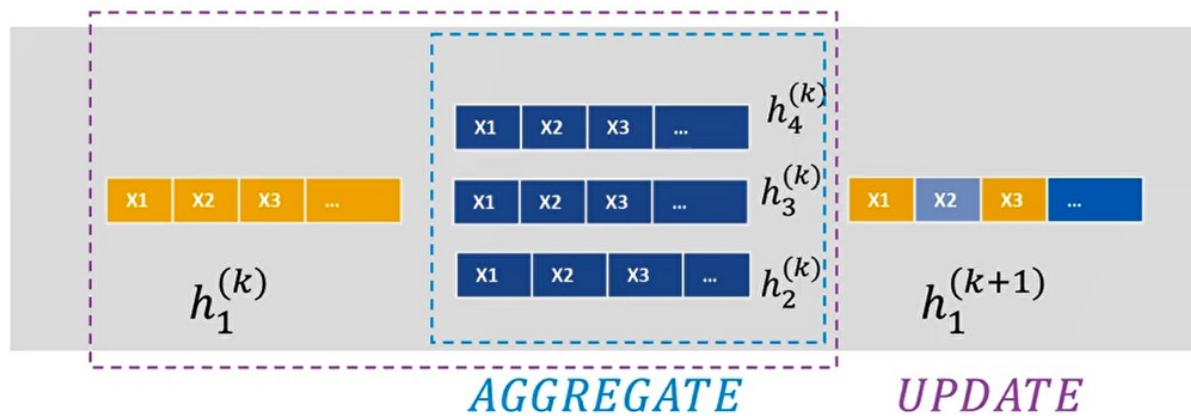
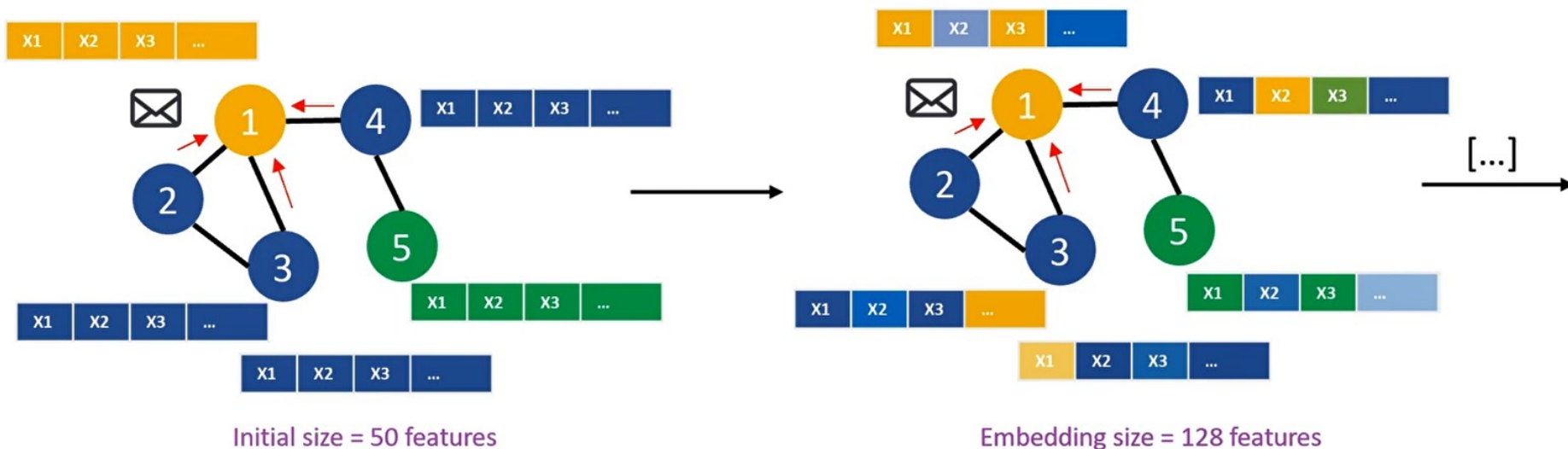
What is happening in the Message Passing Layers?



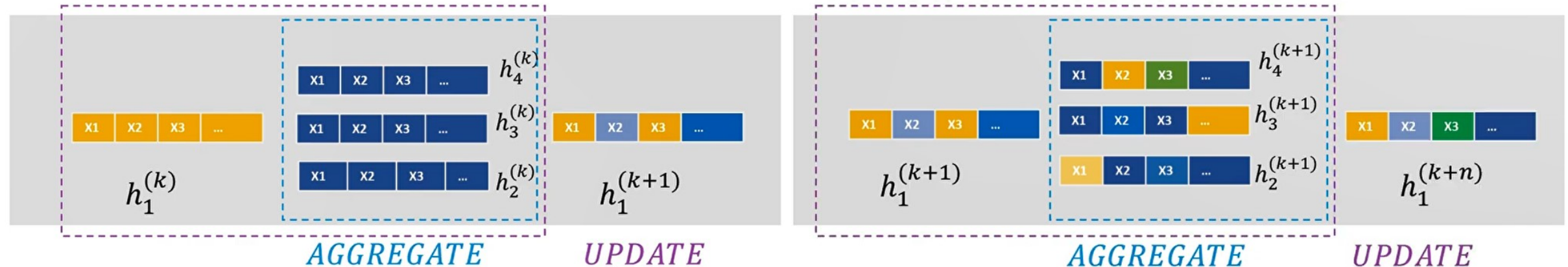
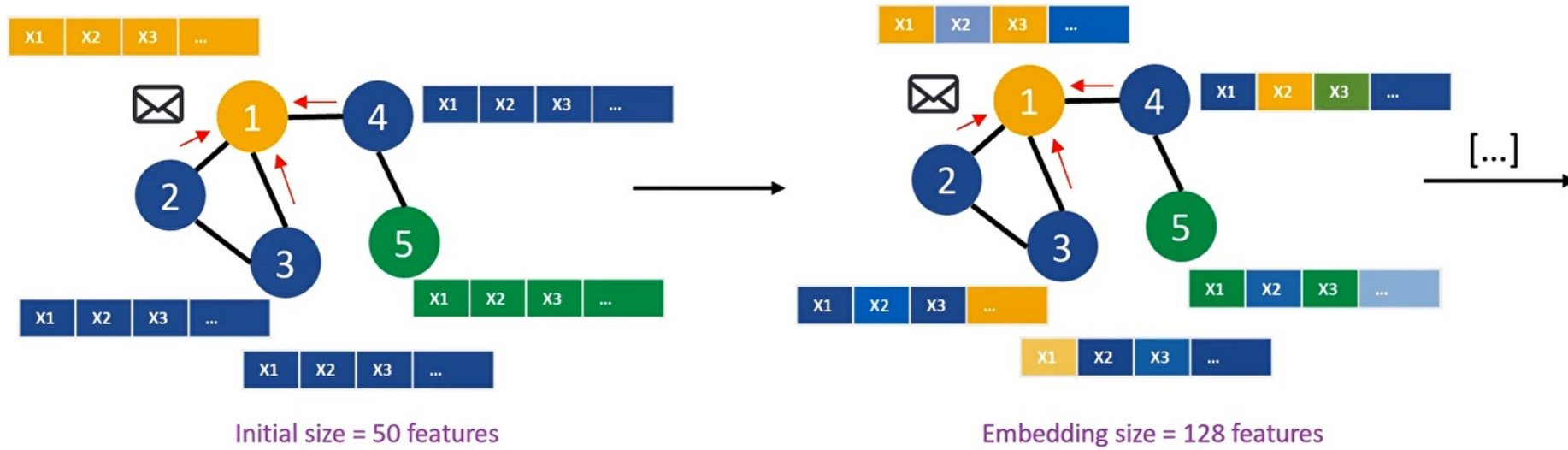
What is happening in the Message Passing Layers?



What is happening in the Message Passing Layers?

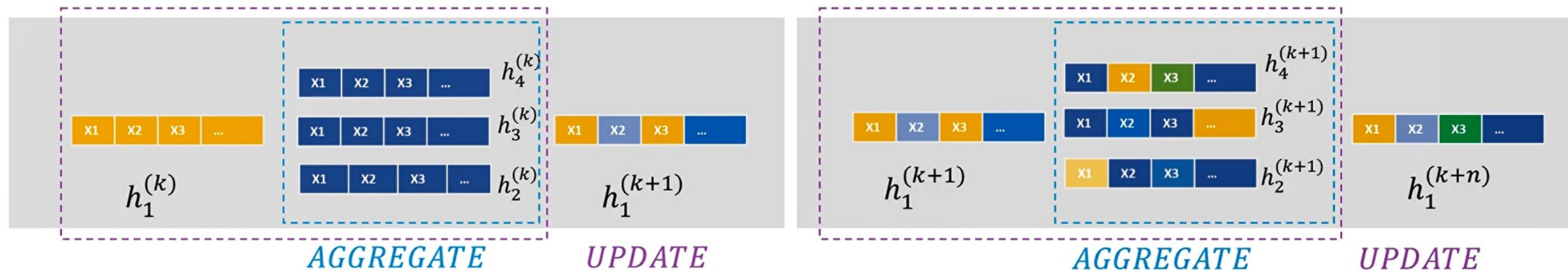
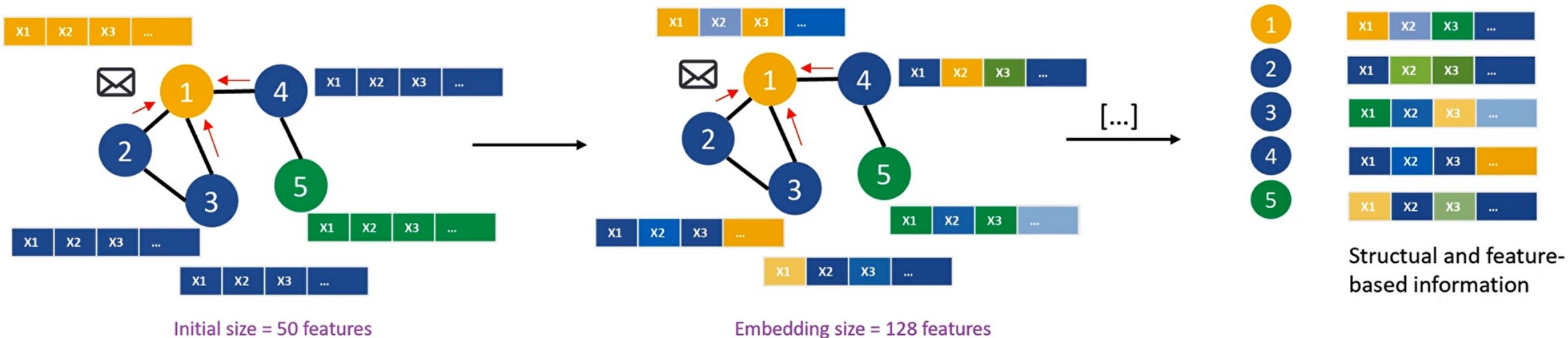


What is happening in the Message Passing Layers?

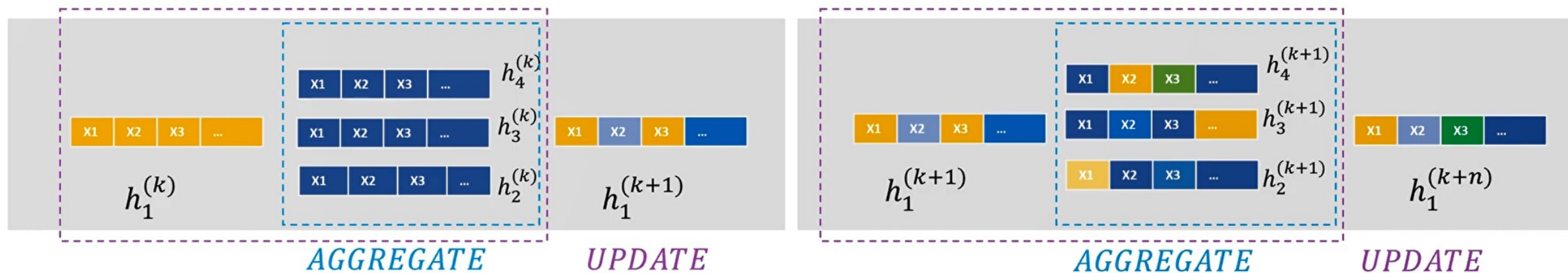
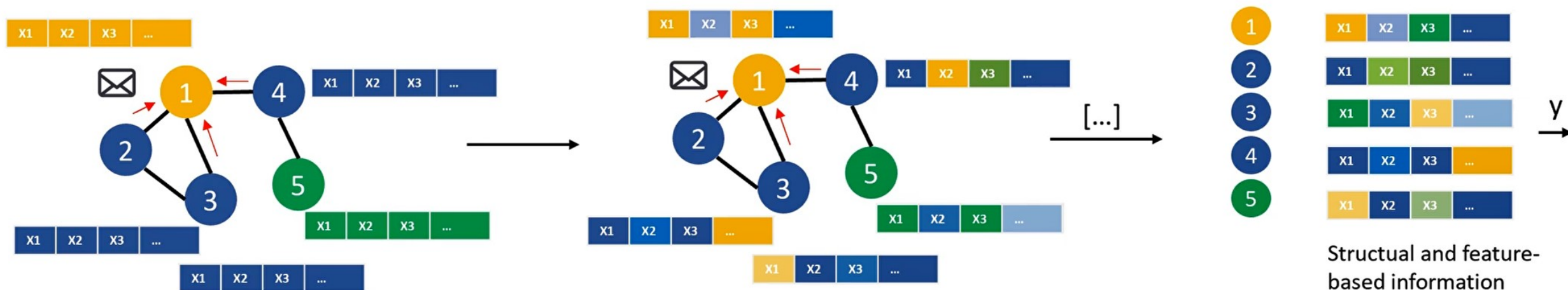




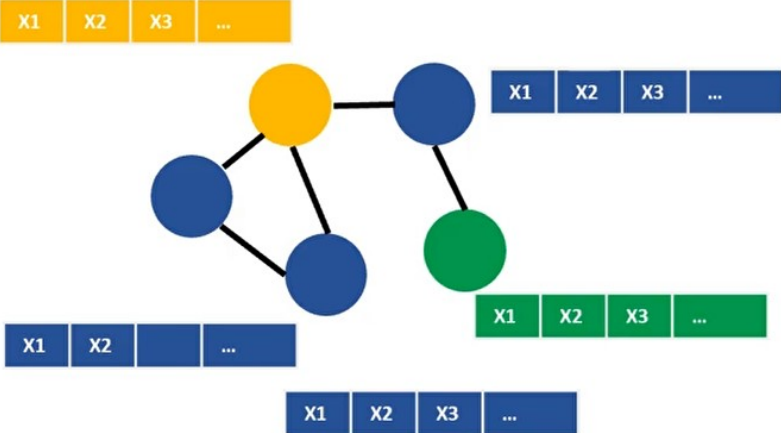
What is happening in the Message Passing Layers?



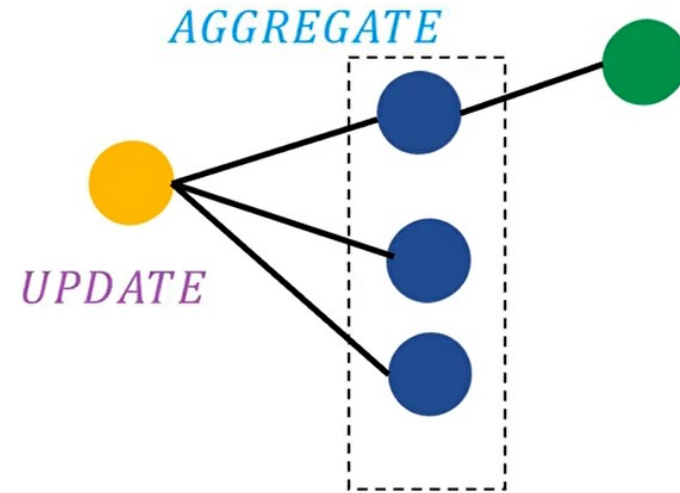
What is happening in the Message Passing Layers?



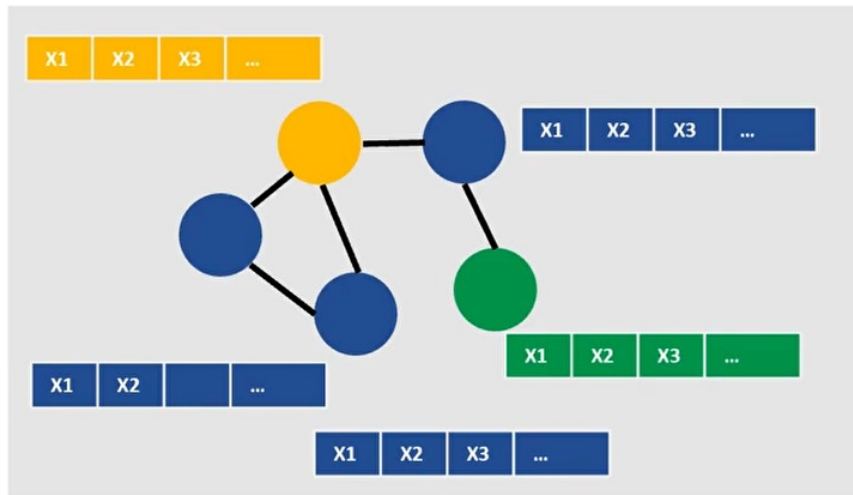
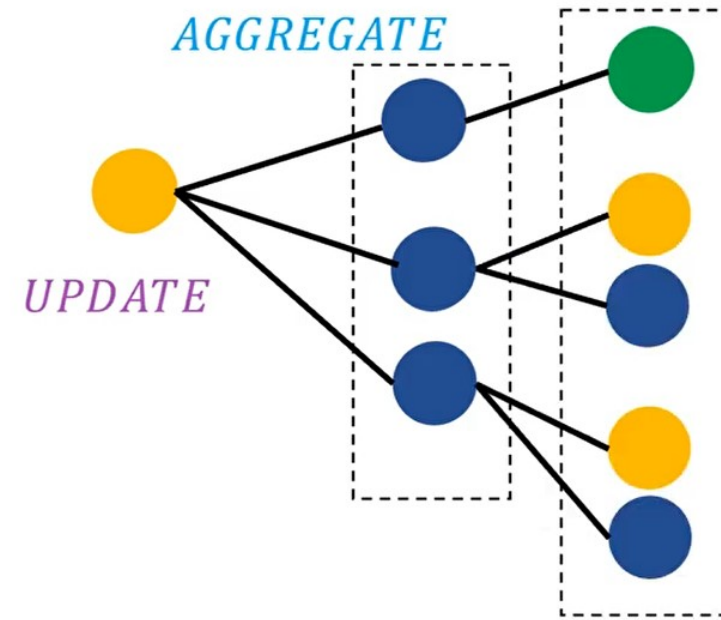
Computation Graph Representation



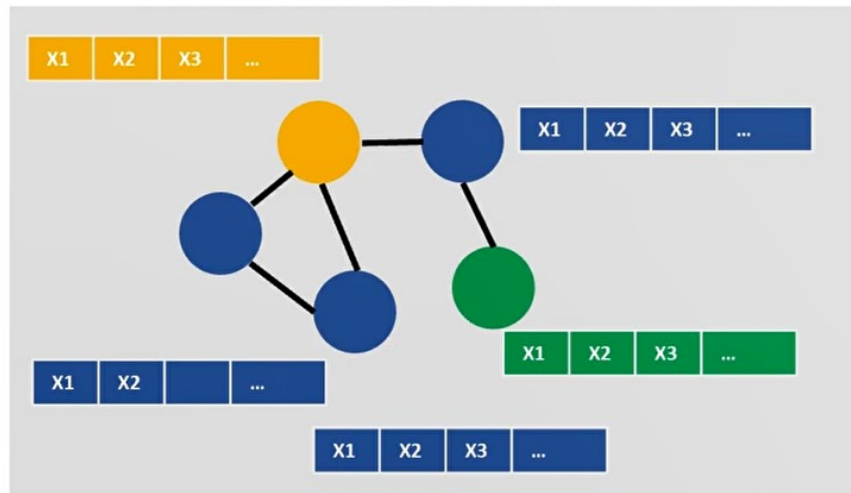
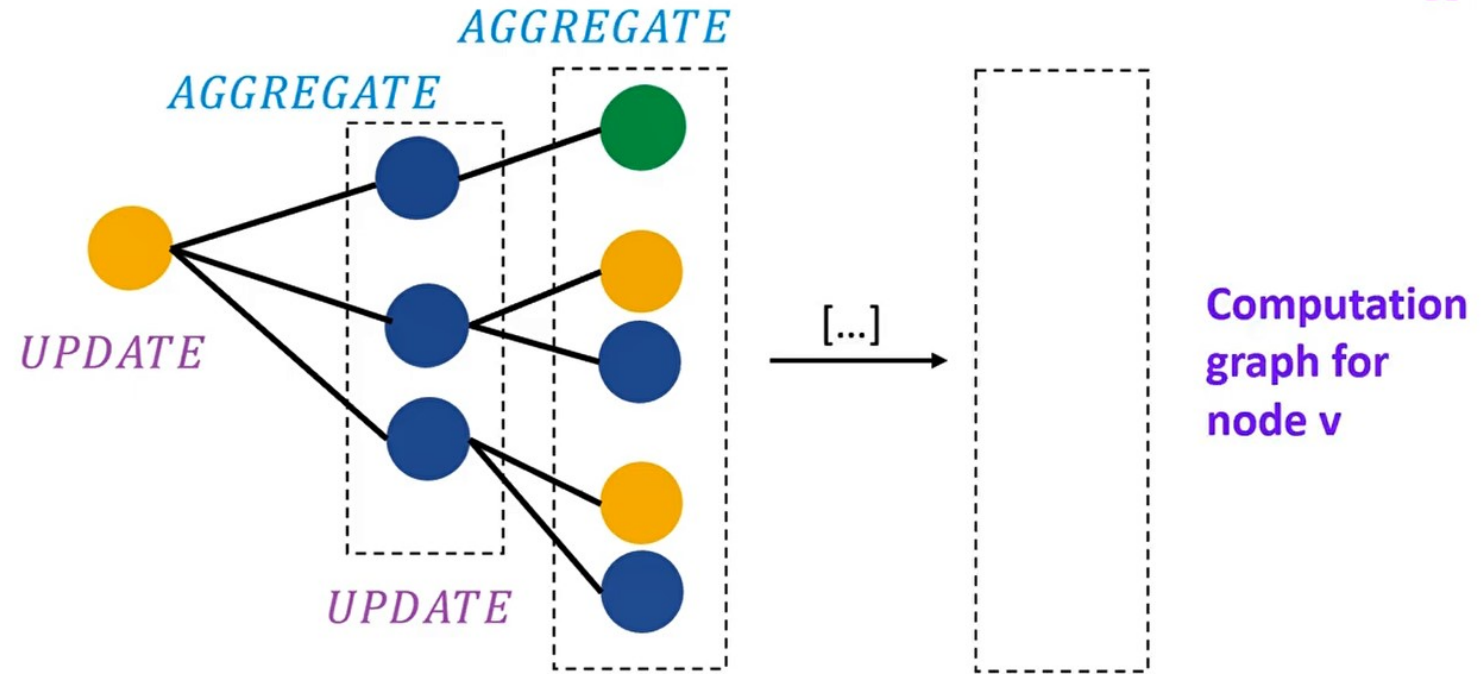
Computation Graph Representation



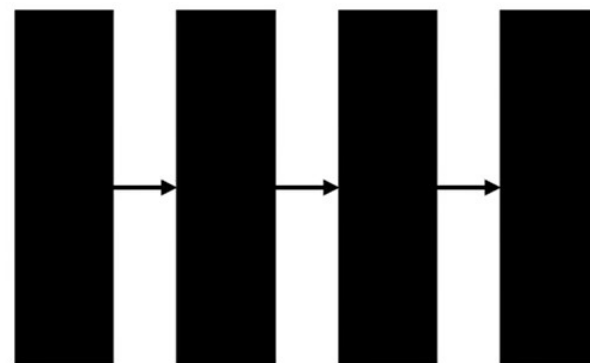
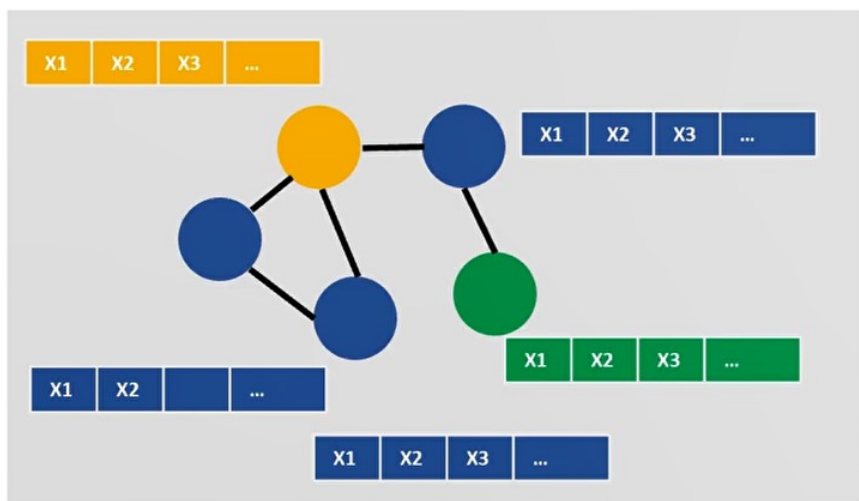
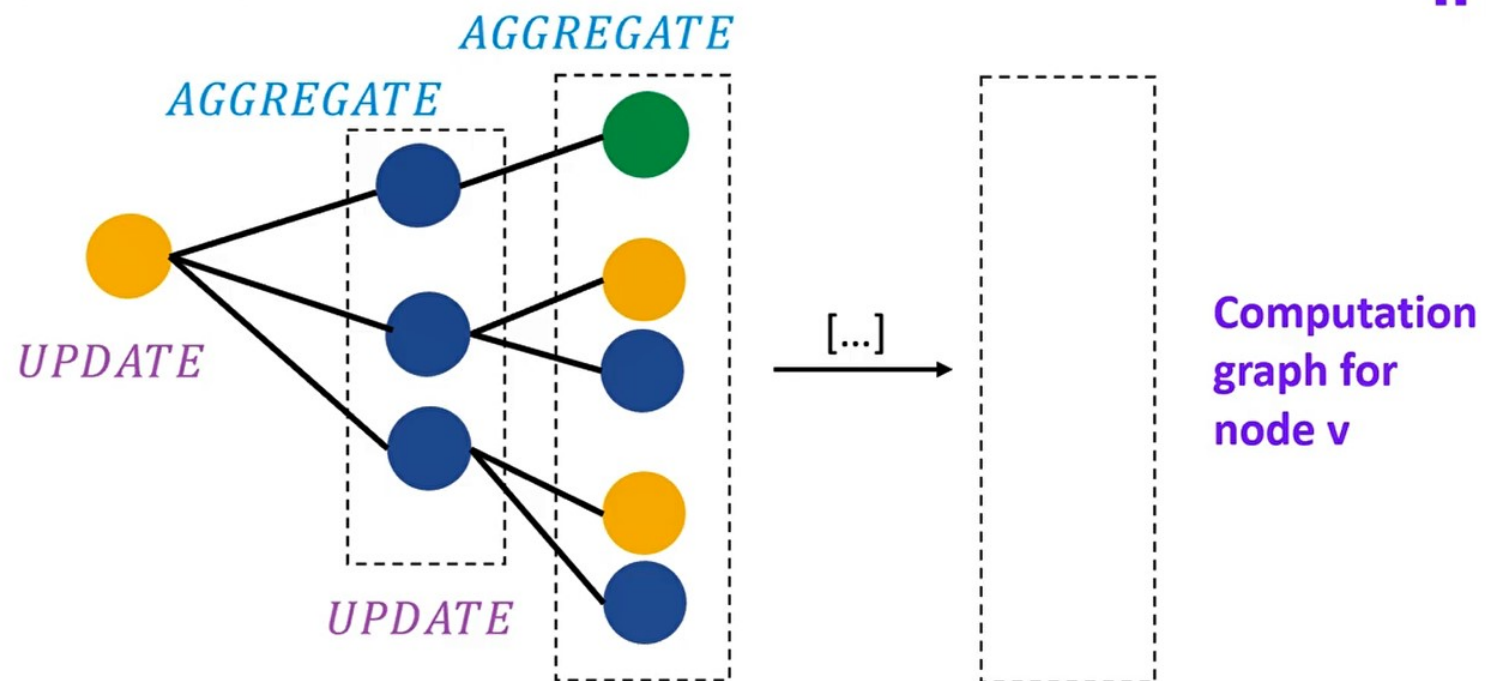
Computation Graph Representation



Computation Graph Representation

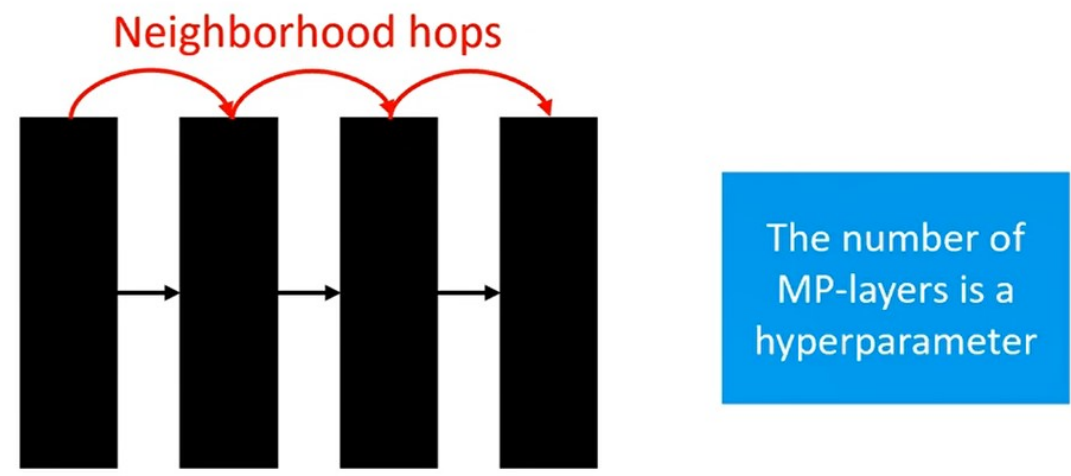
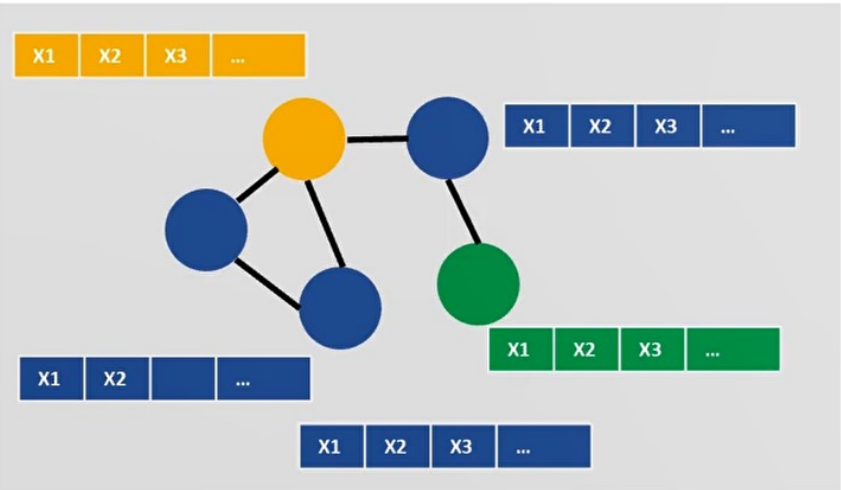
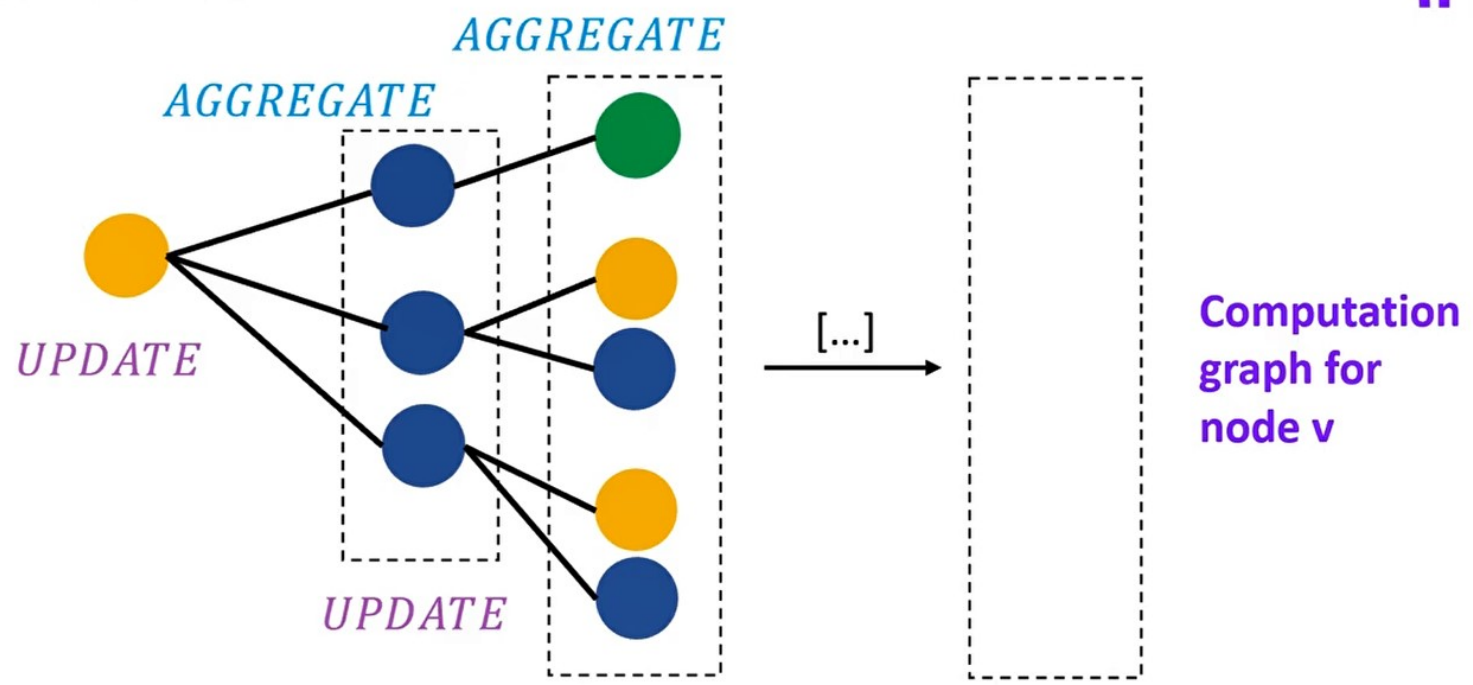


Computation Graph Representation

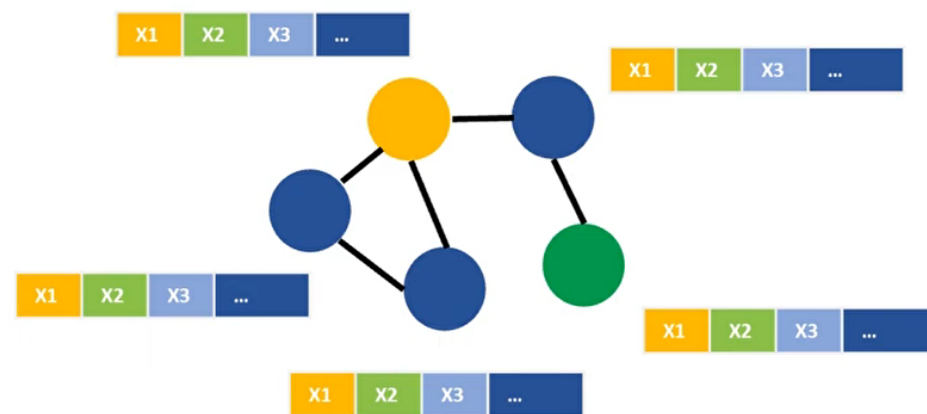
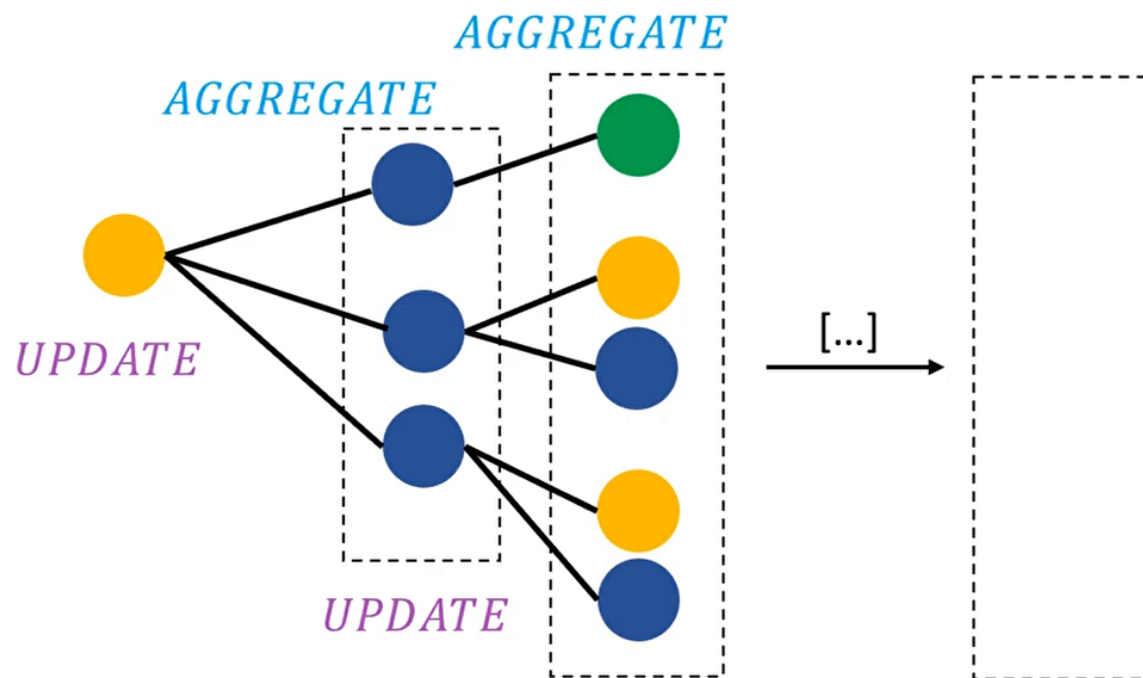


The number of MP-layers is a hyperparameter

Computation Graph Representation



Over-smoothing in GNNs



Message Passing Update and Aggregation Functions

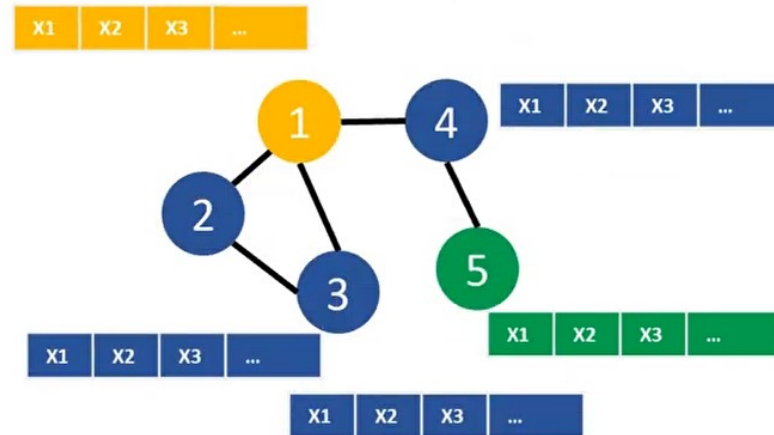


$$h_u^{(k+1)} = \textit{UPDATE}^{(k)} \left(h_u^{(k)}, \textit{AGGREGATE}^{(k)}(\{h_v^{(k)}, \forall v \in \mathcal{N}(u)\}) \right)$$

Message Passing Update and Aggregation Functions



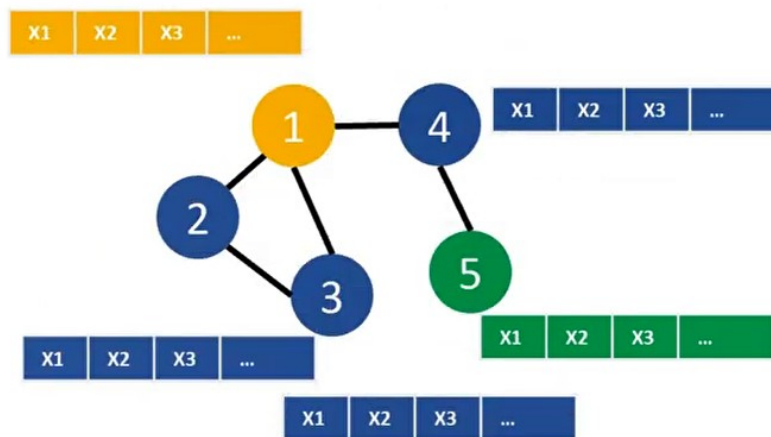
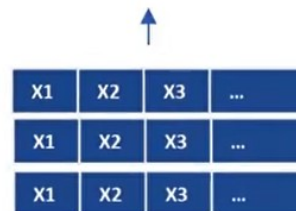
$$h_u^{(k+1)} = \text{UPDATE}^{(k)} \left(h_u^{(k)}, \text{AGGREGATE}^{(k)}(\{h_v^{(k)}, \forall v \in \mathcal{N}(u)\}) \right)$$



Message Passing Update and Aggregation Functions



$$h_u^{(k+1)} = \text{UPDATE}^{(k)} \left(h_u^{(k)}, \text{AGGREGATE}^{(k)}(\{h_v^{(k)}, \forall v \in \mathcal{N}(u)\}) \right)$$



Message Passing Update and Aggregation Functions

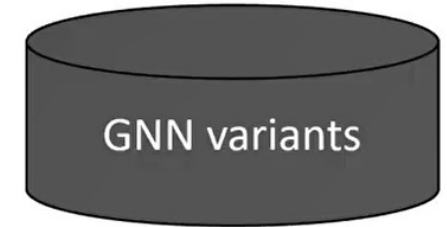


Differentiable functions

$$h_u^{(k+1)} = \text{UPDATE}^{(k)} \left(h_u^{(k)}, \text{AGGREGATE}^{(k)}(\{h_v^{(k)}, \forall v \in \mathcal{N}(u)\}) \right)$$

- Mean
- Max
- Neural Network
- Recurrent NN

- Mean
- Max
- Normalized Sum
- Neural Network



GNN variants



AGGREGATE
(permutation invariant)



UPDATE



Graph Convolutional Networks,
Kipf and Welling [2016]

$$\mathbf{h}_v^{(k)} = \sigma \left(\overset{\text{Self-loop}}{\mathbf{W}^{(k)}} \sum_{v \in \mathcal{N}(u) \cup \{u\}} \frac{\mathbf{h}_v}{\sqrt{|\mathcal{N}(u)| |\mathcal{N}(v)|}} \right) \quad \text{Sum of normalized neighbor embeddings}$$

Multi-Layer-Perceptron as
Aggregator, Zaheer et al. [2017]

Aggregated message

$$\mathbf{m}_{\mathcal{N}(u)} = \underset{\text{trainable!}}{\text{MLP}_{\theta}} \left(\sum_{v \in \mathcal{N}(u)} \text{MLP}_{\phi}(\mathbf{h}_v) \right) \quad \text{Send states through a MLP}$$

Graph Attention Networks,
Veličković et al. [2017]

$$\mathbf{m}_{\mathcal{N}(u)} = \sum_{v \in \mathcal{N}(u)} \alpha_{u,v} \mathbf{h}_v \quad \alpha_{u,v} = \frac{\exp(\mathbf{a}^{\top} [\mathbf{W}\mathbf{h}_u \oplus \mathbf{W}\mathbf{h}_v])}{\sum_{v' \in \mathcal{N}(u)} \exp(\mathbf{a}^{\top} [\mathbf{W}\mathbf{h}_u \oplus \mathbf{W}\mathbf{h}_{v'}])}$$

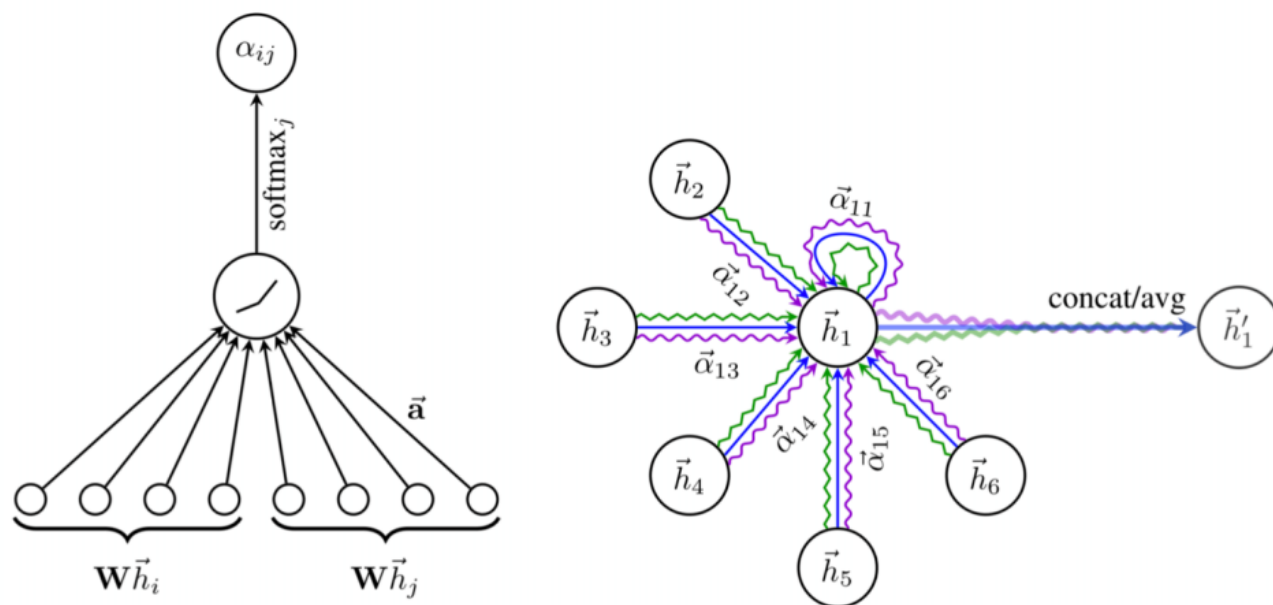
Attention weights

Gated Graph Neural Networks,
Li et al. [2015]

$$\mathbf{h}_u^{(k)} = \text{GRU}(\mathbf{h}_u^{(k-1)}, \mathbf{m}_{\mathcal{N}(u)}^{(k)}) \quad \text{Recurrent update of the state}$$

Graph Neural Networks (GNNs) with **Attention**

Monti et al. (CVPR 2017), Hoshen (NIPS 2017), Veličković et al. (ICLR 2018)



[Figure from Veličković et al. (ICLR 2018)]

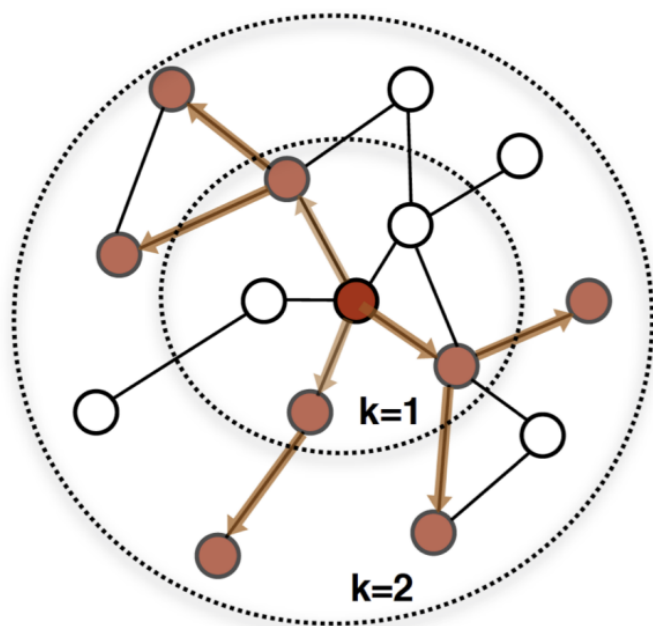
Pros:

- No need to store intermediate edge-based activation vectors (when using dot-product attn.)
- Slower than GCNs but faster than GNNs with edge embeddings

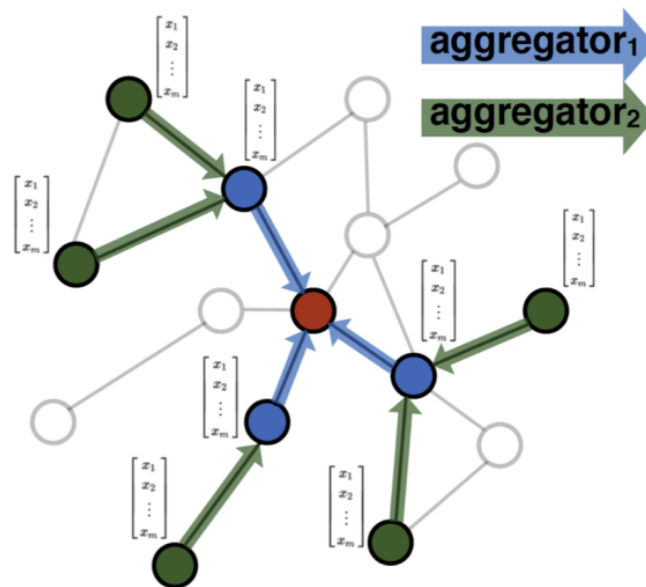
Cons:

- (Most likely) less expressive than GNNs with edge embeddings
- Can be more difficult to optimize

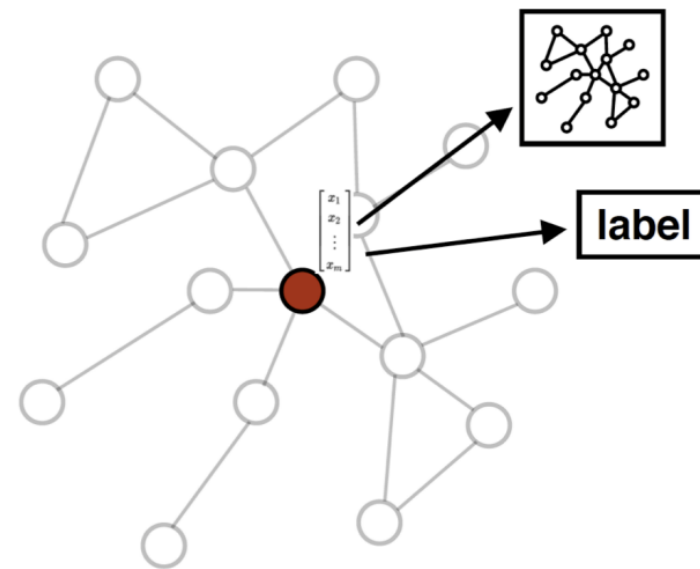
$$\vec{h}'_i = \sigma \left(\frac{1}{K} \sum_{k=1}^K \sum_{j \in \mathcal{N}_i} \alpha_{ij}^k \mathbf{W}^k \vec{h}_j \right) \quad \alpha_{ij} = \frac{\exp \left(\text{LeakyReLU} \left(\vec{a}^T [\mathbf{W} \vec{h}_i \| \mathbf{W} \vec{h}_j] \right) \right)}{\sum_{k \in \mathcal{N}_i} \exp \left(\text{LeakyReLU} \left(\vec{a}^T [\mathbf{W} \vec{h}_i \| \mathbf{W} \vec{h}_k] \right) \right)}$$



1. Sample neighborhood



2. Aggregate feature information from neighbors



3. Predict graph context and label using aggregated information

Figure 1: Visual illustration of the GraphSAGE sample and aggregate approach.

GNN variants

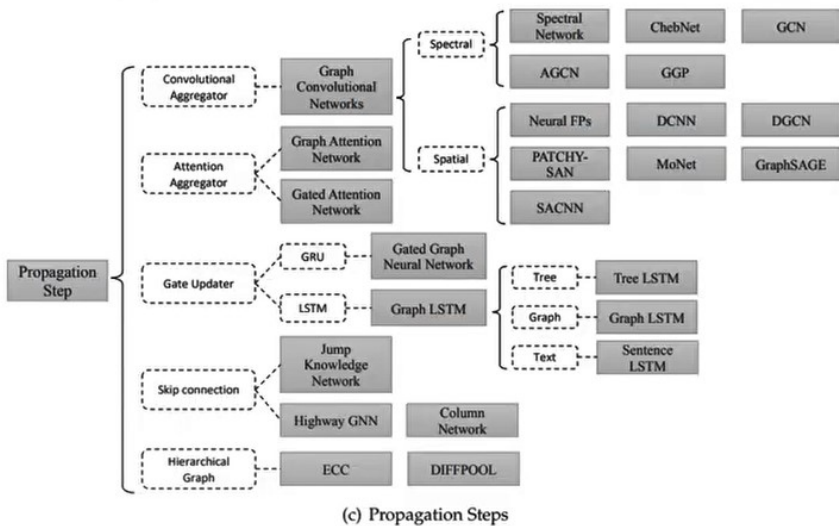
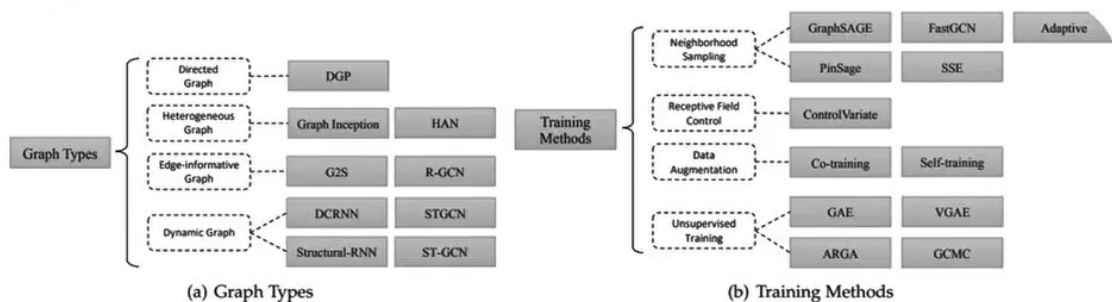


TABLE 2
Different variants of graph neural networks.

Name	Variant	Aggregator	Updater
Spectral Methods	ChebNet	$N_k = T_k(\tilde{L})X$	$H = \sum_{k=0}^K N_k \Theta_k$
	1 st -order model	$N_0 = X$ $N_1 = D^{-\frac{1}{2}} A D^{-\frac{1}{2}} X$	$H = N_0 \Theta_0 + N_1 \Theta_1$
	Single parameter	$N = (I_N + D^{-\frac{1}{2}} A D^{-\frac{1}{2}}) X$	$H = N \Theta$
	GCN	$N = \tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}} X$	$H = N \Theta$
Non-spectral Methods	Neural FPs	$h_{N_v}^t = h_v^{t-1} + \sum_{k=1}^{N_v} h_k^{t-1}$	$h_v^t = \sigma(h_{N_v}^t W_L^{N_v})$
	DCNN	Node classification: $N = P^* X$ Graph classification: $N = 1_N^T P^* X / N$	$H = f(W^c \odot N)$
	GraphSAGE	$h_{N_v}^t = \text{AGGREGATE}_t(\{h_u^{t-1}, \forall u \in N_v\})$	$h_v^t = \sigma(W^t \cdot [h_v^{t-1} \ h_{N_v}^t])$
Graph Attention Networks	GAT	$\alpha_{vk} = \frac{\exp(\text{LeakyReLU}(a^T [W_h v \ W_h k]))}{\sum_{j \in N_v} \exp(\text{LeakyReLU}(a^T [W_h v \ W_h j]))}$ $h_{N_v}^t = \sigma(\sum_{k \in N_v} \alpha_{vk} W h_k)$ Multi-head concatenation: $h_{N_v}^t = \parallel_{m=1}^M \sigma(\sum_{k \in N_v} \alpha_{vk}^m W^m h_k)$ Multi-head average: $h_{N_v}^t = \sigma(\frac{1}{M} \sum_{m=1}^M \sum_{k \in N_v} \alpha_{vk}^m W^m h_k)$	$h_v^t = h_{N_v}^t$
Gated Graph Neural Networks	GGNN	$h_{N_v}^t = \sum_{k \in N_v} h_k^{t-1} + b$	$z_v^t = \sigma(W^z h_{N_v}^t + U^z h_v^{t-1})$ $r_v^t = \sigma(W^r h_{N_v}^t + U^r h_v^{t-1})$ $h_v^t = \tanh(W h_{N_v}^t + U(h_v^{t-1} \odot r_v^t))$ $h_v^t = (1 - z_v^t) \odot h_v^{t-1} + z_v^t \odot h_v^t$
			$h_v^t = \sigma(W h_v^{t-1} + U h_v^{t-1})$
			$P_i^k = (1 - \alpha_i^k) \odot P_{i-1}^k + \alpha_i^k \odot P_i^k$ $P_i^k = \text{comp}(M P_i^{k-1} + \Omega(\alpha_i^k \odot P_{i-1}^k))$ $\alpha_i^k = \sigma(M_\alpha P_i^{k-1} + \Omega(\alpha_i^k \odot P_{i-1}^k))$ $\alpha_i^k = \sigma(M_\alpha P_i^{k-1} + \Omega(\alpha_i^k \odot P_{i-1}^k))$
			$P_i^k = \sigma\left(\frac{1}{M} \sum_{m=1}^M \sum_{k \in N_v} \alpha_{vk}^m W^m h_k\right)$

Further literature



William L. Hamilton, **Graph Representation Learning Book** - https://www.cs.mcgill.ca/~wlh/grl_book/files/GRL_Book.pdf

GRAPH REPRESENTATION LEARNING

WILLIAM L. HAMILTON

McGill University

2020

References

- Hamilton, William L. *Graph representation learning*. Morgan & Claypool Publishers, 2020.
- Zhou, Jie, et al. "Graph neural networks: A review of methods and applications." *AI open* 1 (2020): 57-81.
- Jin, Zhihua, et al. "Gnnlens: A visual analytics approach for prediction error diagnosis of graph neural networks." *IEEE Transactions on Visualization and Computer Graphics* (2022).
- Youtub
 - https://www.youtube.com/watch?v=bDqQF3sT1W0&t=6397s&ab_channel=%D9%85%D8%AD%D9%85%D8%AF%D8%B1%D8%B4%D8%A7%D8%AF%7CMohamedRashad
 - <https://www.youtube.com/playlist?list=PLV8yxwGOxvvoNkzPfCx2i8an--Tkt7O8Z>
 - <https://www.youtube.com/playlist?list=PLoROMvodv4rPLKxlpqhjhPg dQy7imNkDn>