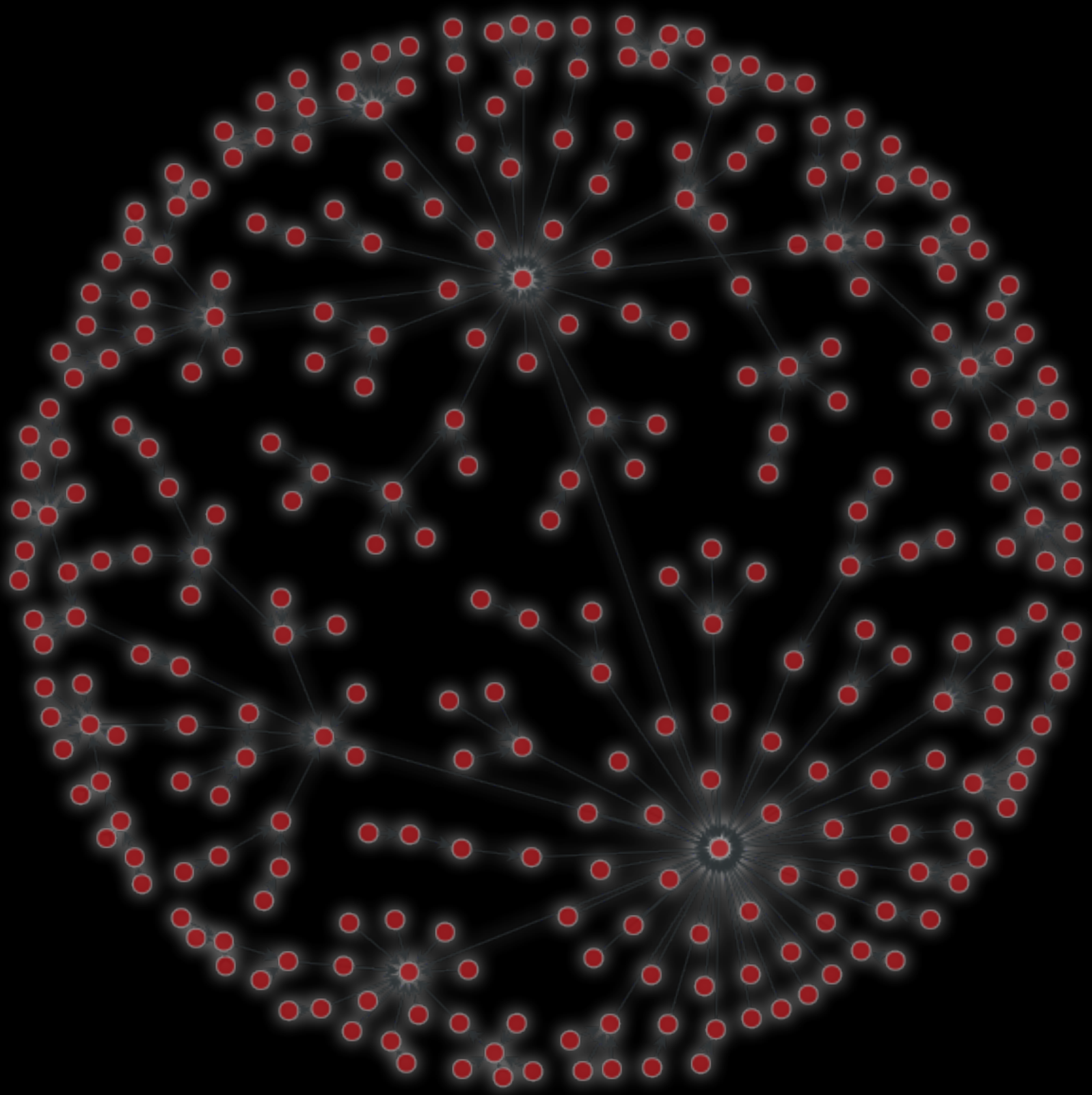


Graph Neural Networks

1. Know what to use to implement a Graph Neural Network

1. Know what to use to implement a Graph Neural Network
2. Intuition for the kinds of problems in which GNNs will provide an advantage

1. Know what to use to implement a Graph Neural Network

2. Intuition for the kinds of problems in which GNNs will provide an advantage

3. Understand why structure is crucial in determining the behavior of interacting systems

1. Know what to use to implement a Graph Neural Network

2. Intuition for the kinds of problems in which GNNs will provide an advantage

3. Understand why structure is crucial in determining the behavior of interacting systems

4. Understand why **relational inductive biases** are critical for learning about interacting systems

This talk

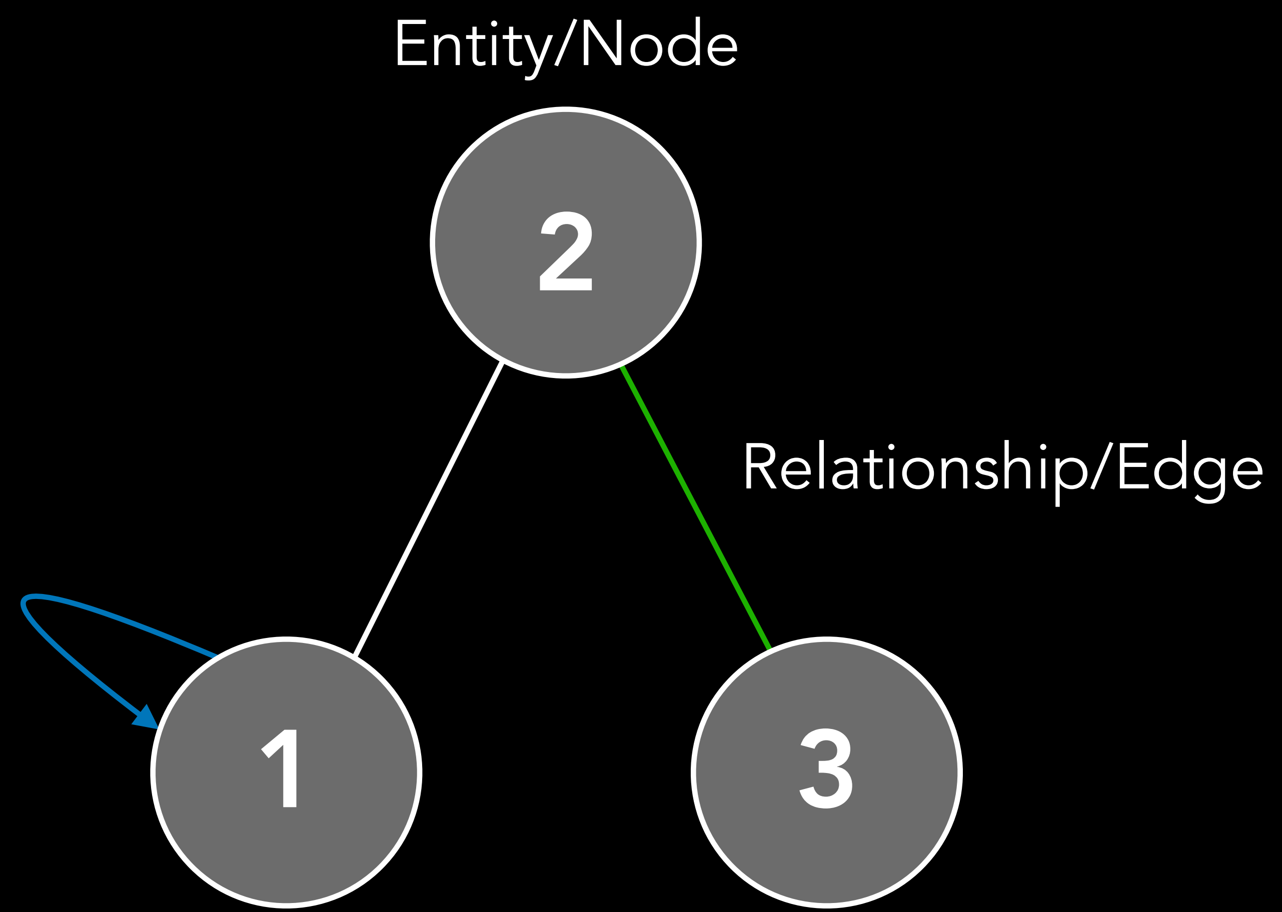
- **Motivation**
- **Mechanisms**
- **Survey**
- **Challenges**

Structure	What do you mean?
GNN	What is this?
Relevance	Why should I care?

Structure

GNN

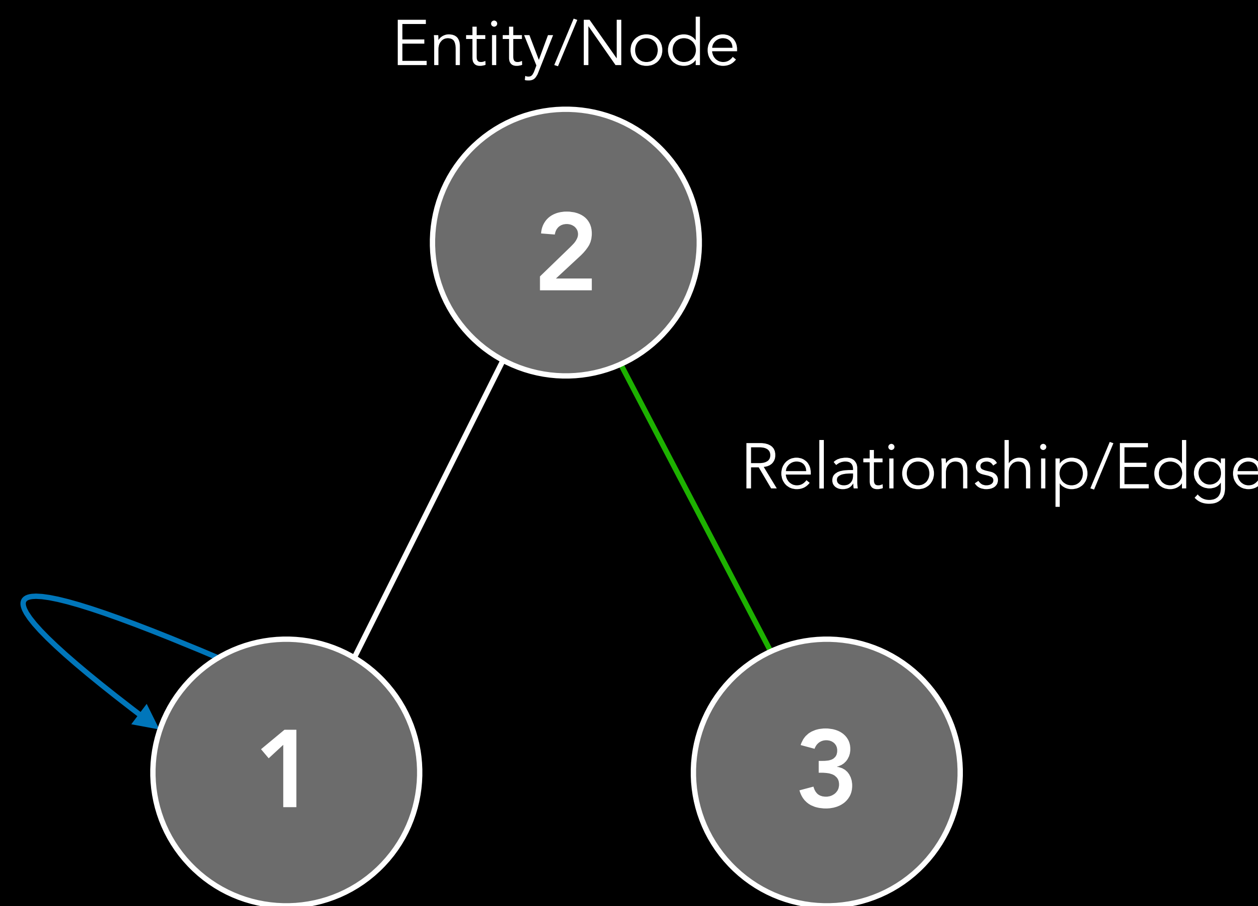
Relevance



Structure

GNN

Relevance



	1	2	3
1	1	1	0
2	1	0	1
3	0	1	0

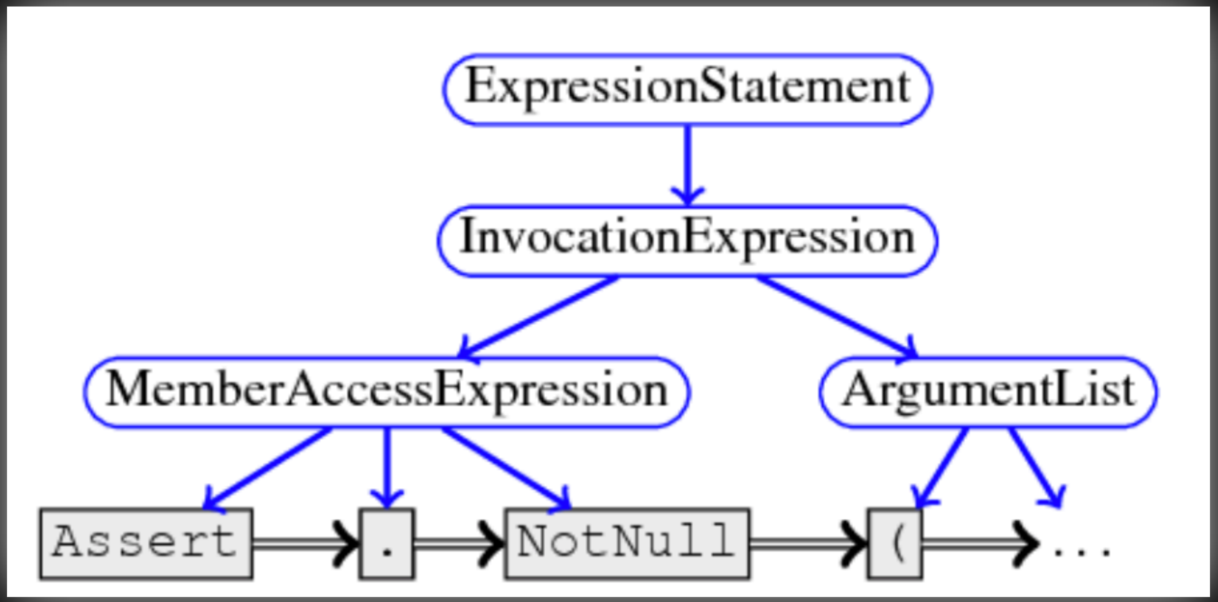
Adjacency Matrix

Structure

GNN

Relevance

Programs



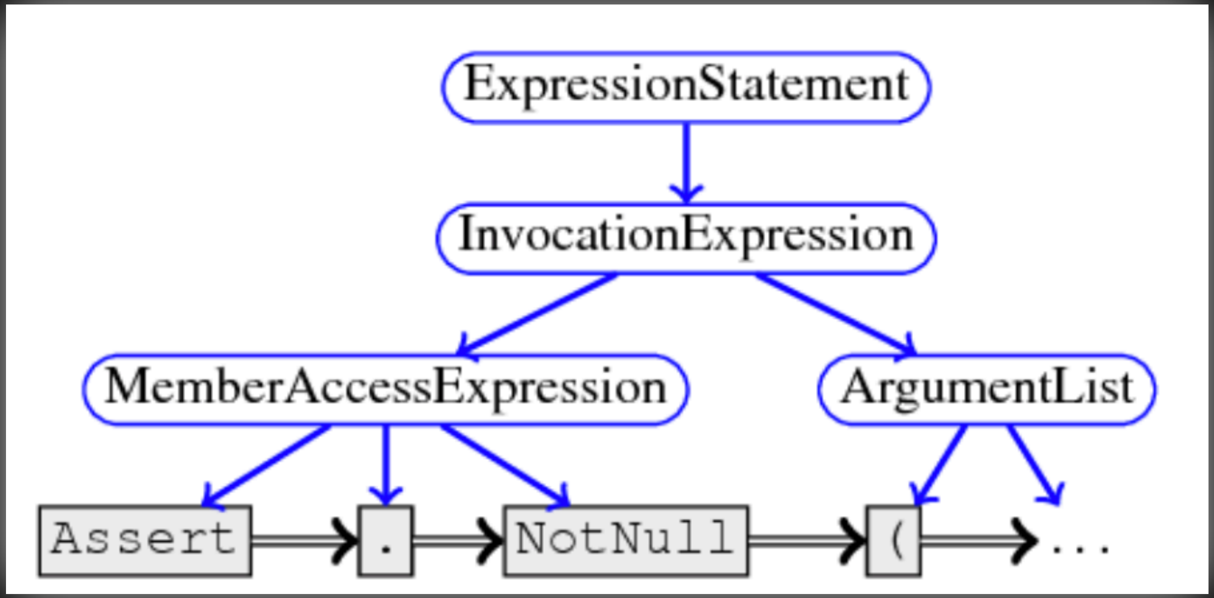
https://www.semanticscholar.org/paper/Learning-to-Represent-Programs

Structure

GNN

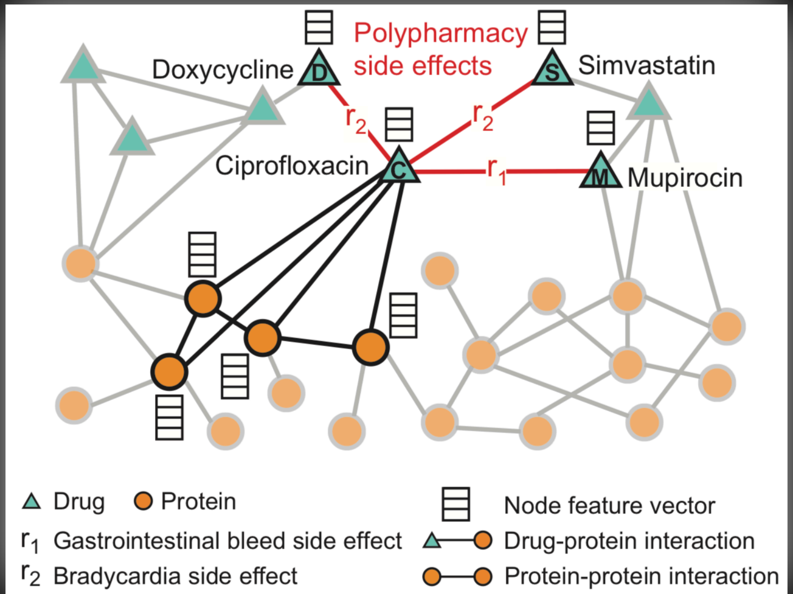
Relevance

Programs



https://www.semanticscholar.org/paper/Learning-to-Represent-Programs

Drug Interactions



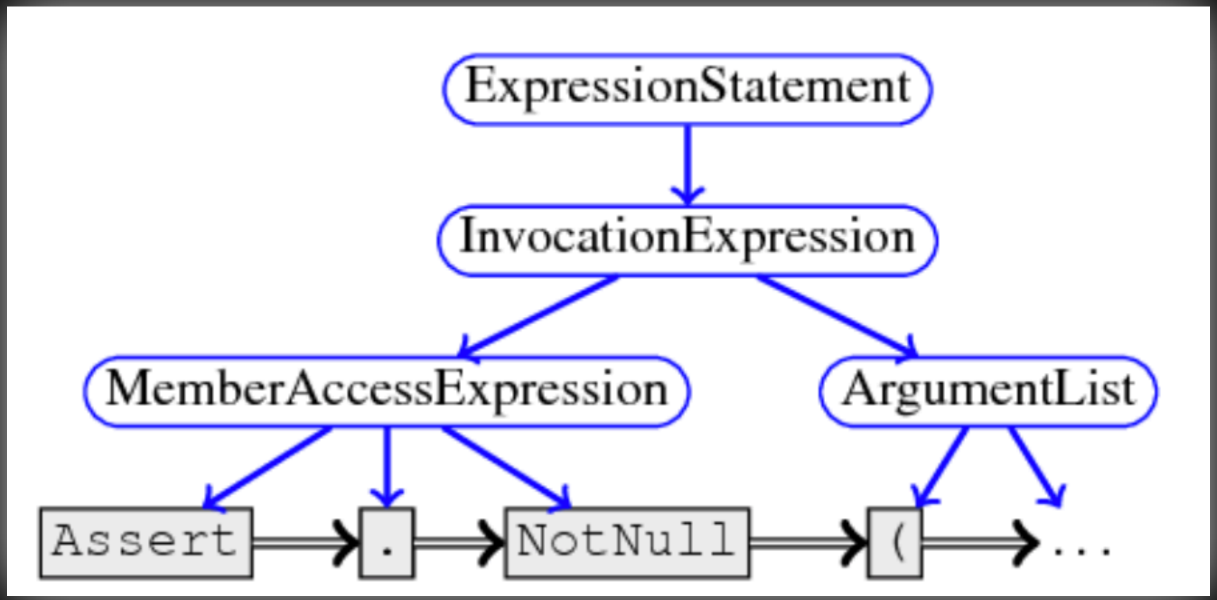
https://arxiv.org/pdf/1802.00543.pdf

Structure

GNN

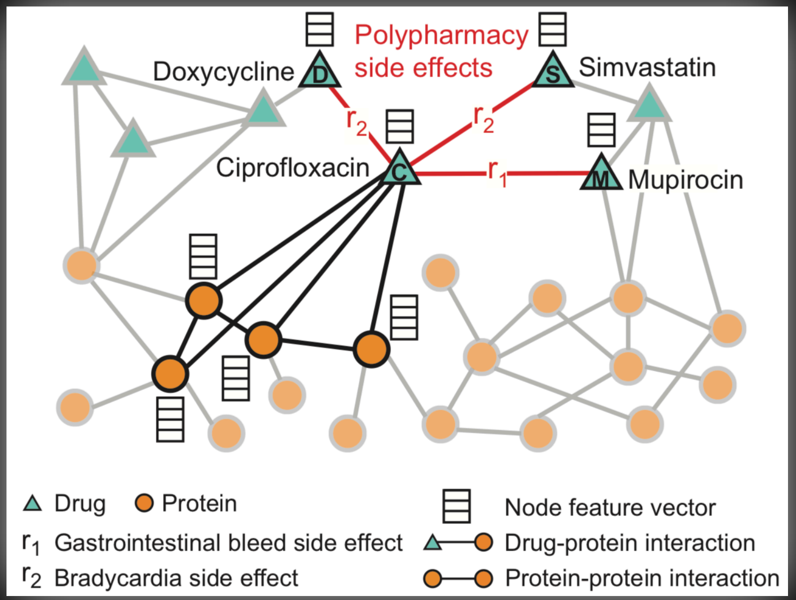
Relevance

Programs



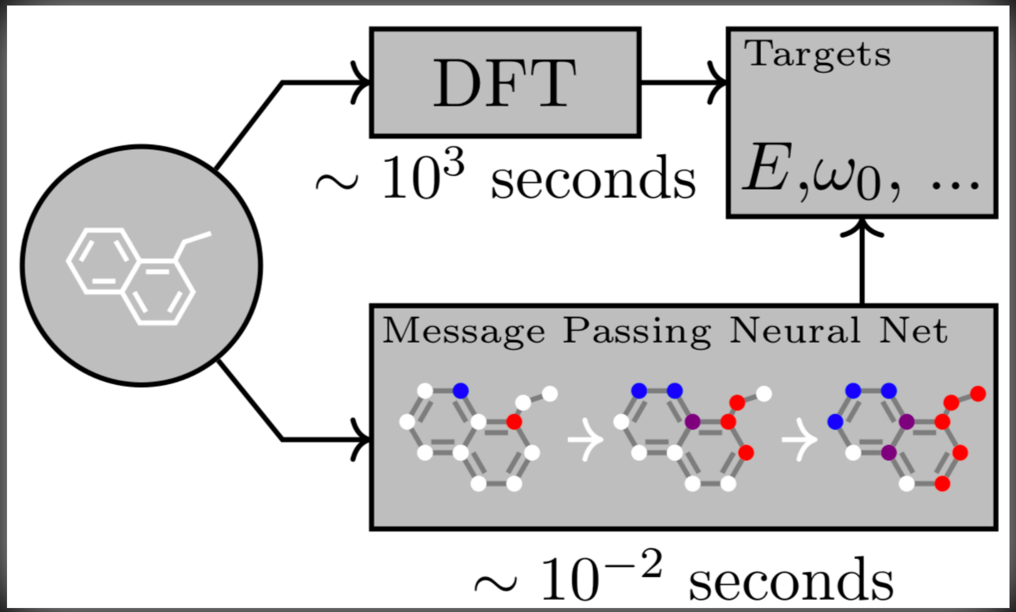
https://www.semanticscholar.org/paper/Learning-to-Represent-Programs

Drug Interactions



https://arxiv.org/pdf/1802.00543.pdf

Physical Systems



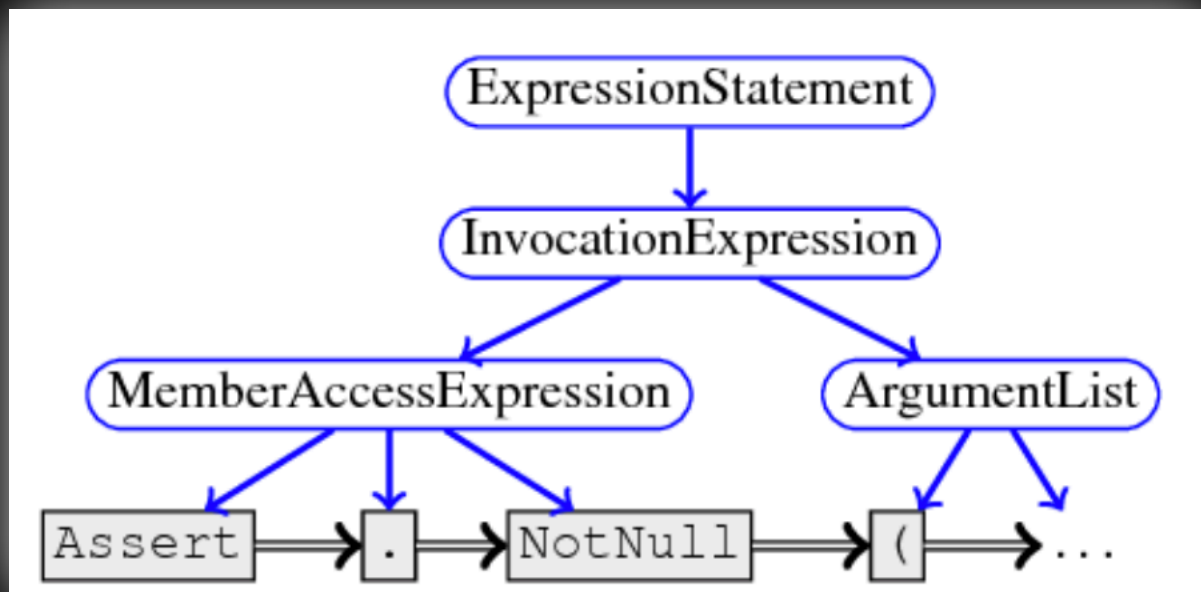
https://arxiv.org/pdf/1704.01212.pdf

Structure

GNN

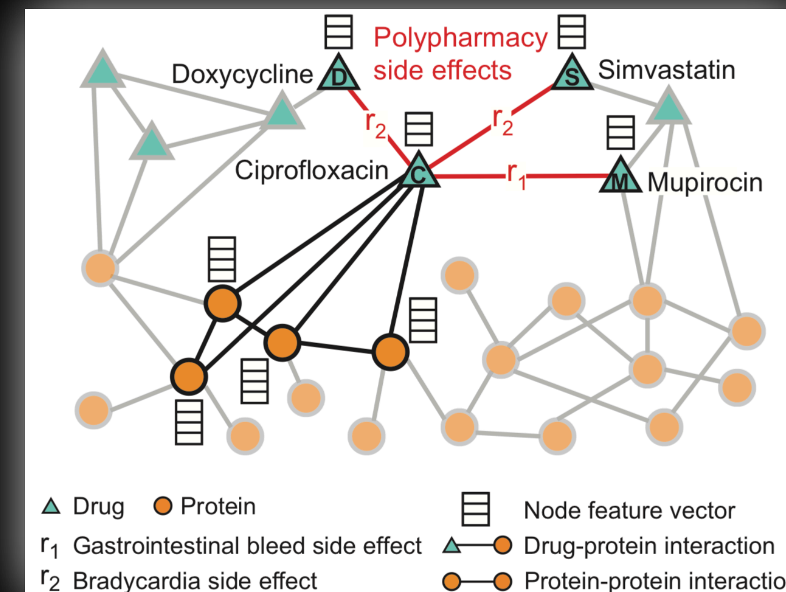
Relevance

Programs



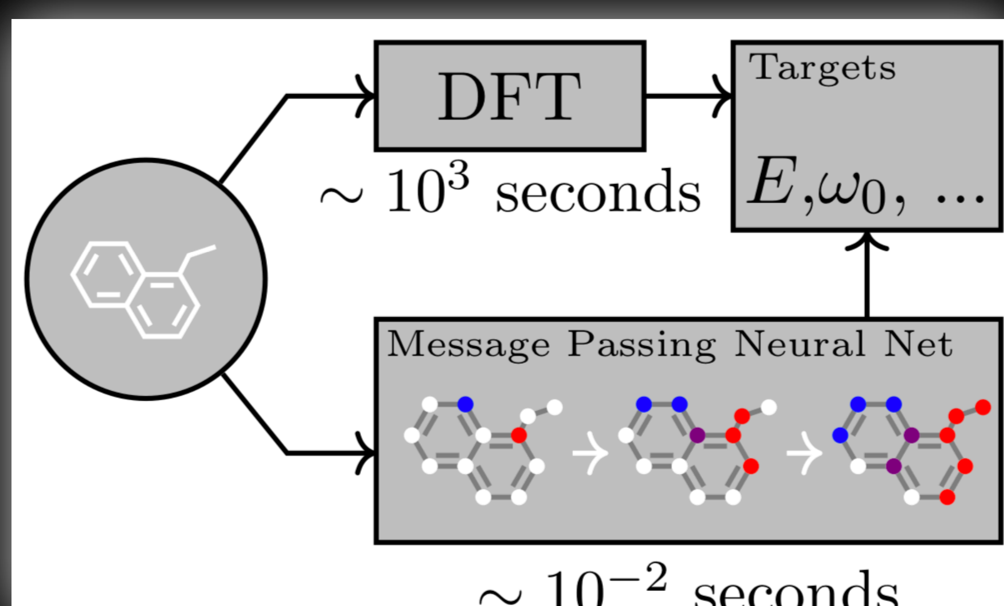
https://www.semanticscholar.org/paper/Learning-to-Represent-Programs

Drug Interactions



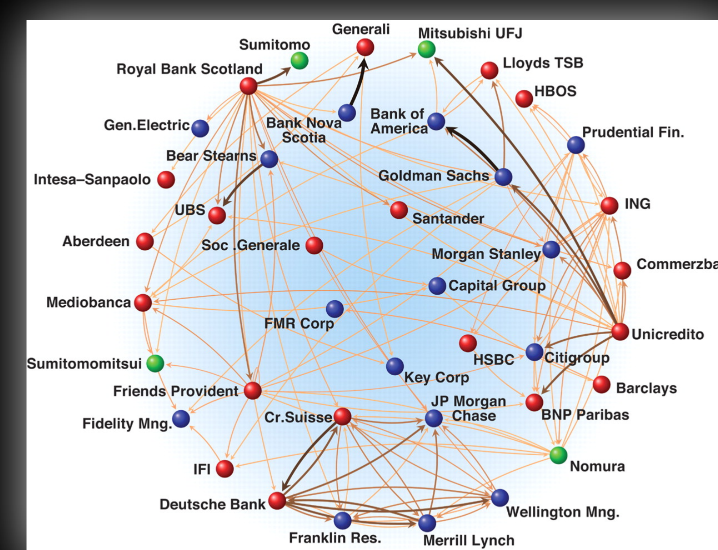
https://arxiv.org/pdf/1802.00543.pdf

Physical Systems



https://arxiv.org/pdf/1704.01212.pdf

Economic Networks

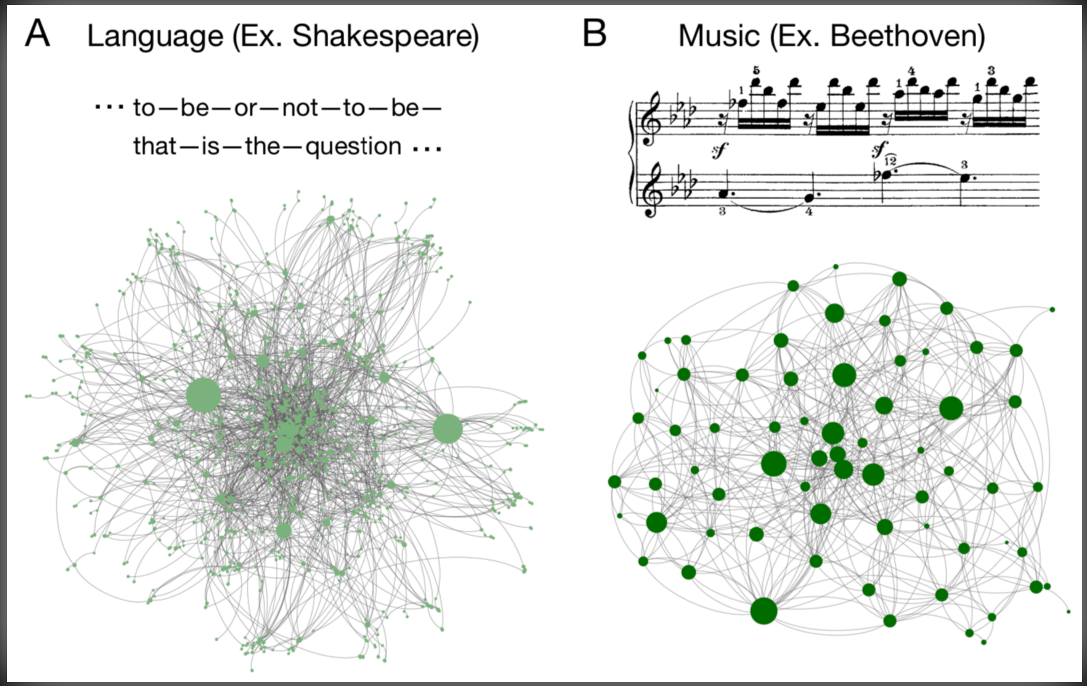


https://science.sciencemag.org/content/325/5939/422

Brief foray into Cognitive Science…

Brief foray into Cognitive Science…

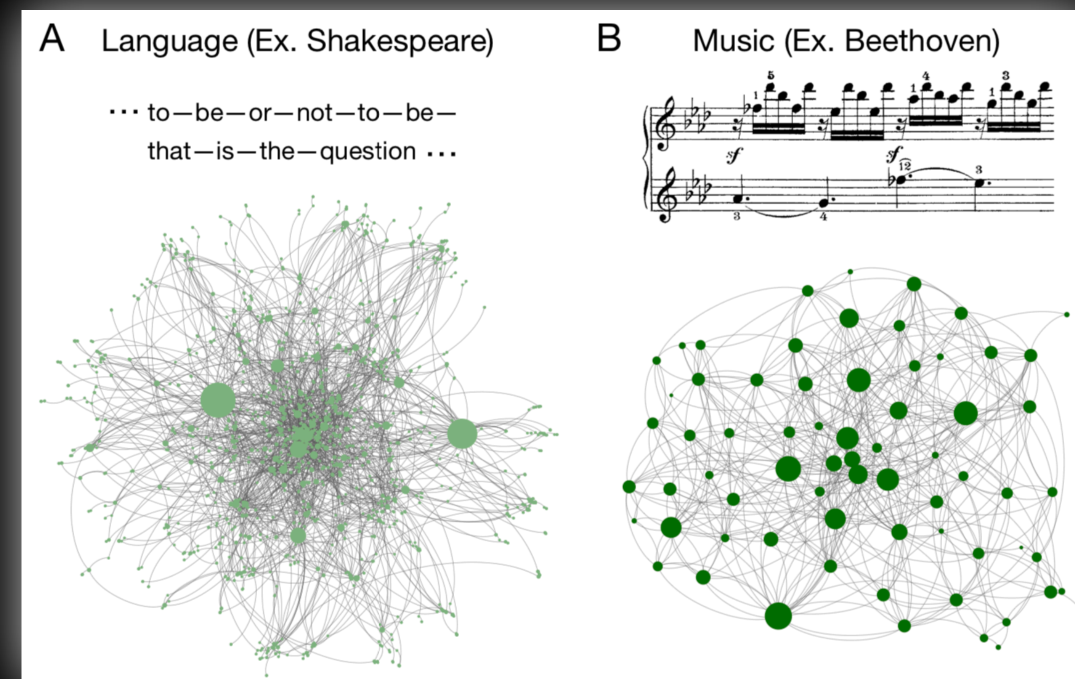
Cognitive Representation



https://arxiv.org/pdf/1909.07186.pdf

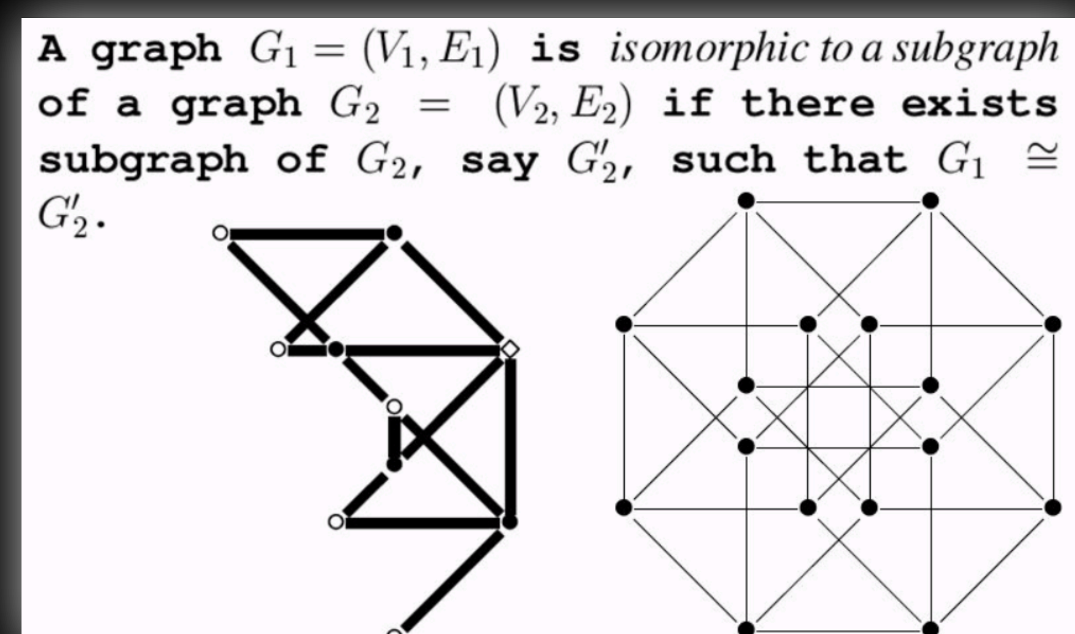
Brief foray into Cognitive Science…

Cognitive Representation



https://arxiv.org/pdf/1909.07186.pdf

Analogy



https://link.springer.com/article/10.1007/s10618-009-0132-7

Structure

GNN

Relevance

GNN as meta-architecture for imparting **relational inductive biases**

Structure

GNN

Relevance

GNN as meta-architecture for imparting **relational inductive biases**

Recurrent units + MLPs + Convolutional units projected onto a graph structure

Structure

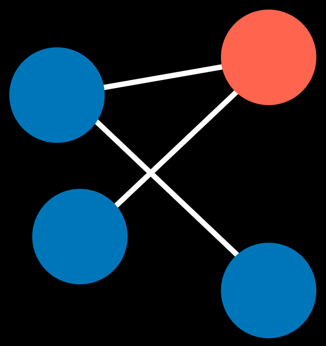
GNN

Relevance

GNN as meta-architecture for imparting **relational inductive biases**

Recurrent units + MLPs + Convolutional units projected onto a graph structure

Node Embedding:



Structure

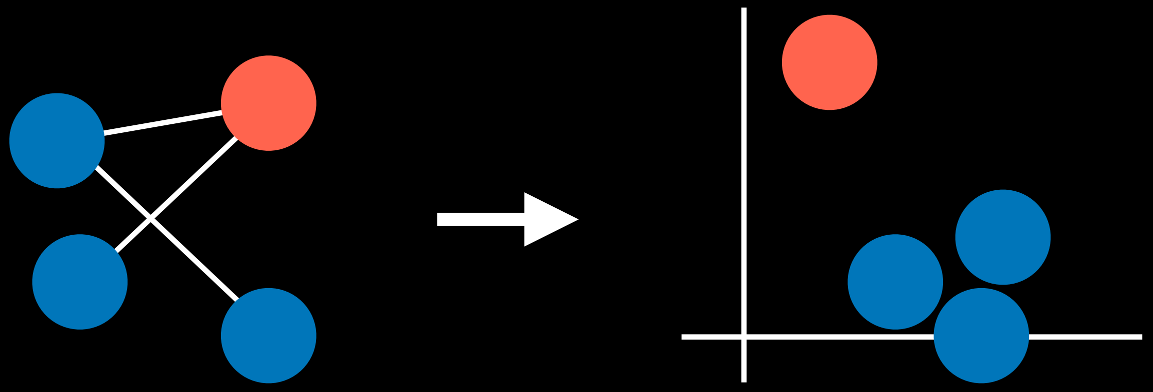
GNN

Relevance

GNN as meta-architecture for imparting **relational inductive biases**

Recurrent units + MLPs + Convolutional units projected onto a graph structure

Node Embedding:



Structure

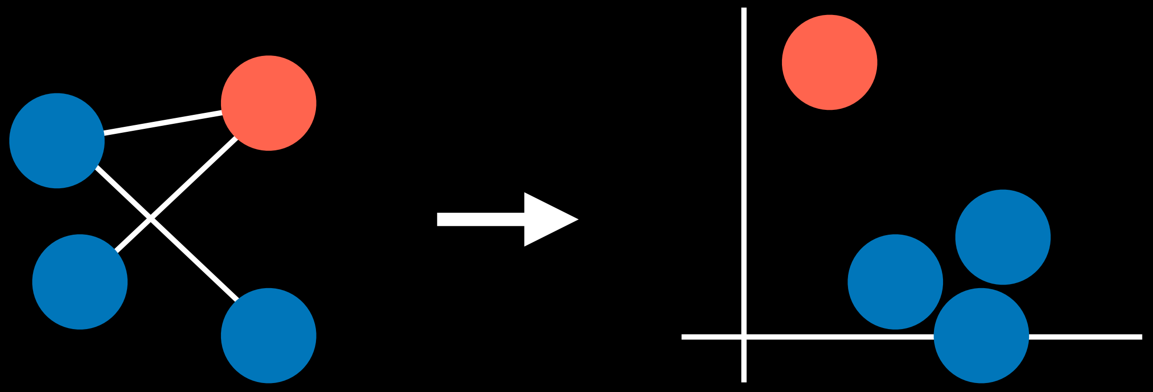
GNN

Relevance

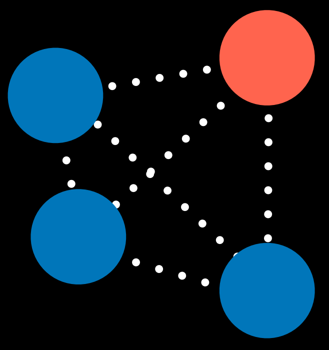
GNN as meta-architecture for imparting **relational inductive biases**

Recurrent units + MLPs + Convolutional units projected onto a graph structure

Node Embedding:



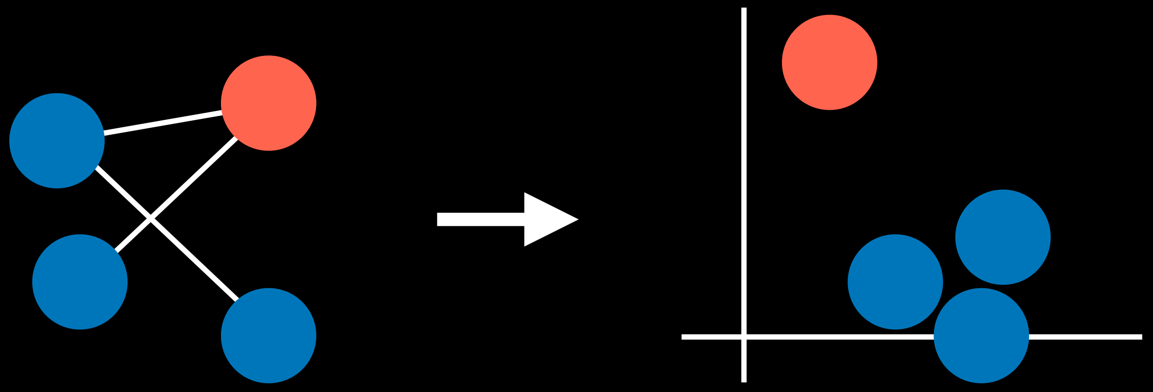
Link Prediction:



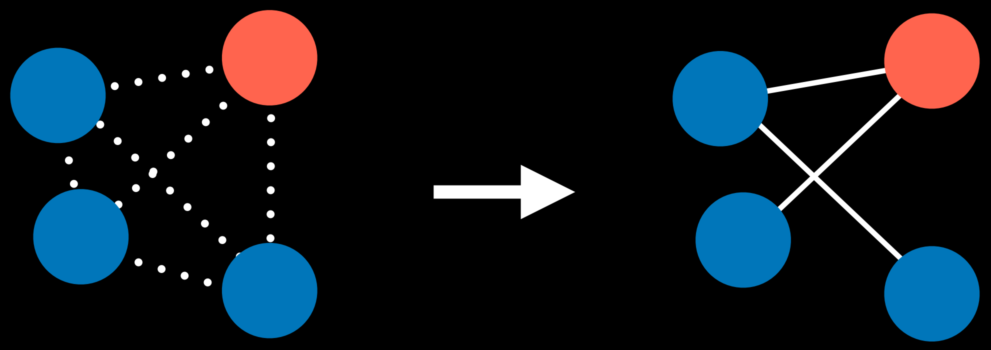
GNN as meta-architecture for imparting **relational inductive biases**

Recurrent units + MLPs + Convolutional units projected onto a graph structure

Node Embedding:



Link Prediction:



Structure

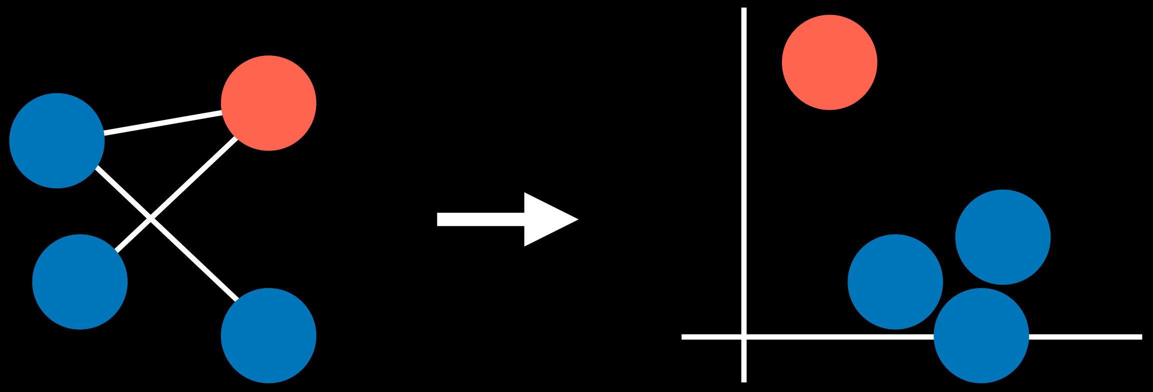
GNN

Relevance

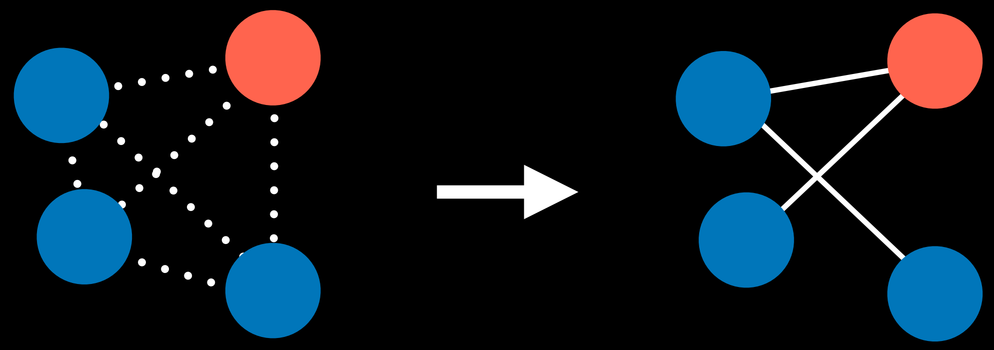
GNN as meta-architecture for imparting **relational inductive biases**

Recurrent units + MLPs + Convolutional units projected onto a graph structure

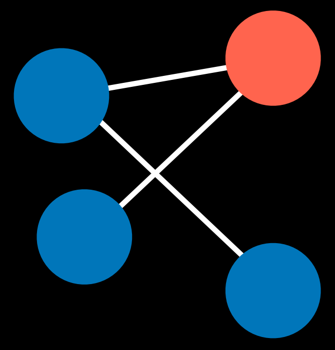
Node Embedding:



Link Prediction:



Graph Embedding:



Structure

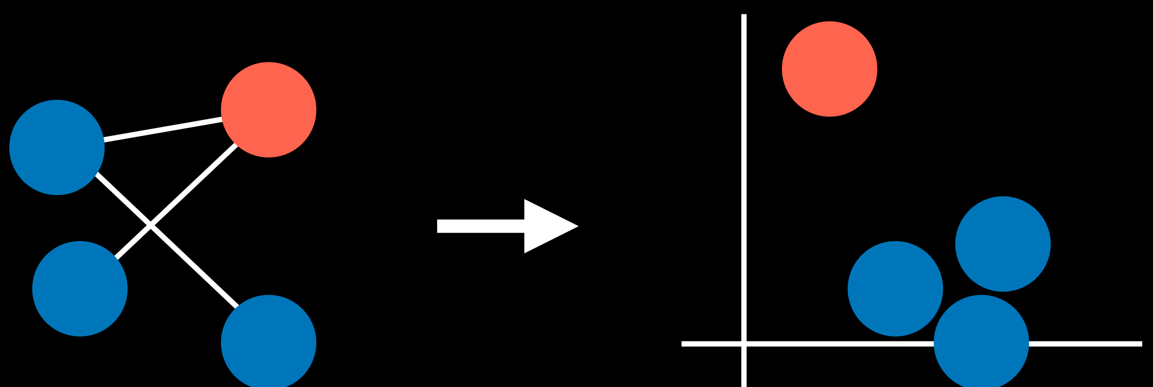
GNN

Relevance

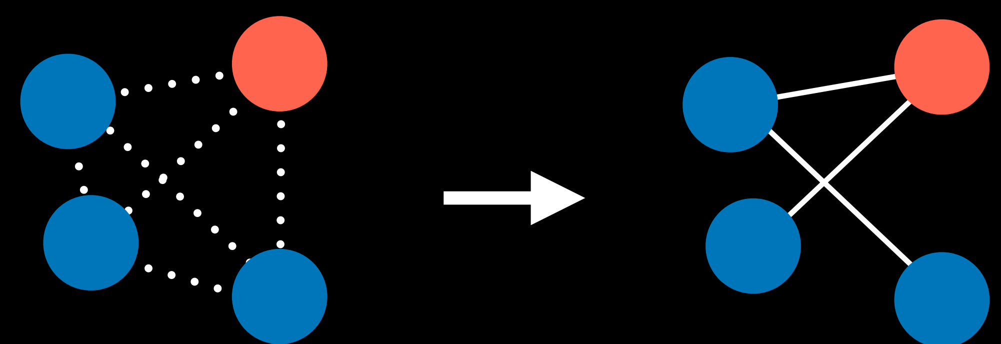
GNN as meta-architecture for imparting **relational inductive biases**

Recurrent units + MLPs + Convolutional units projected onto a graph structure

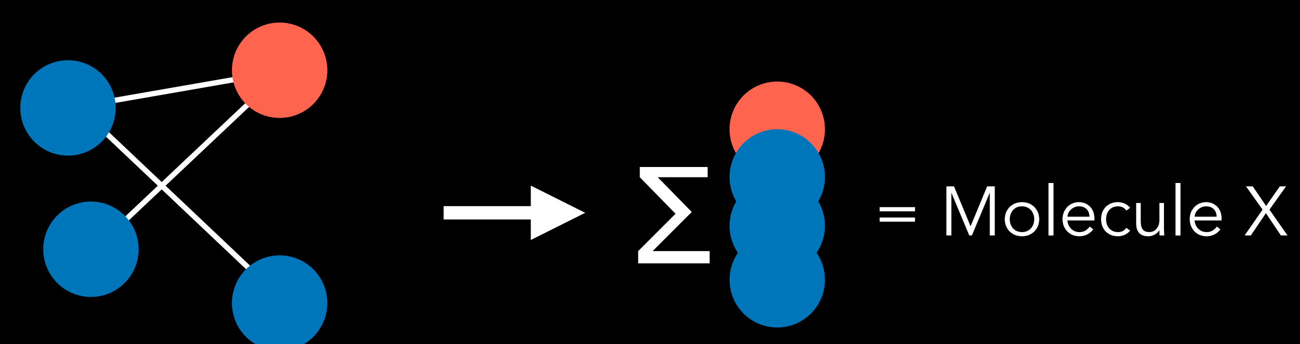
Node Embedding:



Link Prediction:



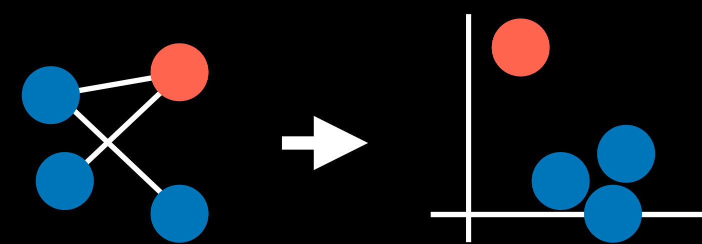
Graph Embedding:



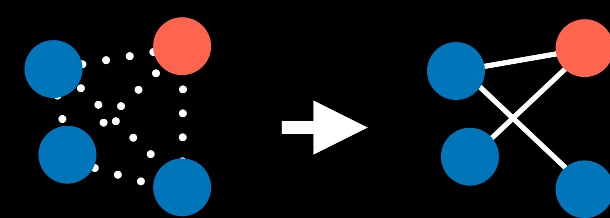
GNN as meta-architecture for imparting **relational inductive biases**

Recurrent units + MLPs + Convolutional units projected onto a graph structure

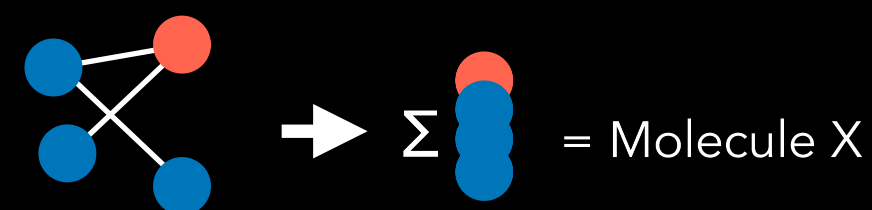
Node Embedding:



Link Prediction:



Graph Embedding:



Component	Entities	Relations	Rel. inductive bias	Invariance
Fully connected	Units	All-to-all	Weak	-
Convolutional	Grid elements	Local	Locality	Spatial translation
Recurrent	Timesteps	Sequential	Sequentiality	Time translation
Graph network	Nodes	Edges	Arbitrary	Node, edge permutations

https://arxiv.org/pdf/1806.01261.pdf

Structure

GNN

Relevance

State of the art in:

- Quantum/Computational Chemistry (chemical synthesis)
- Citation Prediction
- 3D vision
- Recommender systems
- Visual Question Answering

State of the art in:

- Quantum/Computational Chemistry (chemical synthesis)
- Citation Prediction
- 3D vision
- Recommender systems
- Visual Question Answering

2019 NeurIPS opened a new session called "Graph Representation Learning”

Structure

GNN

Relevance

State of the art in:

- Quantum/Computational Chemistry (chemical synthesis)
- Citation Prediction
- 3D vision
- Recommender systems
- Visual Question Answering

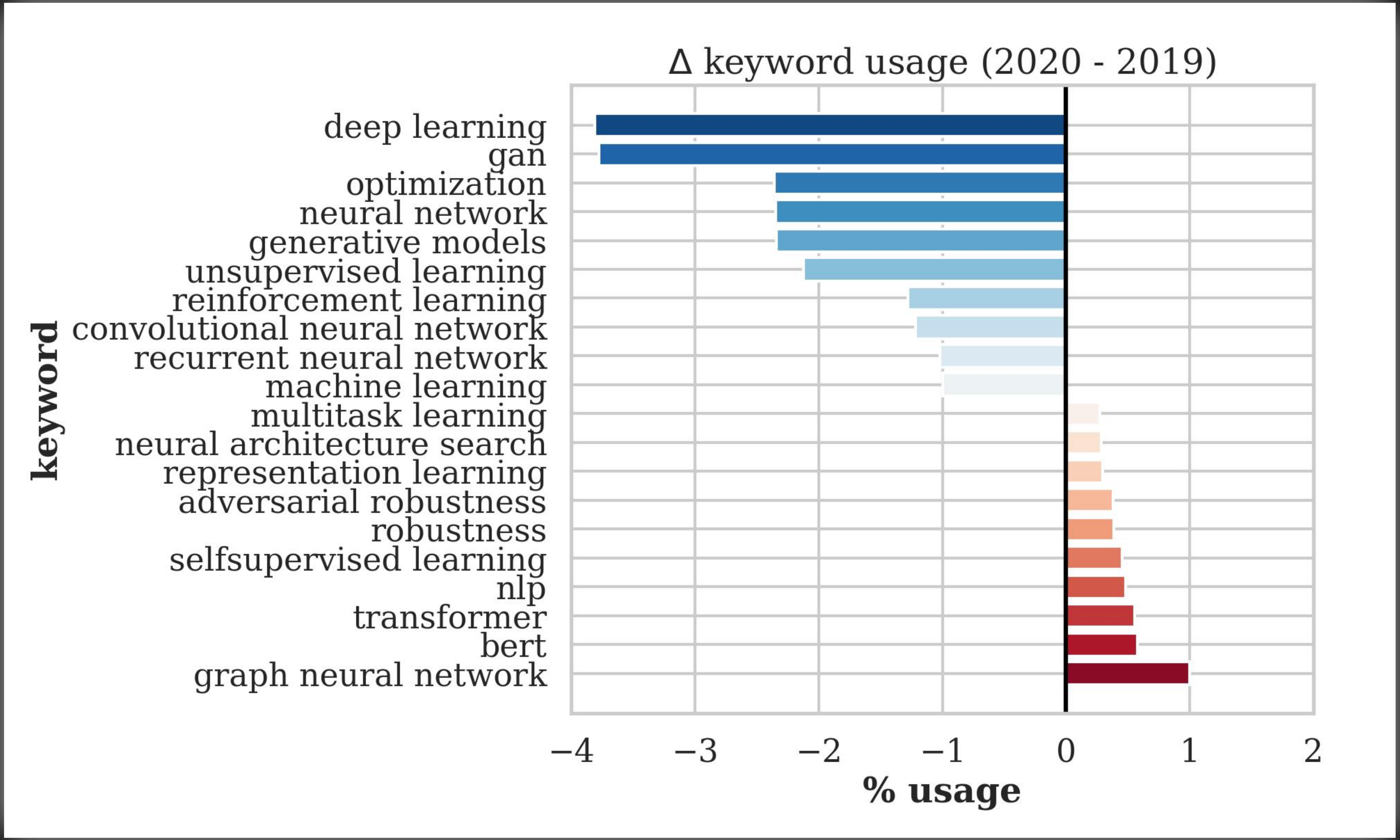
2019 NeurIPS opened a new session called "Graph Representation Learning"

Graph-based methods are gaining prominence…

Structure

GNN

Relevance



https://openreview.net/group?id=ICLR.cc/2020/Conference

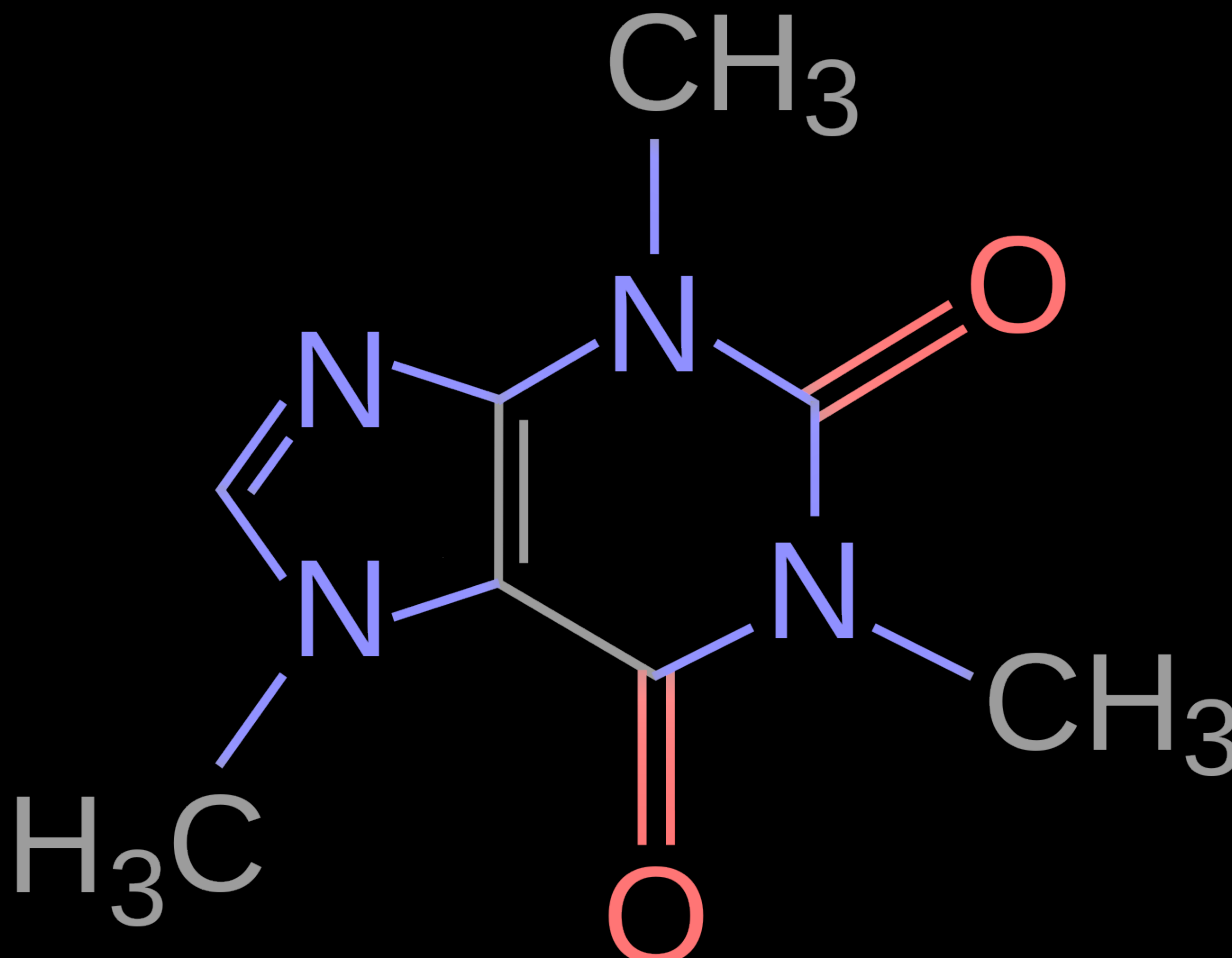
Message Passing Neural Network	Most fundamental kind of GNN
Graph Conv. Network	More recent work, applying principles from CNN architectures in GNNs

Message
Passing
Neural
Network

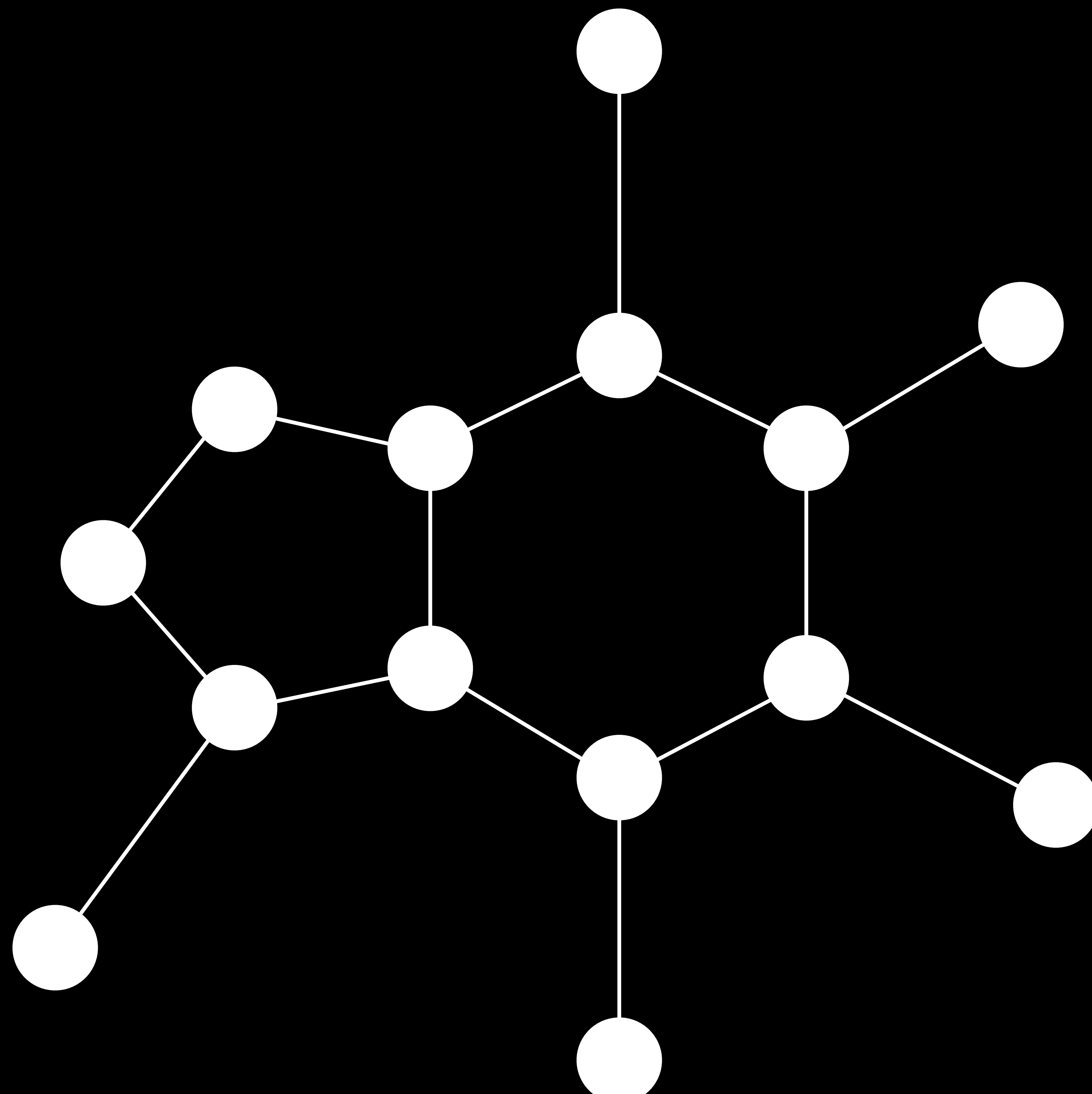
Graph
Conv.
Network

Message
Passing
Neural
Network

Graph
Conv.
Network



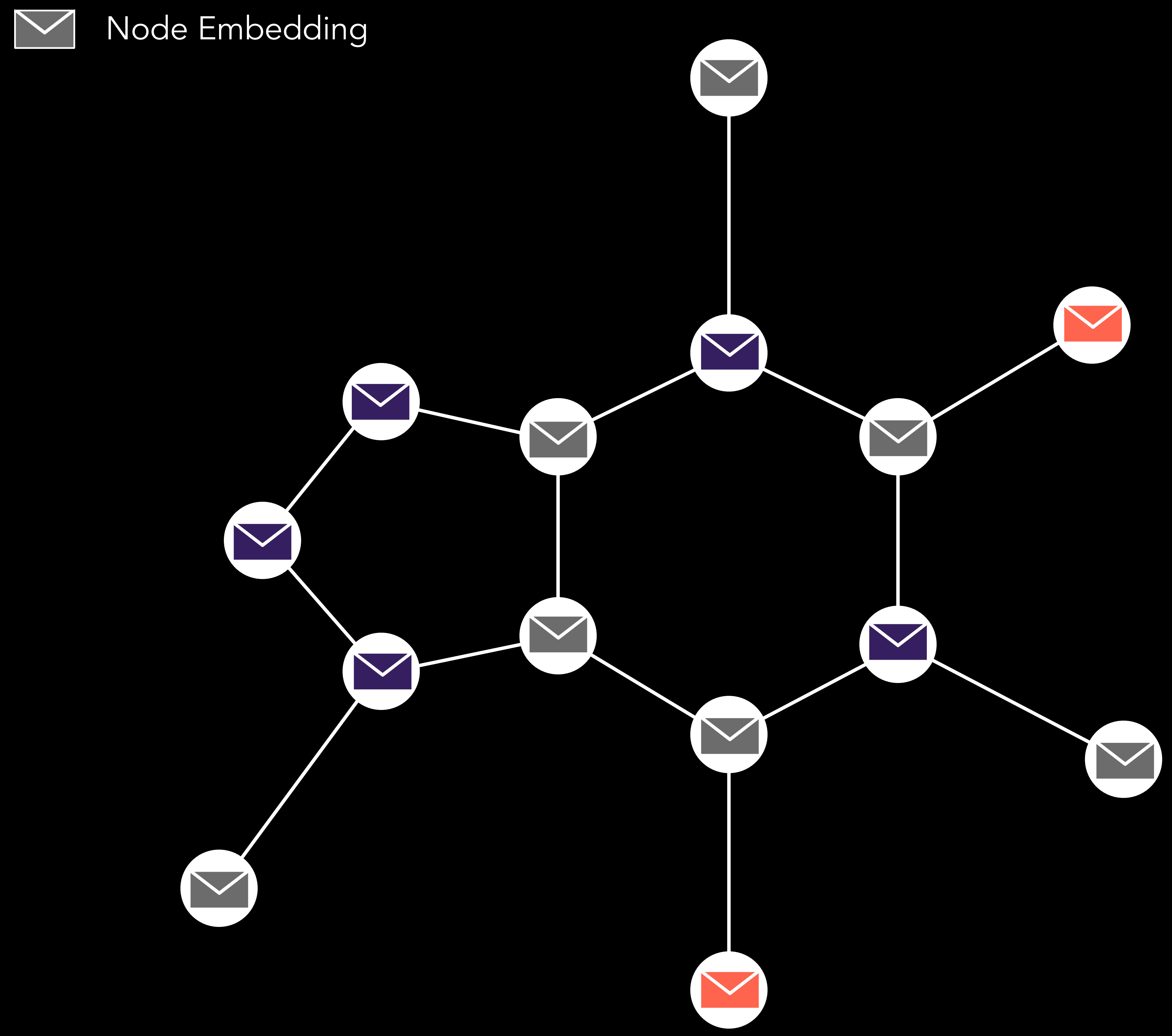
Message
Passing
Neural
Network



Graph
Conv.
Network

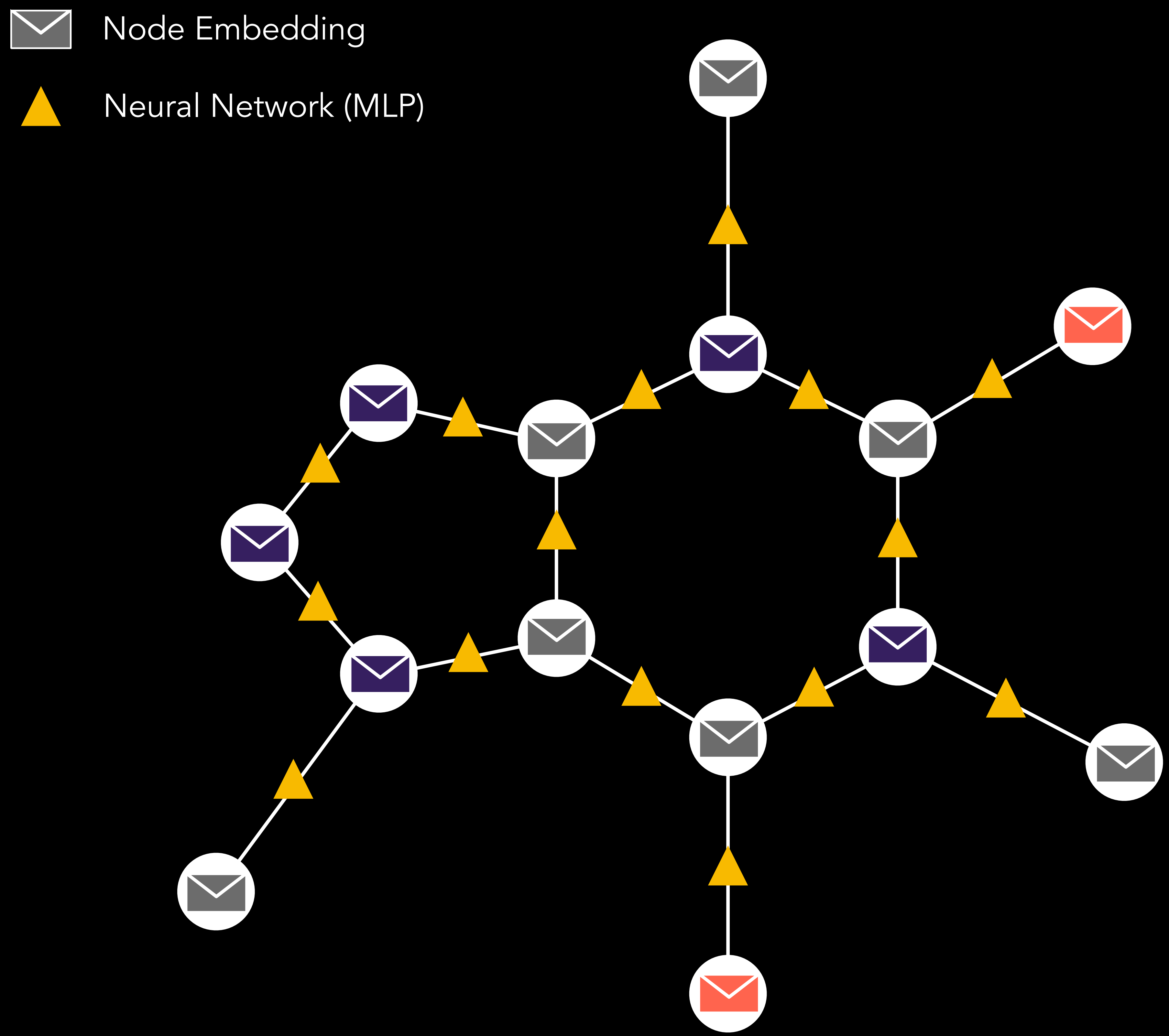
Message
Passing
Neural
Network

Graph
Conv.
Network



Message
Passing
Neural
Network

Graph
Conv.
Network



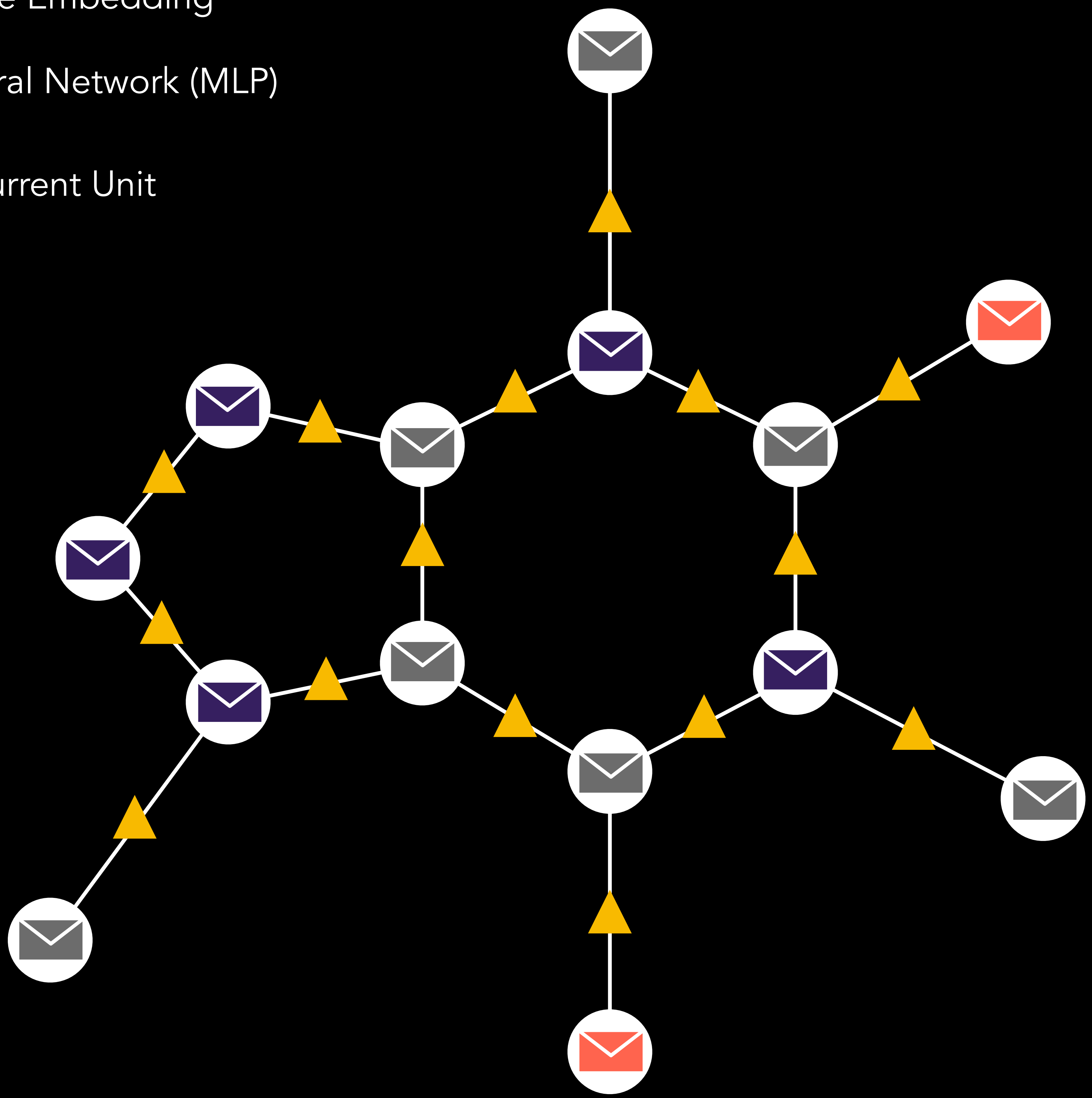
Node Embedding

Neural Network (MLP)

Recurrent Unit

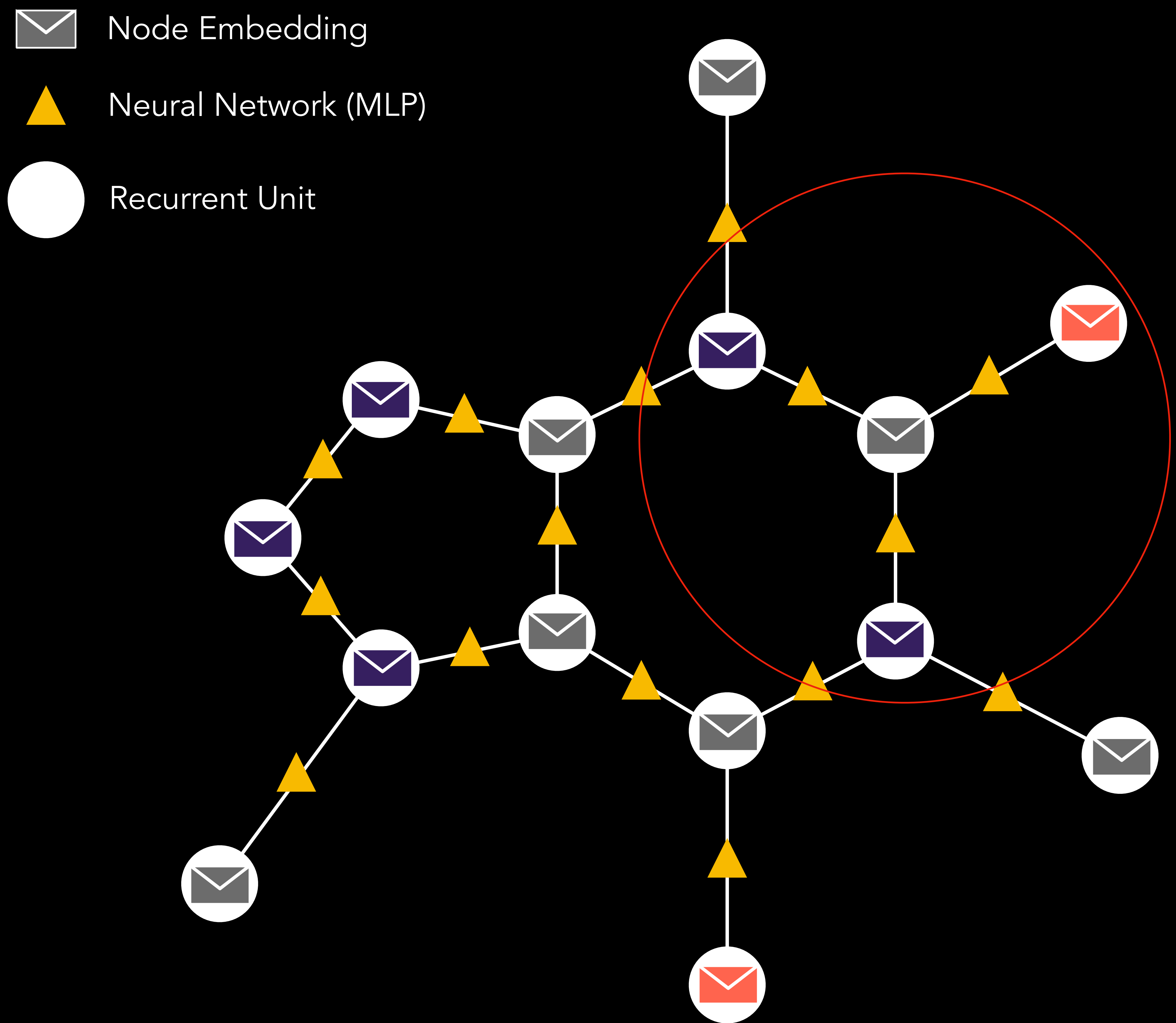
Message
Passing
Neural
Network

Graph
Conv.
Network



Message
Passing
Neural
Network

Graph
Conv.
Network



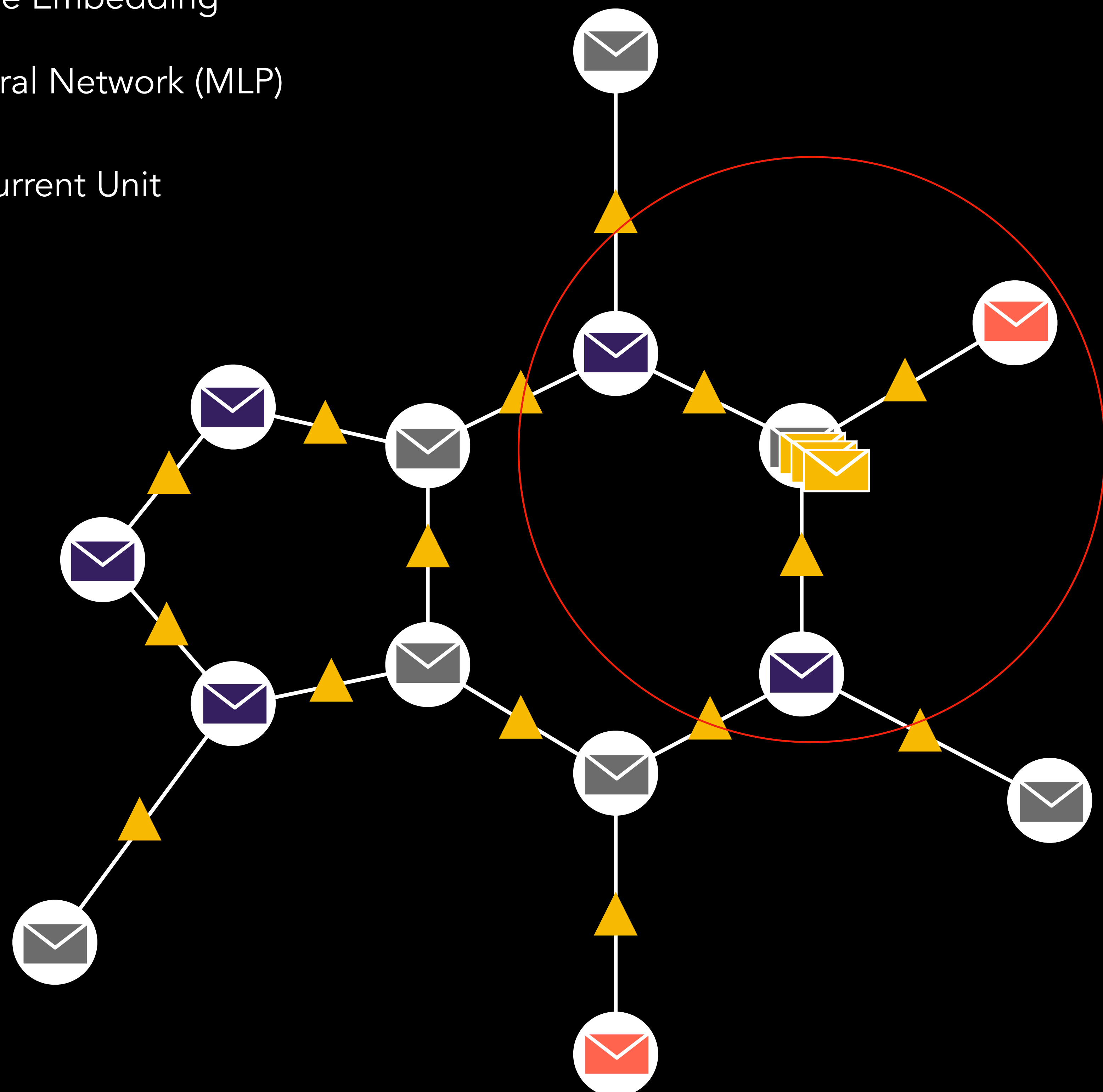
Node Embedding

Neural Network (MLP)

Recurrent Unit

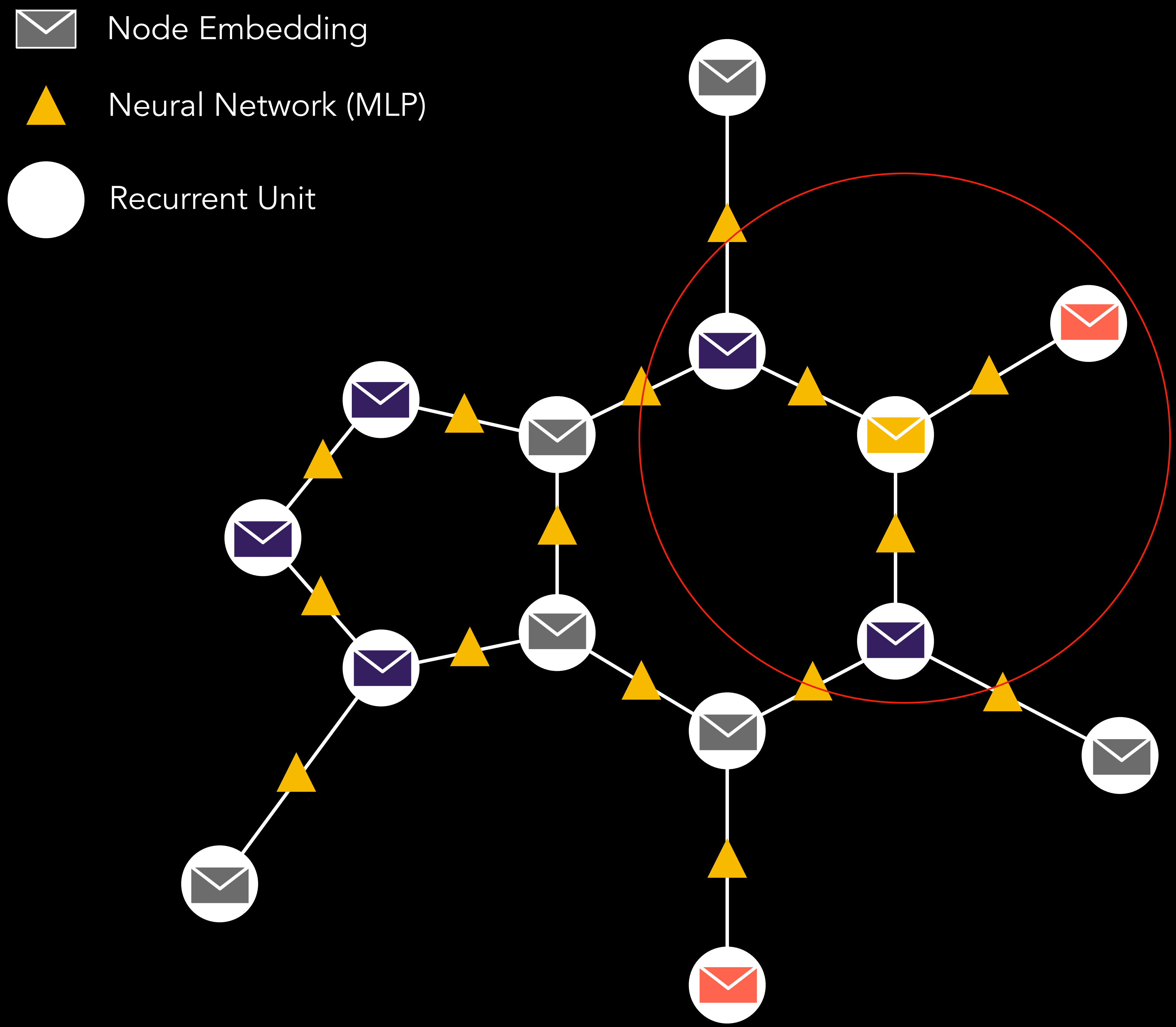
Message
Passing
Neural
Network

Graph
Conv.
Network



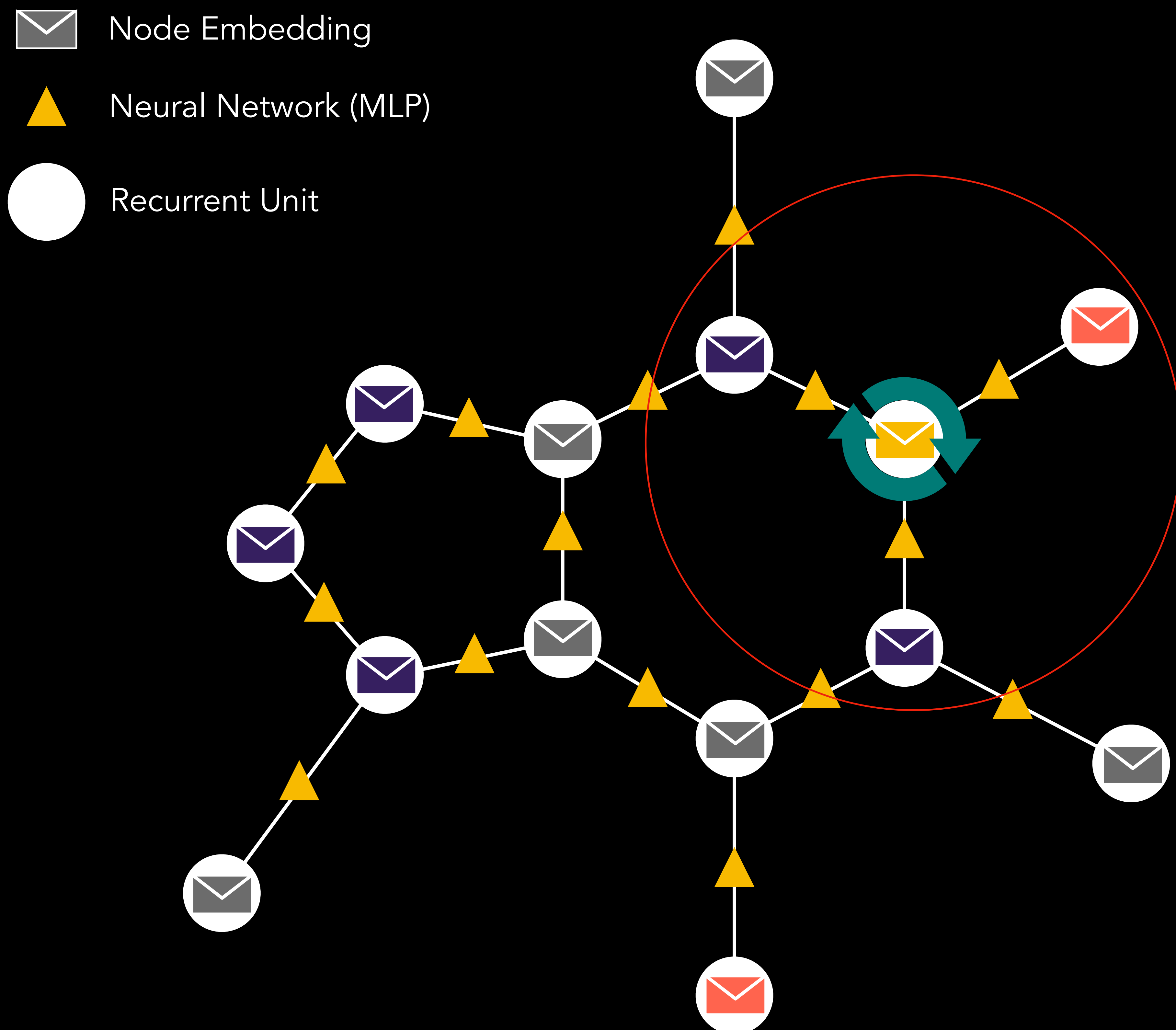
Message
Passing
Neural
Network

Graph
Conv.
Network



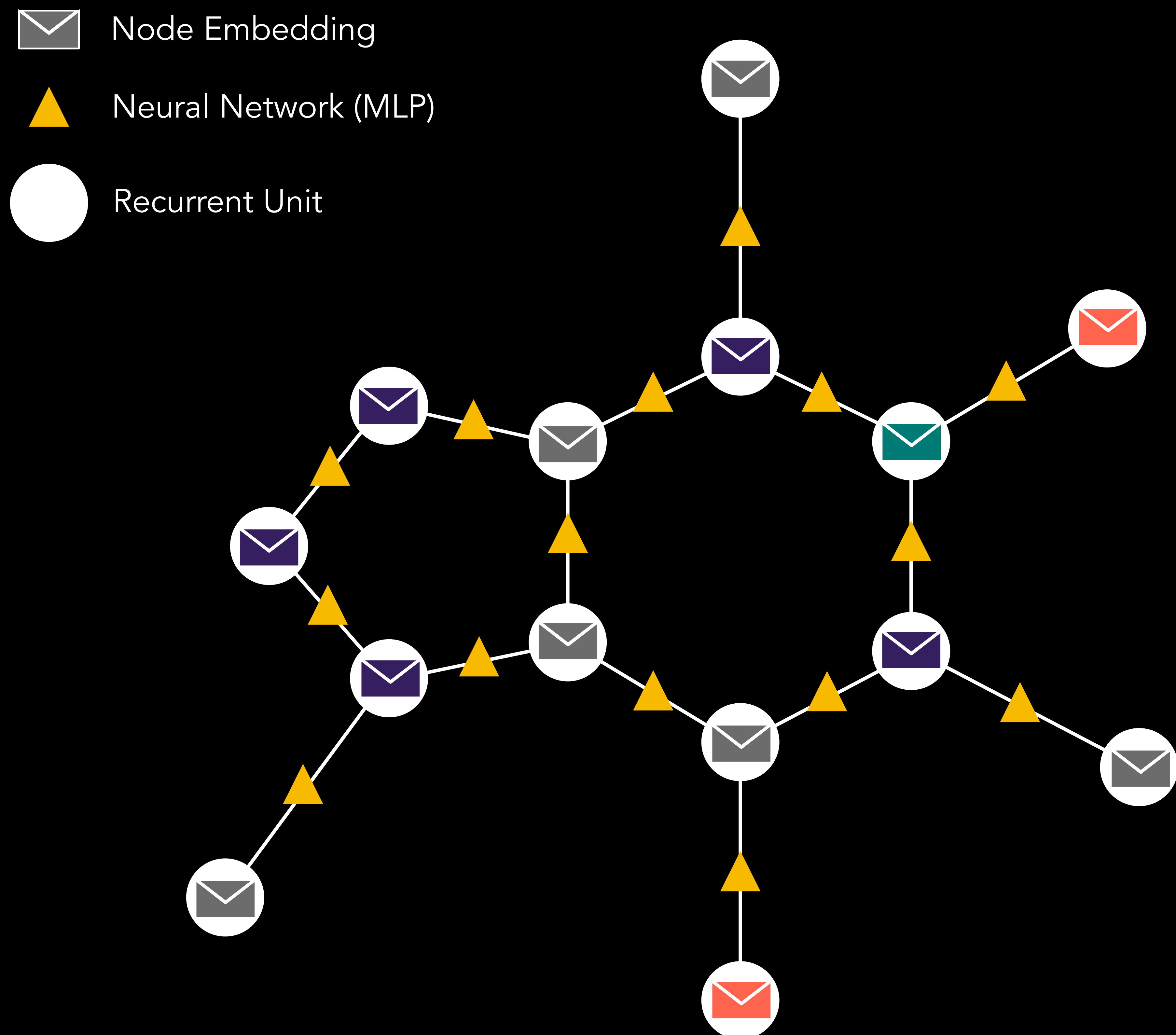
Message
Passing
Neural
Network

Graph
Conv.
Network



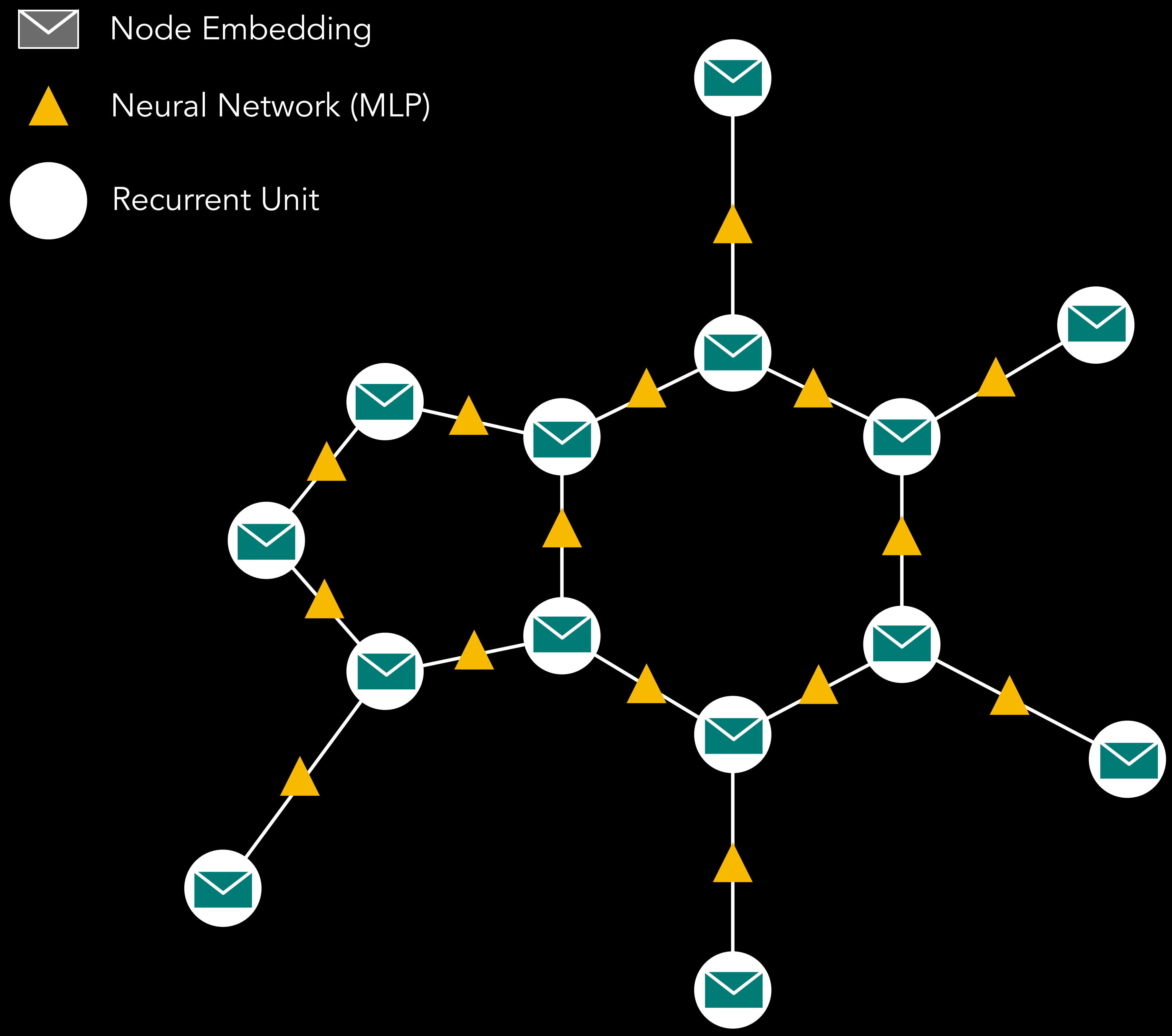
Message
Passing
Neural
Network

Graph
Conv.
Network



Message
Passing
Neural
Network

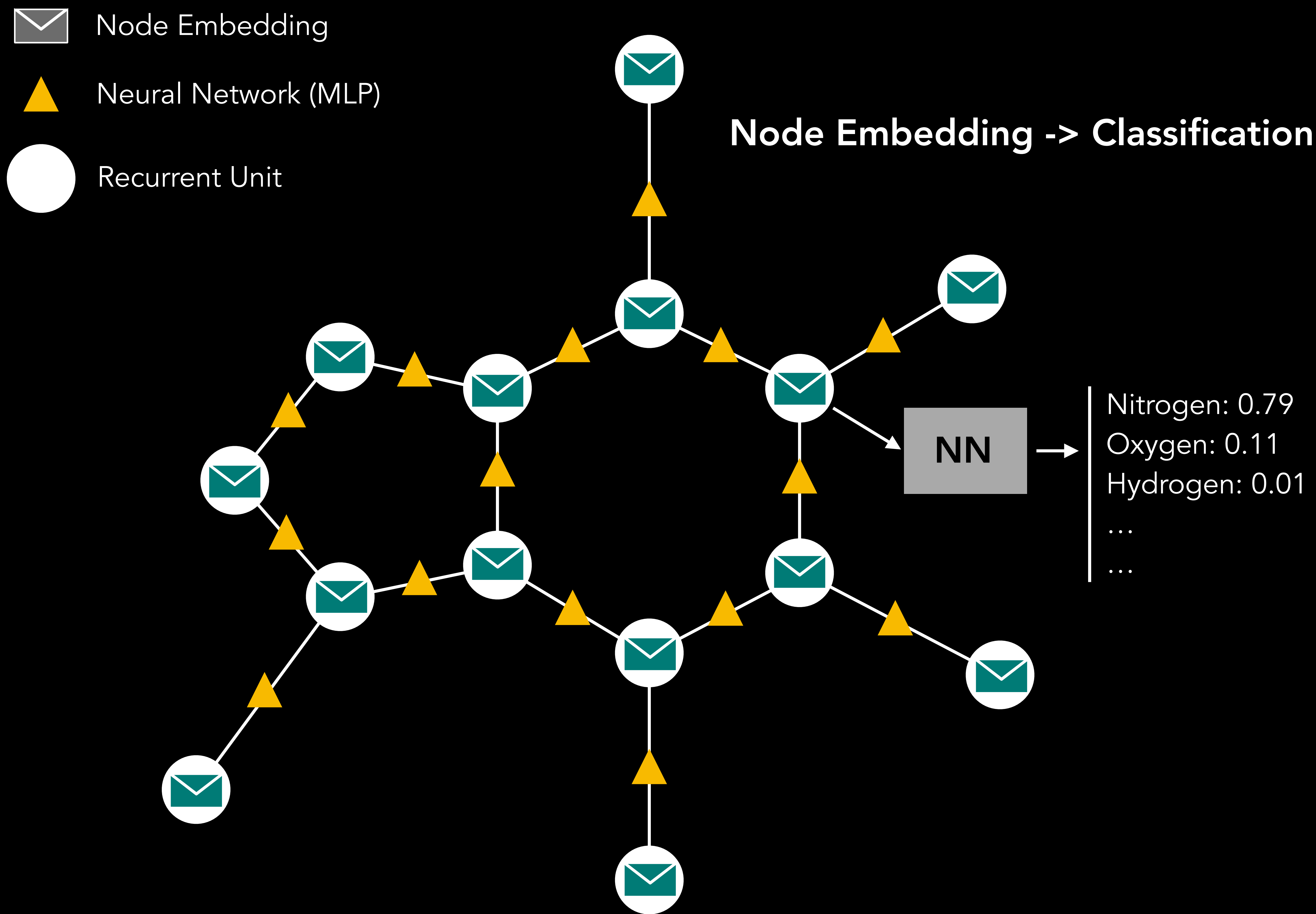
Graph
Conv.
Network



Message
Passing
Neural
Network

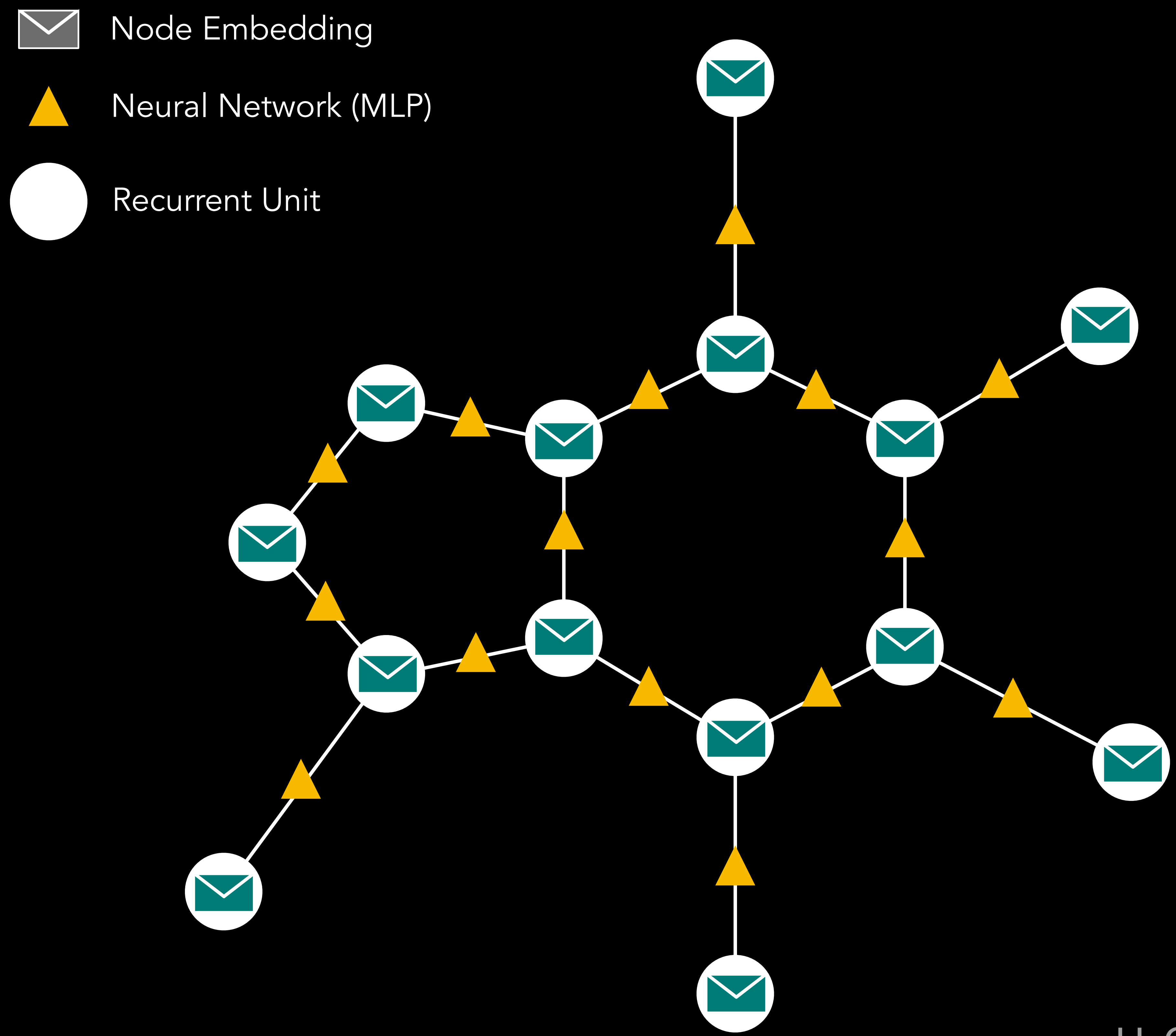
Graph
Conv.
Network

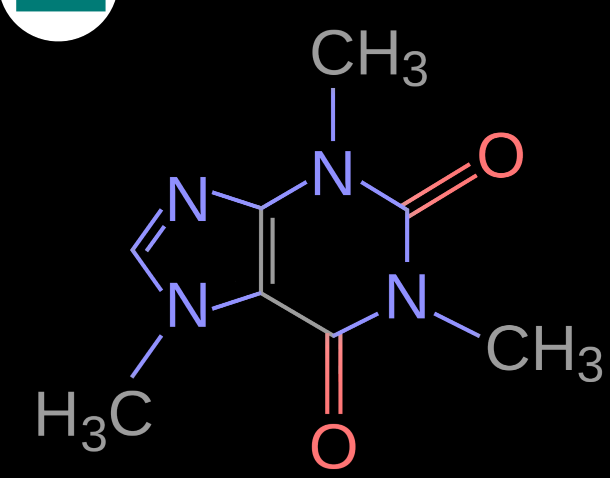
Node Embedding -> Classification



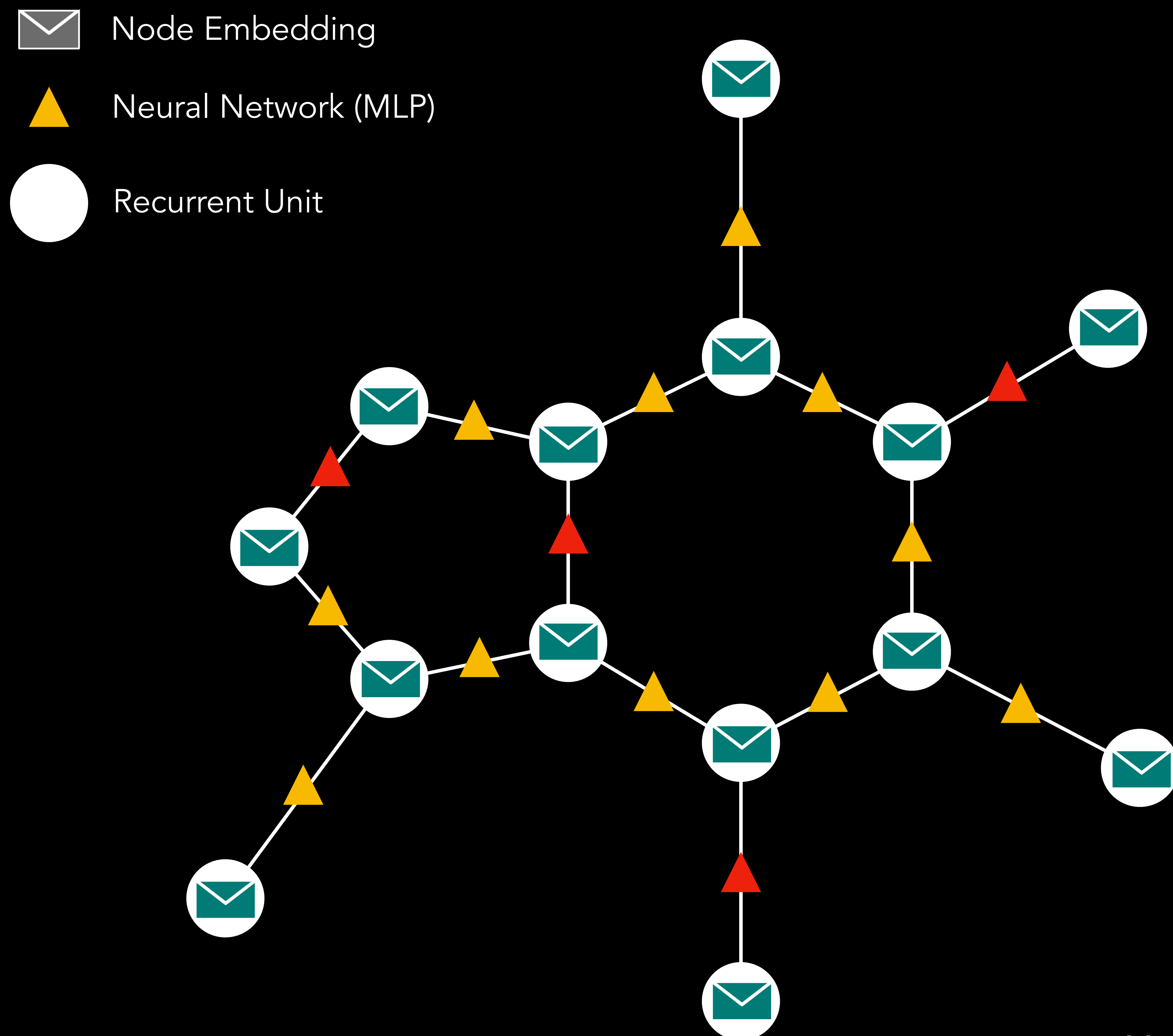
Message
Passing
Neural
Network

Graph
Conv.
Network

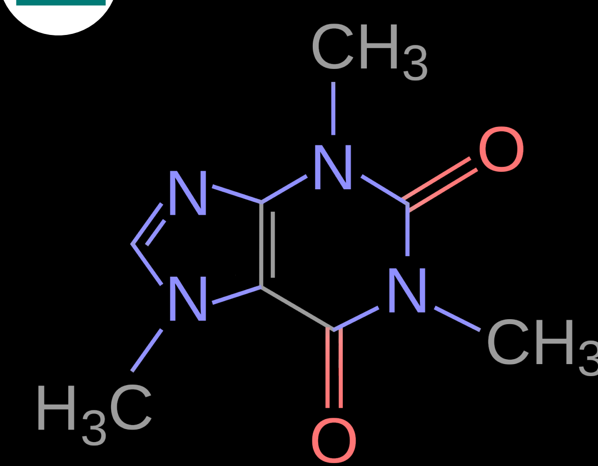




Message
Passing
Neural
Network



Graph
Conv.
Network



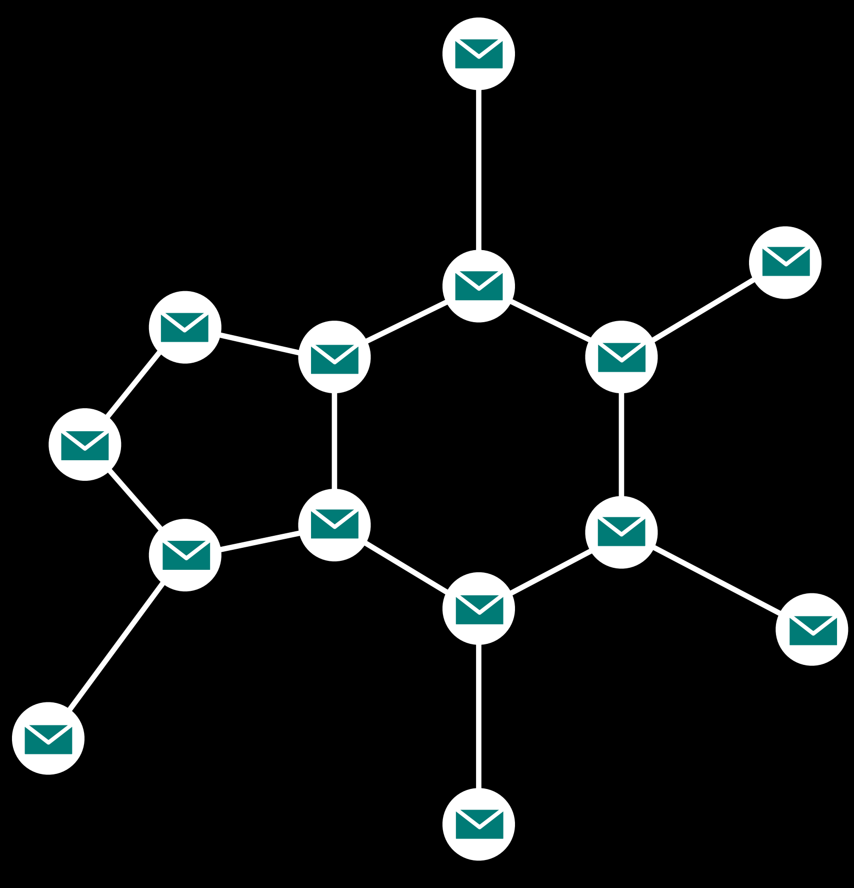
Node Embedding

Neural Network (MLP)

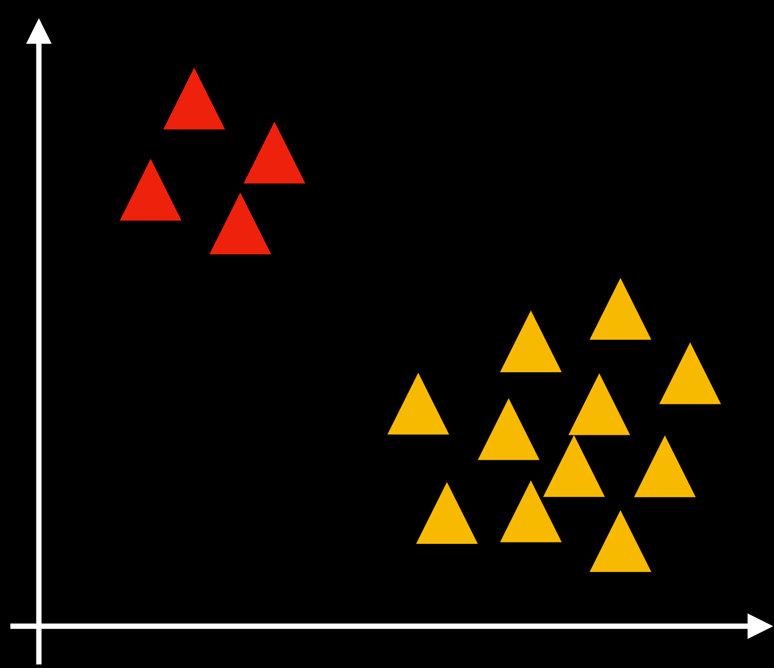
Recurrent Unit

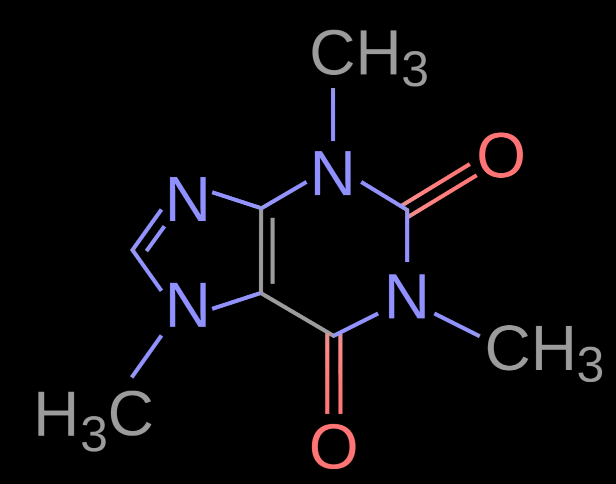
Message
Passing
Neural
Network

Graph
Conv.
Network



Edge Classification/Clustering





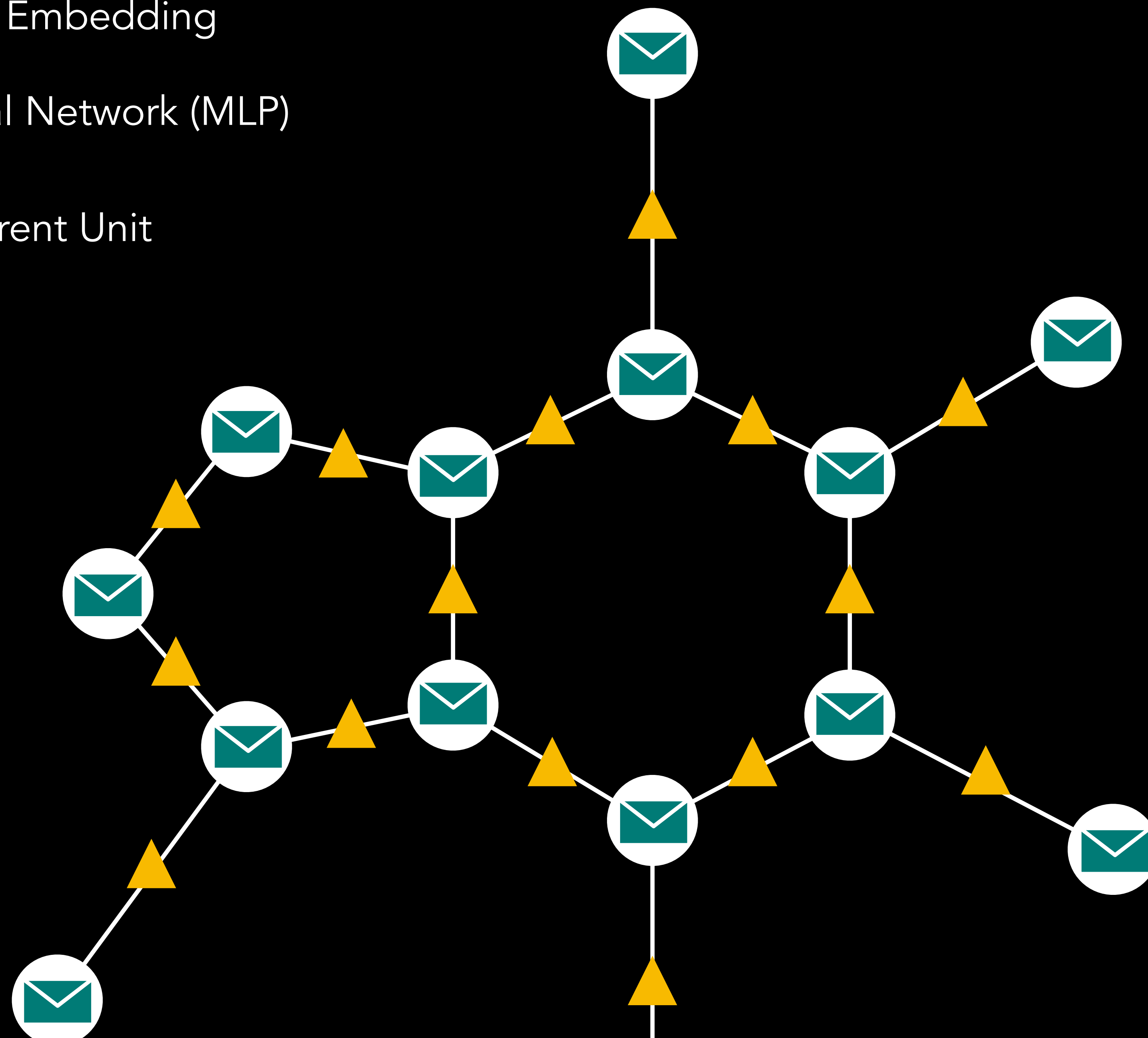
Node Embedding

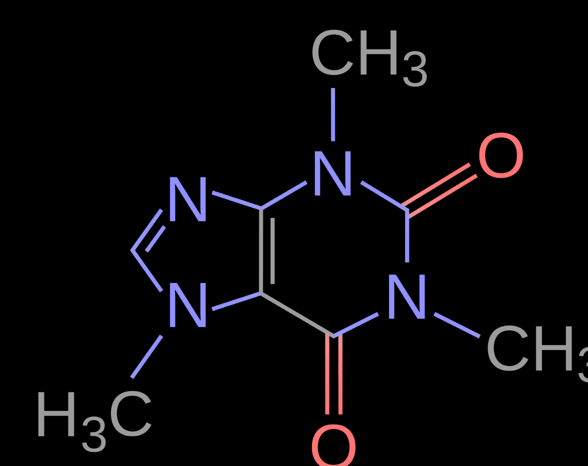
Neural Network (MLP)

Recurrent Unit

Message
Passing
Neural
Network

Graph
Conv.
Network



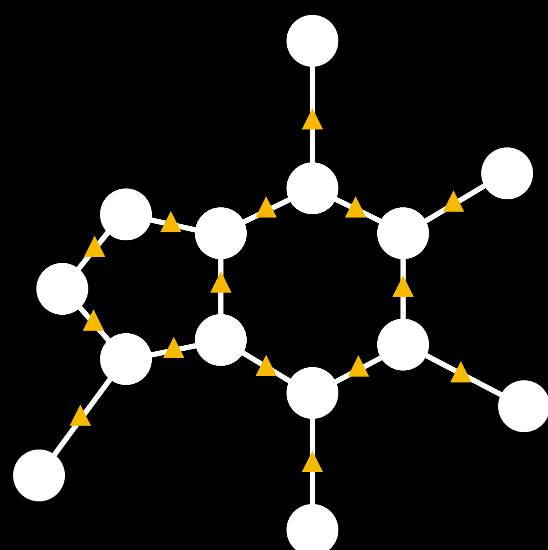


Node Embedding

Neural Network (MLP)

Recurrent Unit

Message
Passing
Neural
Network

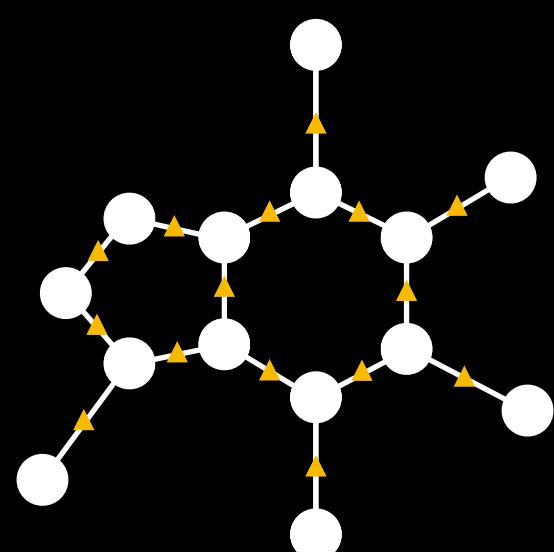


Graph
Conv.
Network

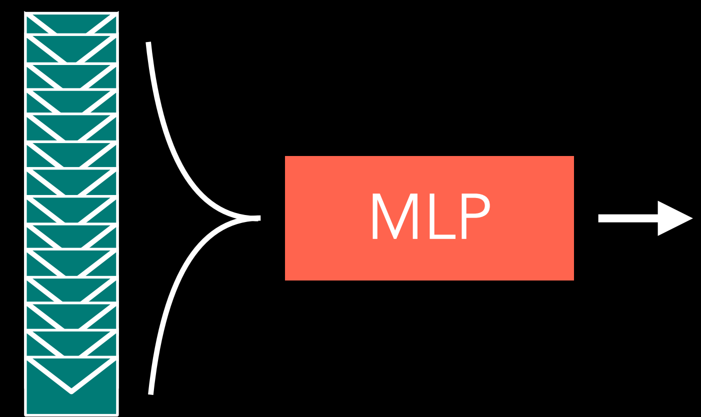
Node Embedding

Neural Network (MLP)

Recurrent Unit



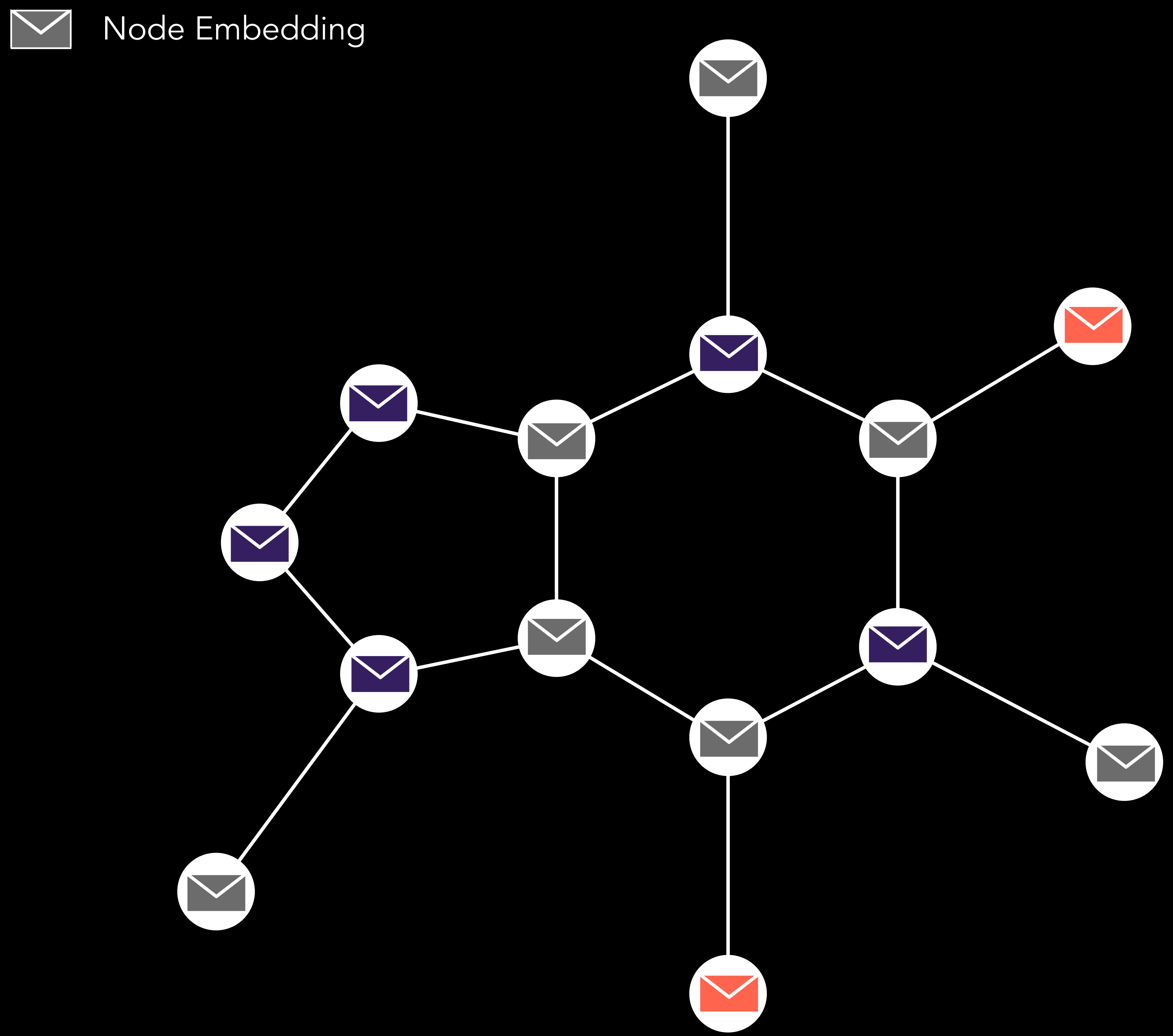
Graph Embedding -> Classification



Caffeine: 0.85
Dopamine: 0.03
Serotonin: 0.01
…
…

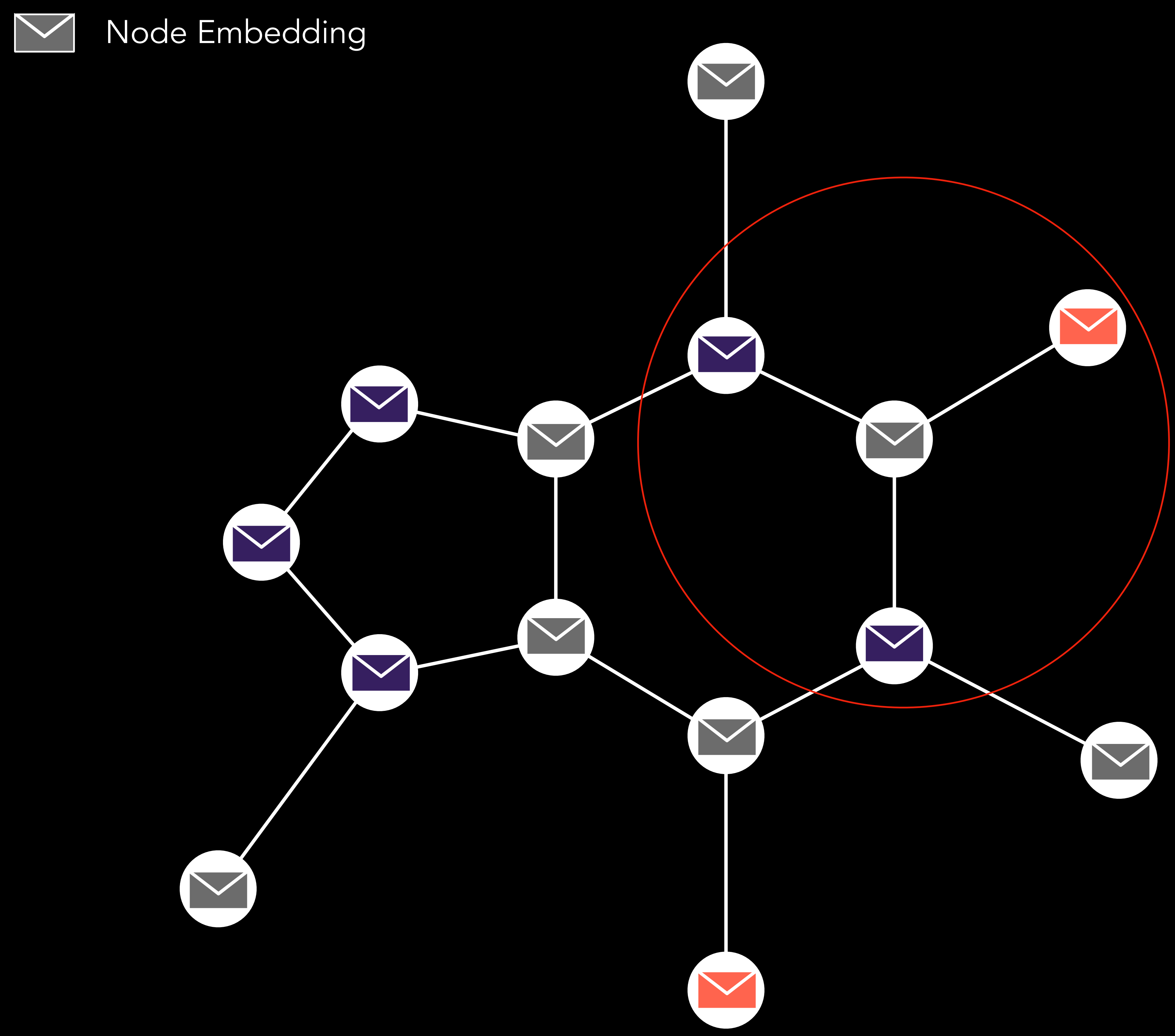
Message
Passing
Neural
Network

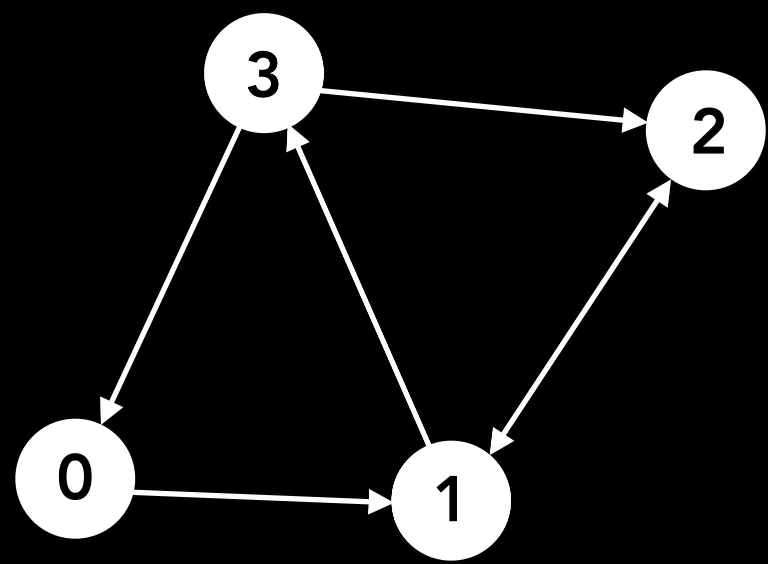
Graph
Conv.
Network



Message
Passing
Neural
Network

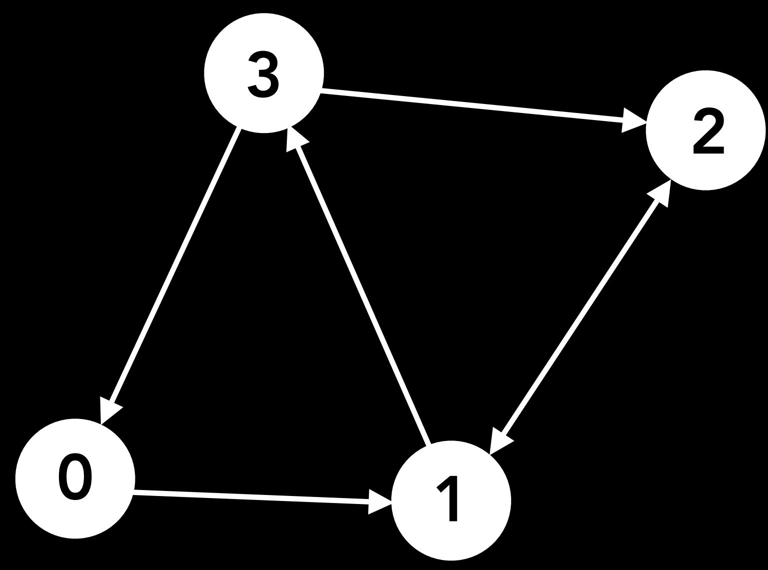
Graph
Conv.
Network





Message
Passing
Neural
Network

Graph
Conv.
Network

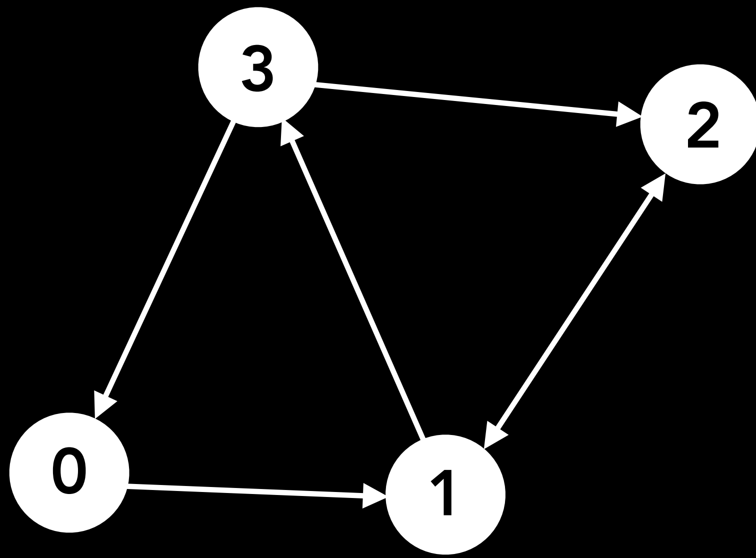


$$H^{(l+1)} = \sigma\left(\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}H^{(l)}W^{(l)}\right)$$

Message
Passing
Neural
Network

Graph
Conv.
Network

Message
Passing
Neural
Network



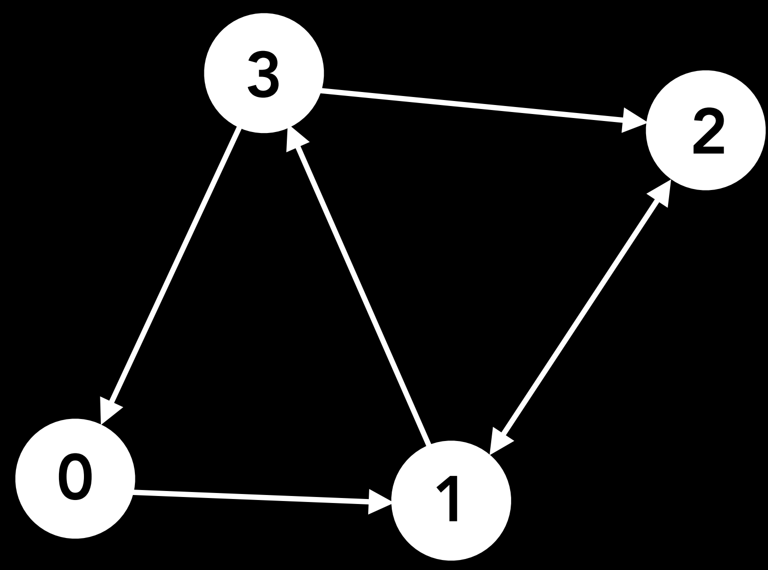
$$H^{(l+1)} = \sigma\left(\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}H^{(l)}W^{(l)}\right)$$

Graph
Conv.
Network

```
A = np.matrix([
     [0, 1, 0, 0],
     [0, 0, 1, 1],
     [0, 1, 0, 0],
     [1, 0, 1, 0]],
    dtype=float)
```

Message
Passing
Neural
Network

Graph
Conv.
Network



$$H^{(l+1)} = \sigma\left(\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}H^{(l)}W^{(l)}\right)$$

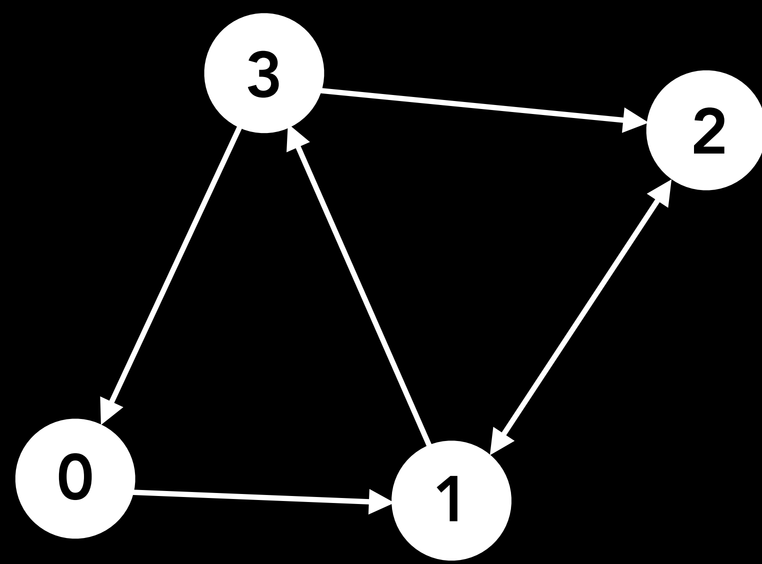
```
A = np.matrix([
     [0, 1, 0, 0],
     [0, 0, 1, 1],
     [0, 1, 0, 0],
     [1, 0, 1, 0]],
    dtype=float)
```

Features:

```
X = matrix([[ 0.,  0.],
            [ 1., -1.],
            [ 2., -2.],
            [ 3., -3.]])
```

Message
Passing
Neural
Network

Graph
Conv.
Network



$$H^{(l+1)} = \sigma\left(\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}H^{(l)}W^{(l)}\right)$$

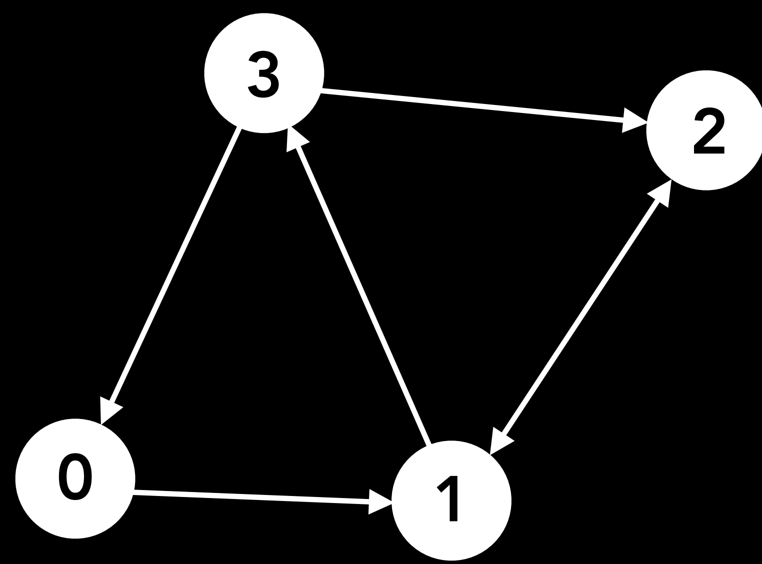
Features:

```
A = np.matrix([
     [0, 1, 0, 0],
     [0, 0, 1, 1],
     [0, 1, 0, 0],
     [1, 0, 1, 0]],
    dtype=float)
```

```
X = matrix([[ 0.,  0.],
            [ 1., -1.],
            [ 2., -2.],
            [ 3., -3.]])
```

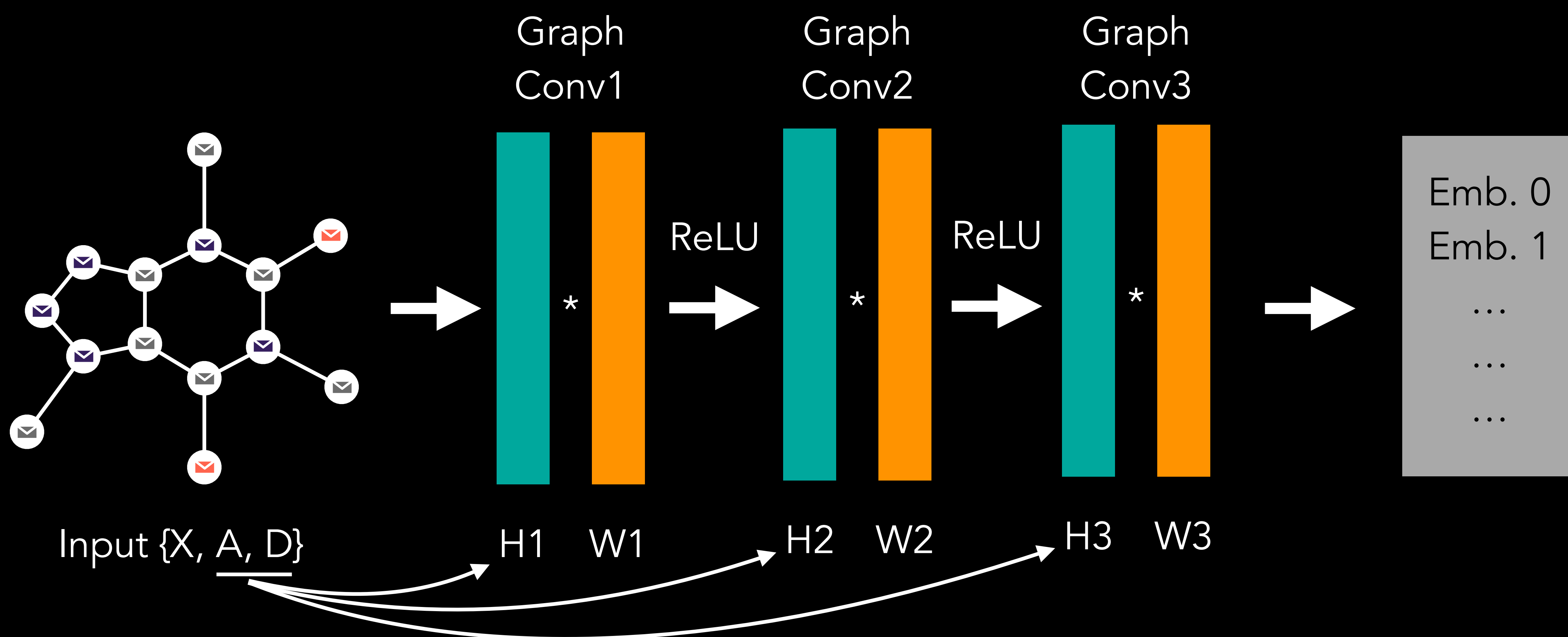
```
A*X = matrix([[ 1., -1.],
              [ 5., -5.],
              [ 1., -1.],
              [ 2., -2.]]
```

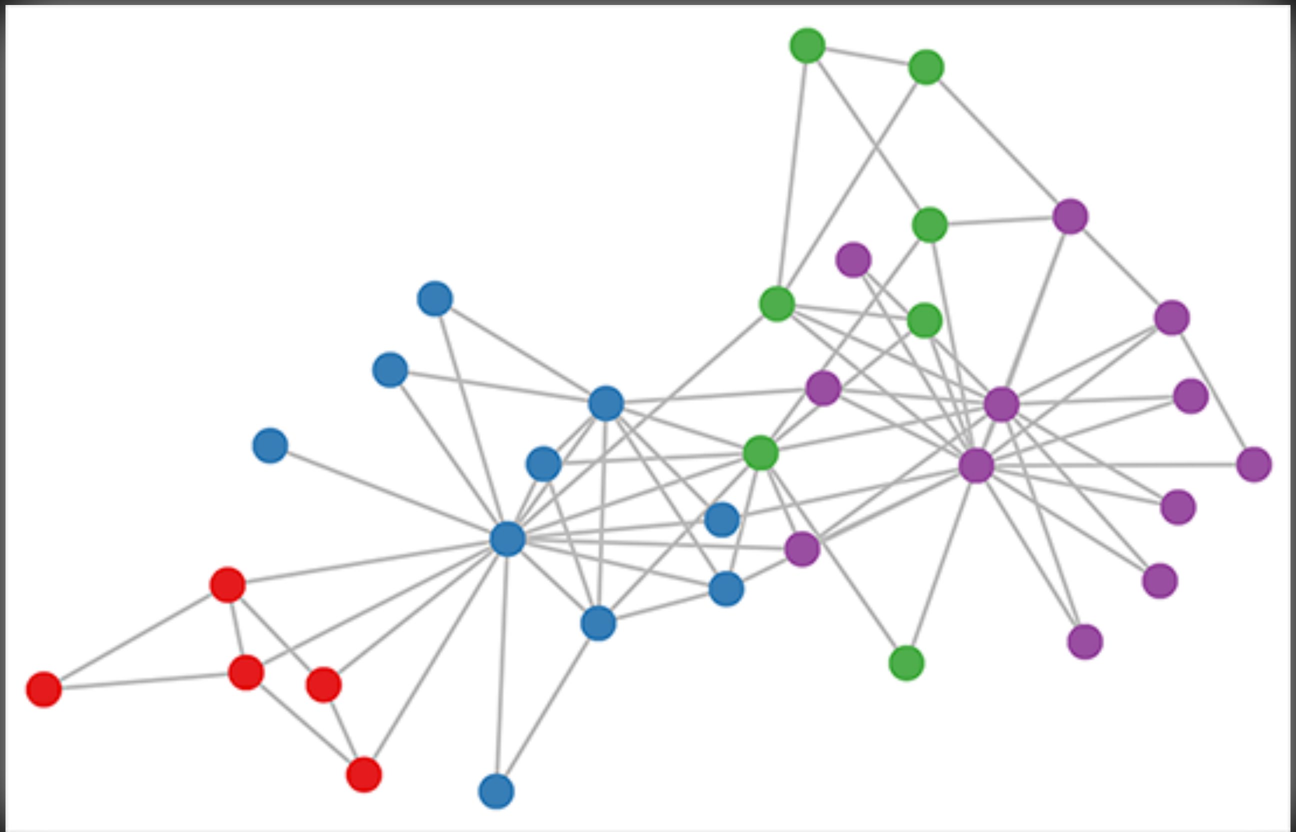
Message
Passing
Neural
Network



$$H^{(l+1)} = \sigma\left(\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}H^{(l)}W^{(l)}\right)$$

Graph
Conv.
Network





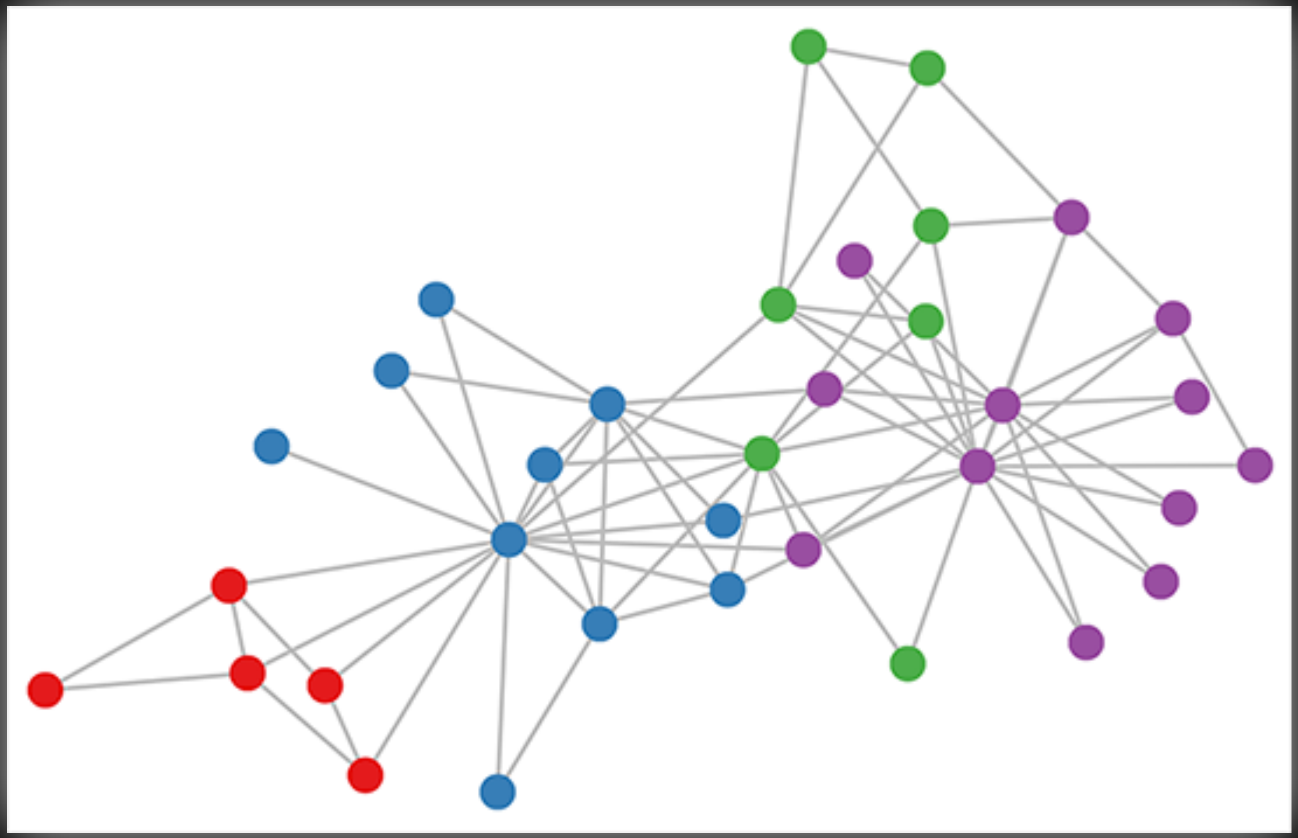
Message
Passing
Neural
Network

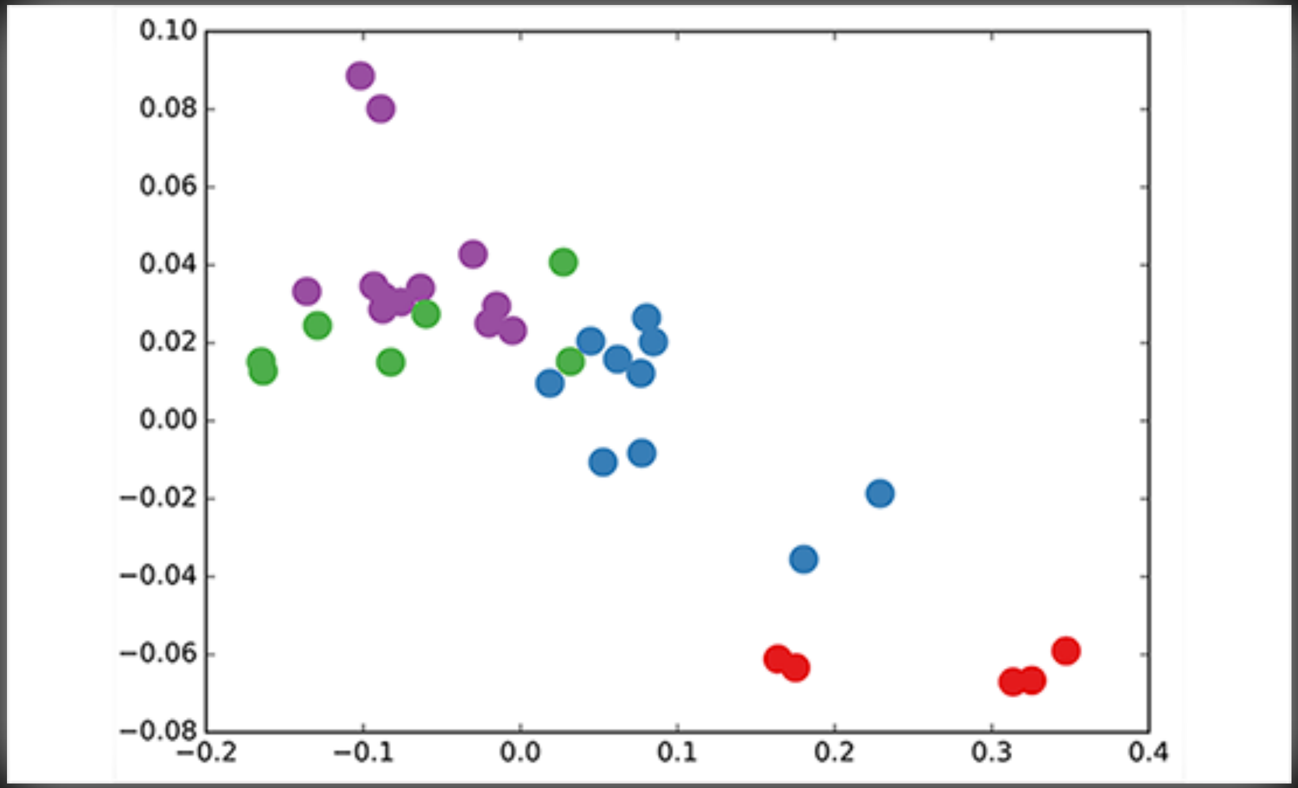
Graph
Conv.
Network

https://tkipf.github.io/graph-convolutional-networks/

Message
Passing
Neural
Network

Graph
Conv.
Network





https://tkipf.github.io/graph-convolutional-networks/

NRI	Neural Relational Inference
Decagon	Polypharmacy prediction
Relational Reasoning	Review graph-based approaches
Implementation	Popular frameworks + datasets

NRI

Goals:

1) learn to infer the latent interaction graph
2) learn dynamics of the interacting system using 1)
3) complete 1) and 2) using only object trajectories as input

Decagon

Relational Reasoning

Implementation

NRI
Decagon
Relational Reasoning
Implementation

Goals:

1) learn to infer the latent interaction graph
2) learn dynamics of the interacting system using 1)
3) complete 1) and 2) using only object trajectories as input

Data:

1) Simulated object trajectories (masses on springs, charged particles, phase coupled oscillators)

NRI

Decagon

Relational Reasoning

Implementation

Goals:

1) learn to infer the latent interaction graph
2) learn dynamics of the interacting system using 1)
3) complete 1) and 2) using only object trajectories as input

Data:

1) Simulated object trajectories (masses on springs, charged particles, phase coupled oscillators)

Model:

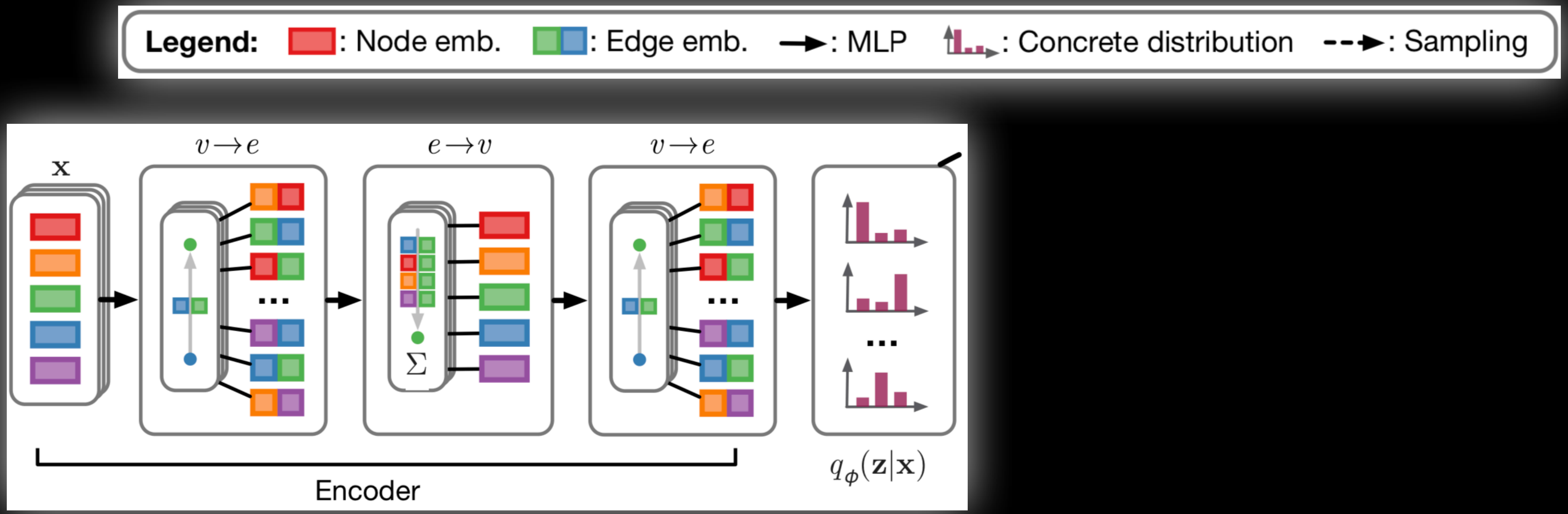
1) Encoder which predicts interactions/types given trajectories
2) Decoder that learns the dynamical model given the interaction graph

NRI

Decagon

Relational Reasoning

Implementation

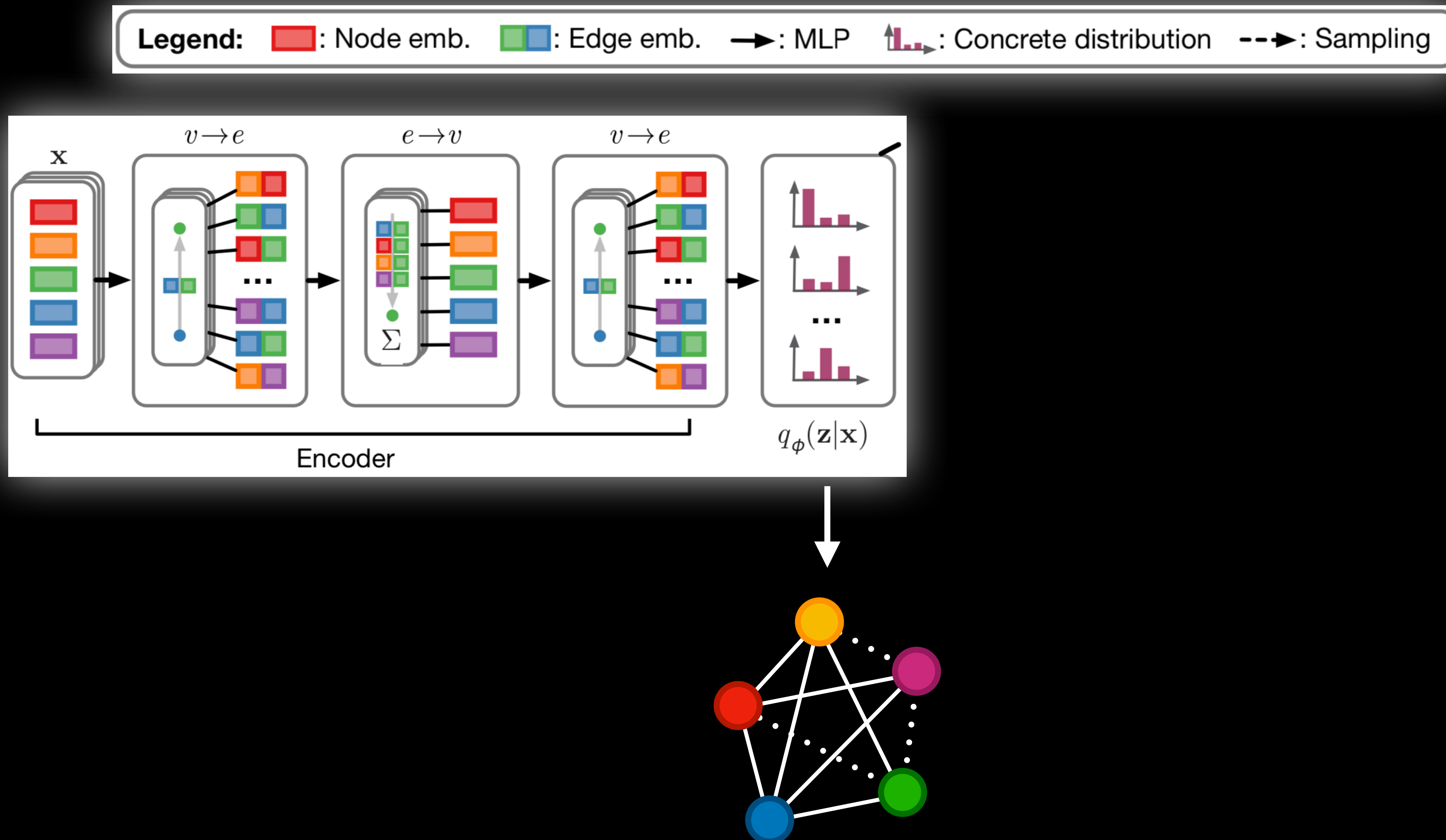


NRI

Decagon

Relational Reasoning

Implementation

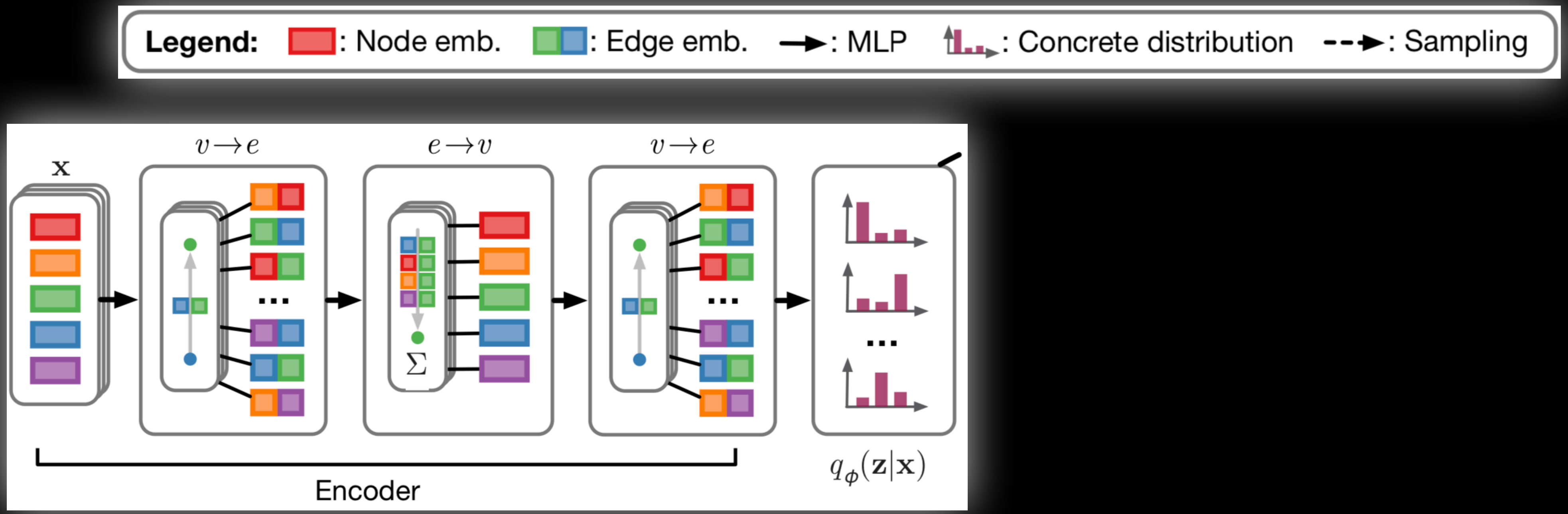


NRI

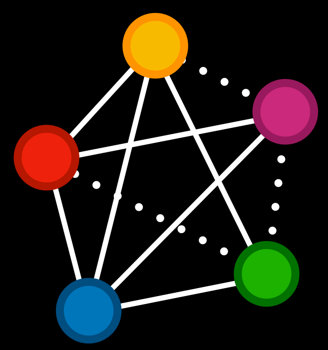
Decagon

Relational Reasoning

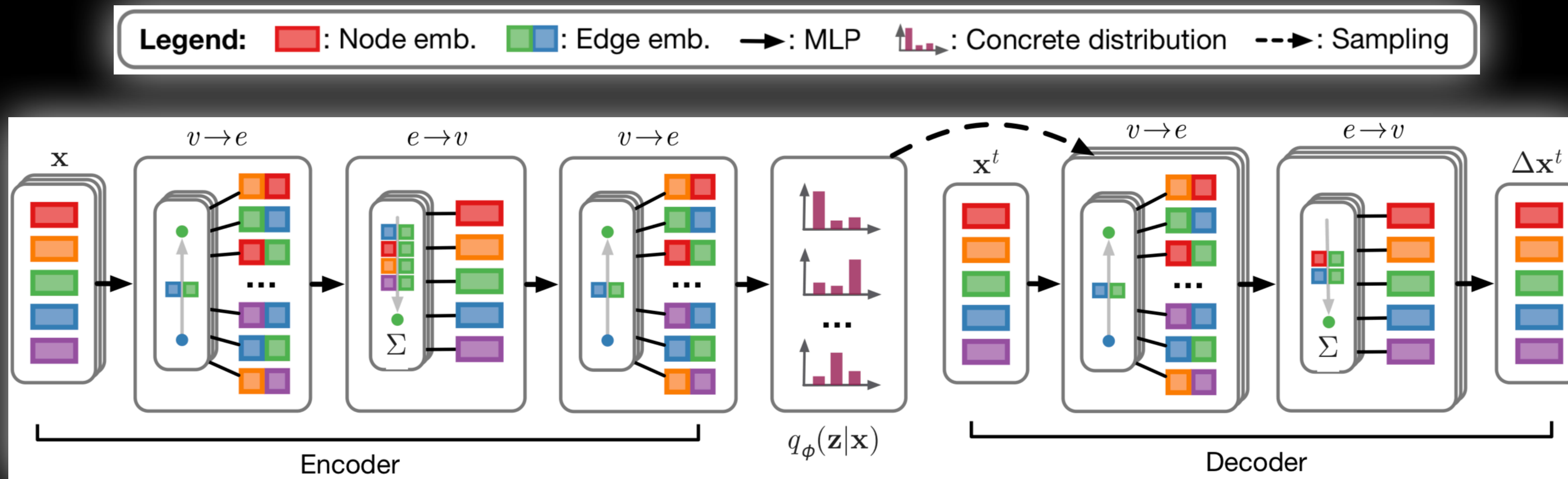
Implementation



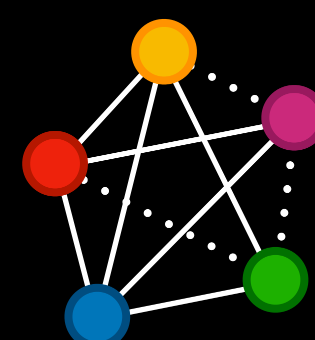
Learned Regularizer



NRI
Decagon
Relational Reasoning
Implementation



Learned Regularizer

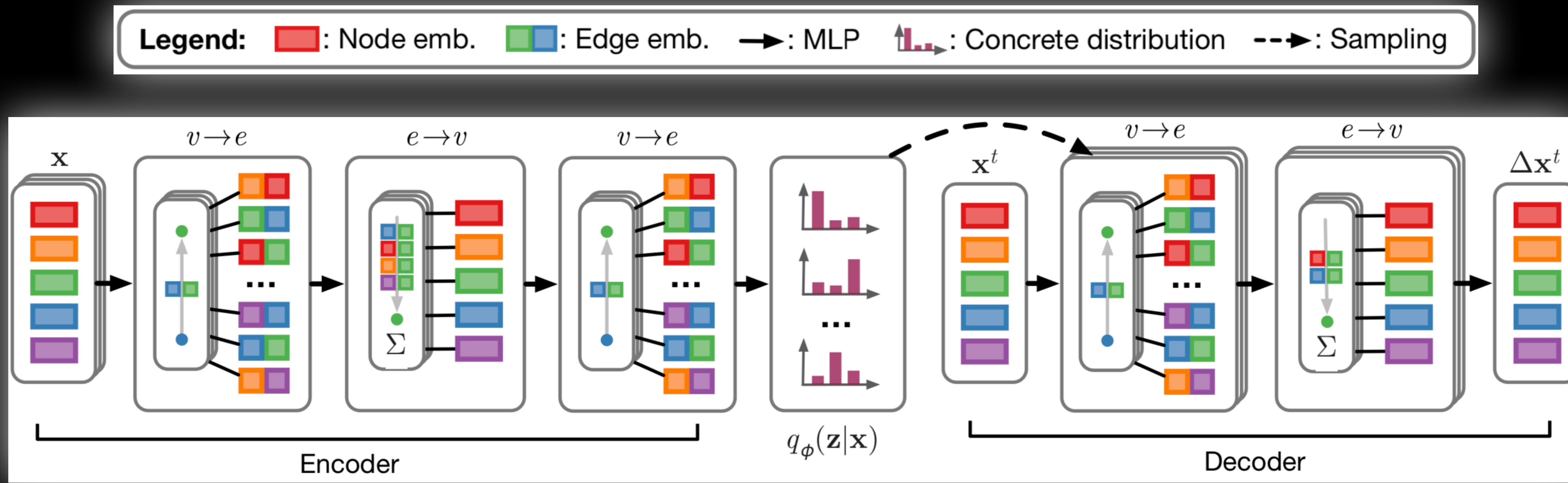


NRI

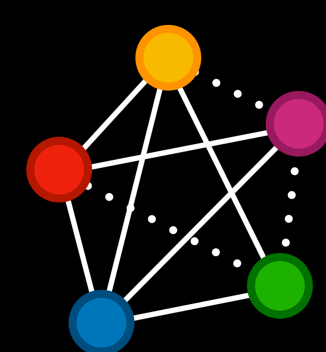
Decagon

Relational Reasoning

Implementation



Learned Regularizer

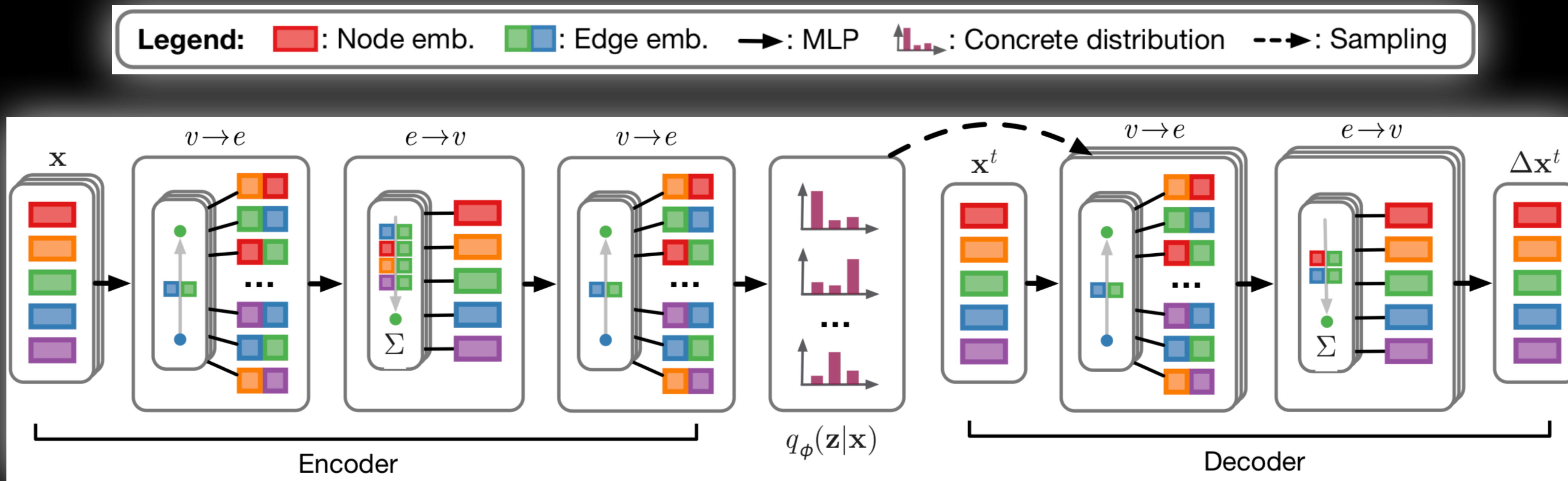


NRI

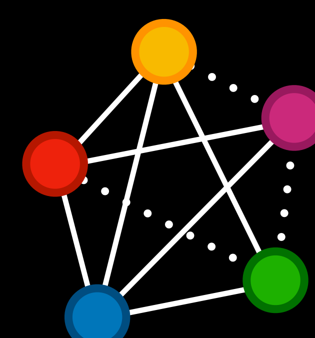
Decagon

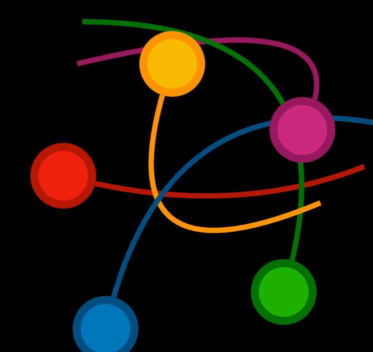
Relational Reasoning

Implementation

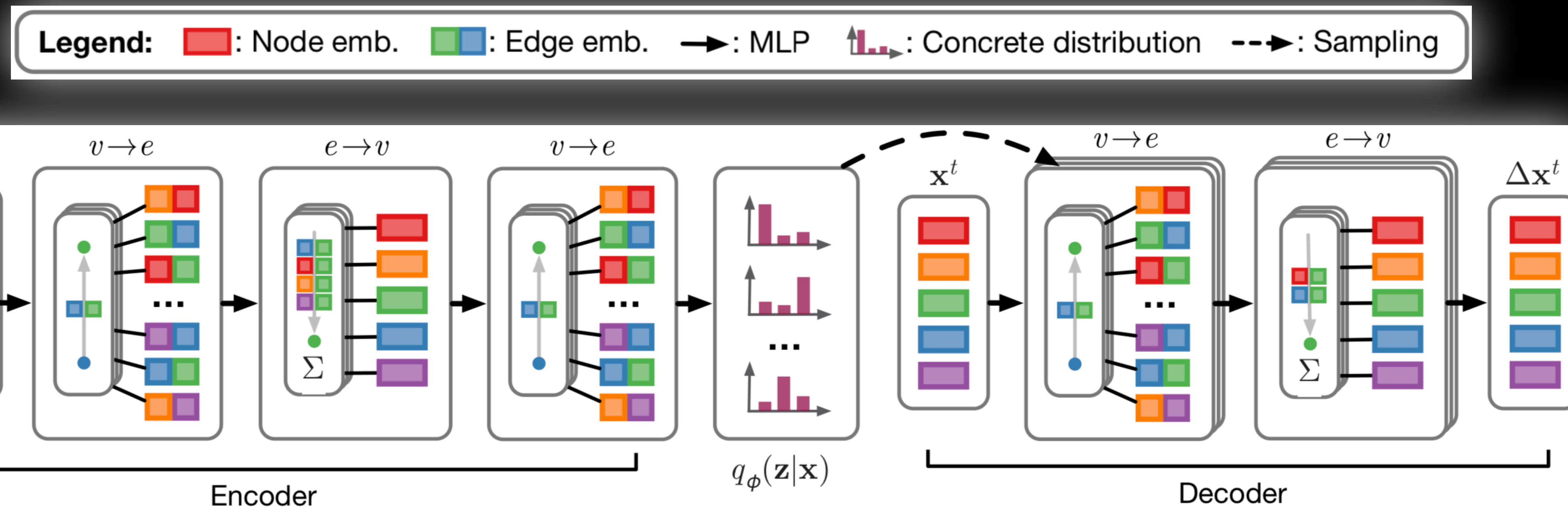


Learned Regularizer

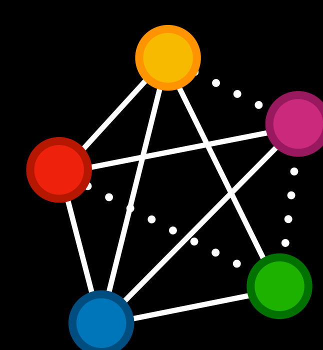


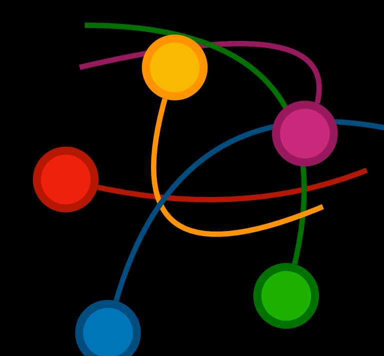


Training Signal:



Learned Regularizer





Training Signal:

Table 1. Accuracy (in %) of unsupervised interaction recovery.

Model	Springs	Charged	Kuramoto
	5 objects		
Corr. (path)	52.4±0.0	55.8±0.0	62.8±0.0
Corr. (LSTM)	52.7±0.9	54.2±2.0	54.4±0.5
NRI (sim.)	**99.8**±0.0	59.6±0.8	–
NRI (learned)	**99.9**±0.0	**82.1**±0.6	**96.0**±0.1

Goal:

1) learn to predict polypharmacy side effects
2) flag and prioritize polypharmacy side effects for follow-up analysis via formal pharmacological studies.

Goal:

1) learn to predict polypharmacy side effects
2) flag and prioritize polypharmacy side effects for follow-up analysis via formal pharmacological studies.

Data:

1) multimodal graph of protein-protein interactions, drug-protein target interactions, and the polypharmacy side effects, which are represented as **drug-drug interactions, where each side effect is an edge of a different type**.

NRI

Decagon

Relational Reasoning

Implementation

Goal:

1) learn to predict polypharmacy side effects
2) flag and prioritize polypharmacy side effects for follow-up analysis via formal pharmacological studies.

Data:

1) multimodal graph of protein-protein interactions, drug-protein target interactions, and the polypharmacy side effects, which are represented as **drug-drug interactions, where each side effect is an edge of a different type**.

Model:

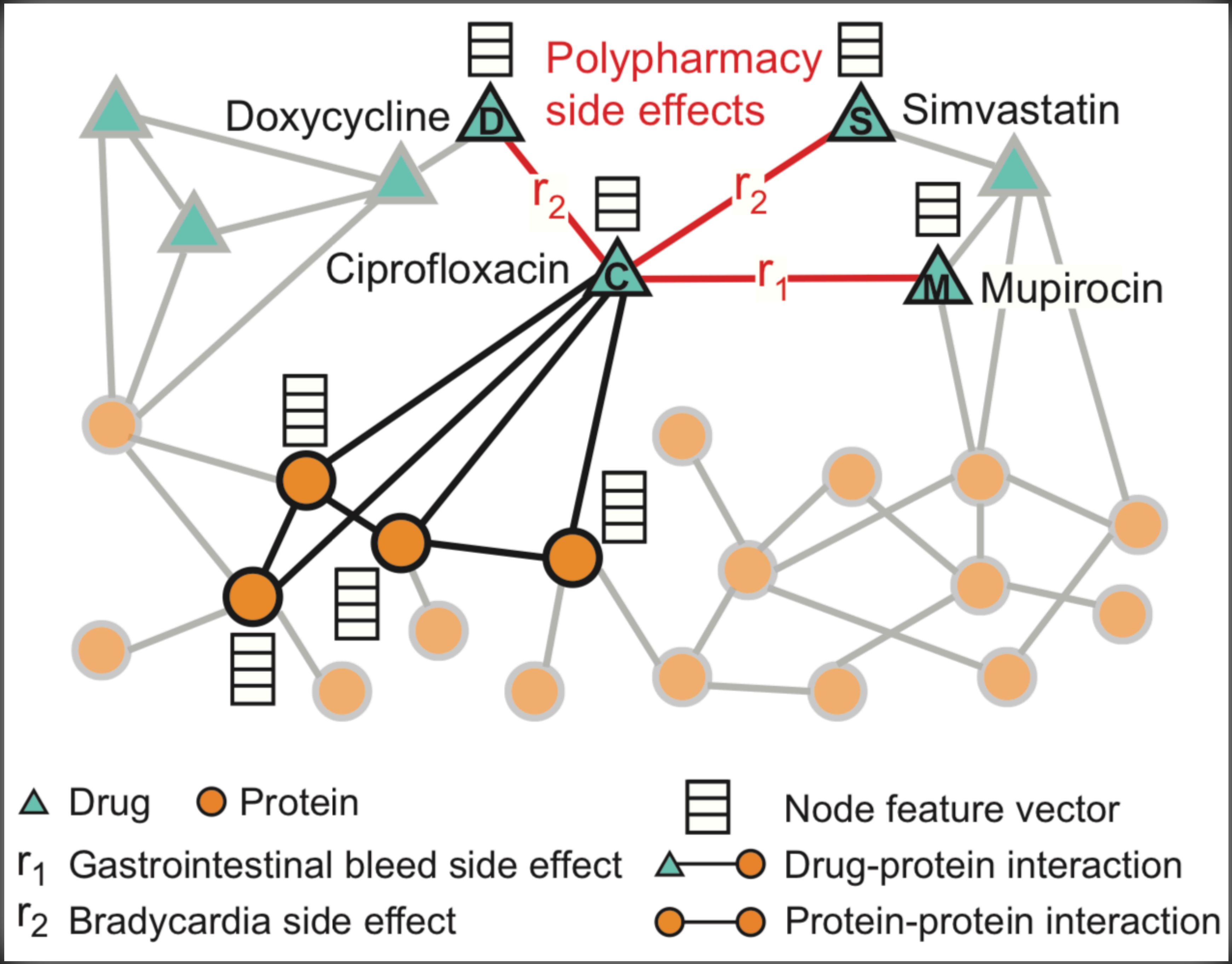
1) (Encoder) Graph Convolutional Network for multi-relational link prediction in multimodal networks
2) (Decoder) Tensor Factorization to reconstruct edges between drugs

NRI

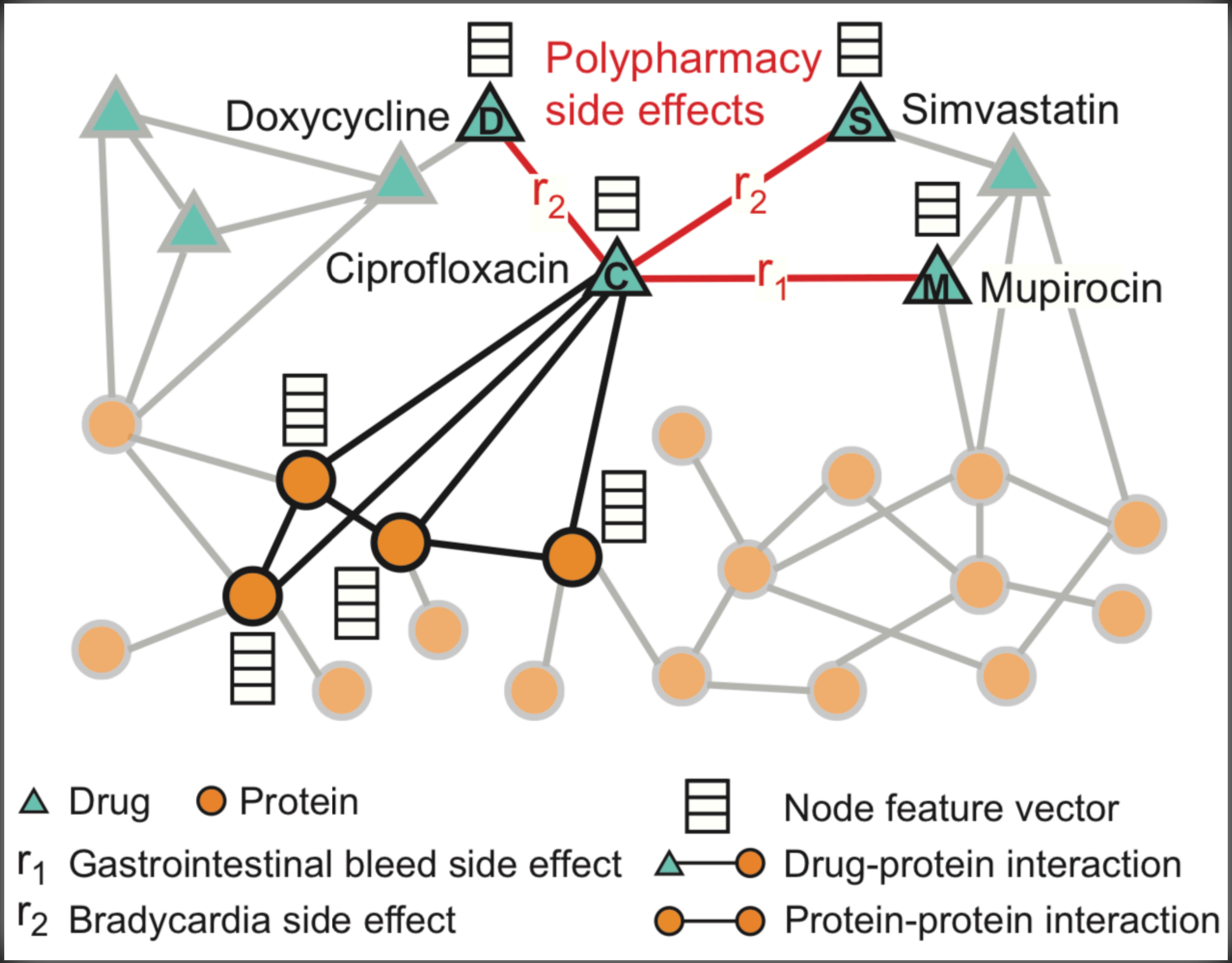
Decagon

Relational Reasoning

Implementation



https://arxiv.org/pdf/1802.00543.pdf



https://arxiv.org/pdf/1802.00543.pdf

NRI

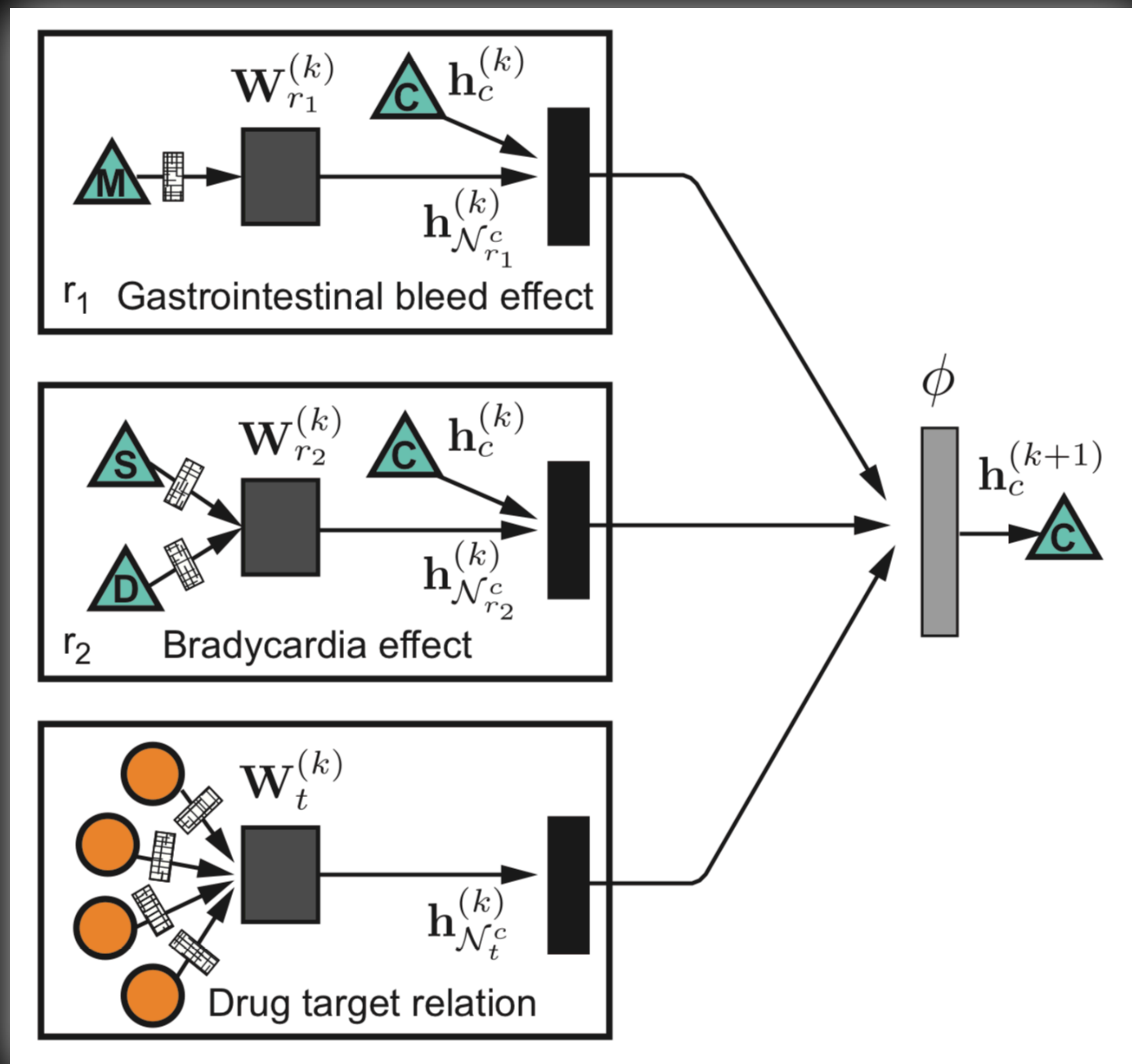
Decagon

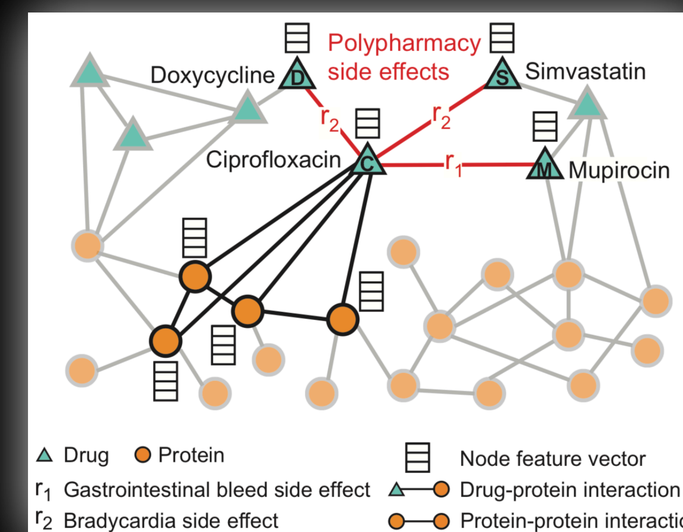
Relational Reasoning

Implementation

$$\mathbf{h}_i^{(k+1)} = \phi\left(\sum_r \sum_{j \in \mathcal{N}_r^i} c_r^{ij} \mathbf{W}_r^{(k)} \mathbf{h}_j^{(k)} + c_r^i \mathbf{h}_i^{(k)}\right)$$

$$\mathbf{h}_i^{(k+1)} = \phi\left(\sum_r \sum_{j \in \mathcal{N}_r^i} c_r^{ij} \mathbf{W}_r^{(k)} \mathbf{h}_j^{(k)} + c_r^i \mathbf{h}_i^{(k)}\right)$$





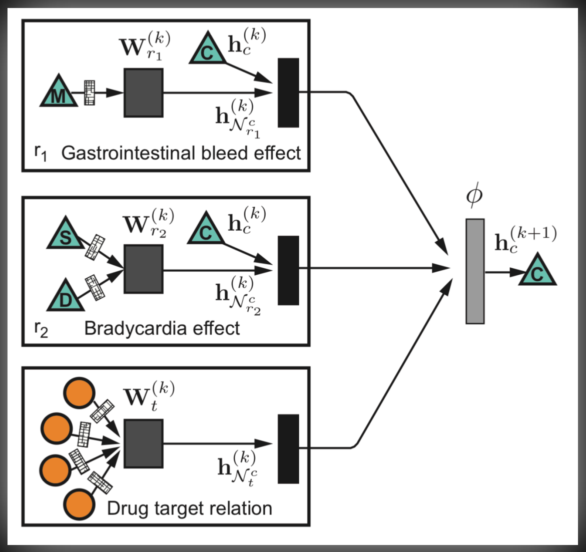
NRI

Decagon

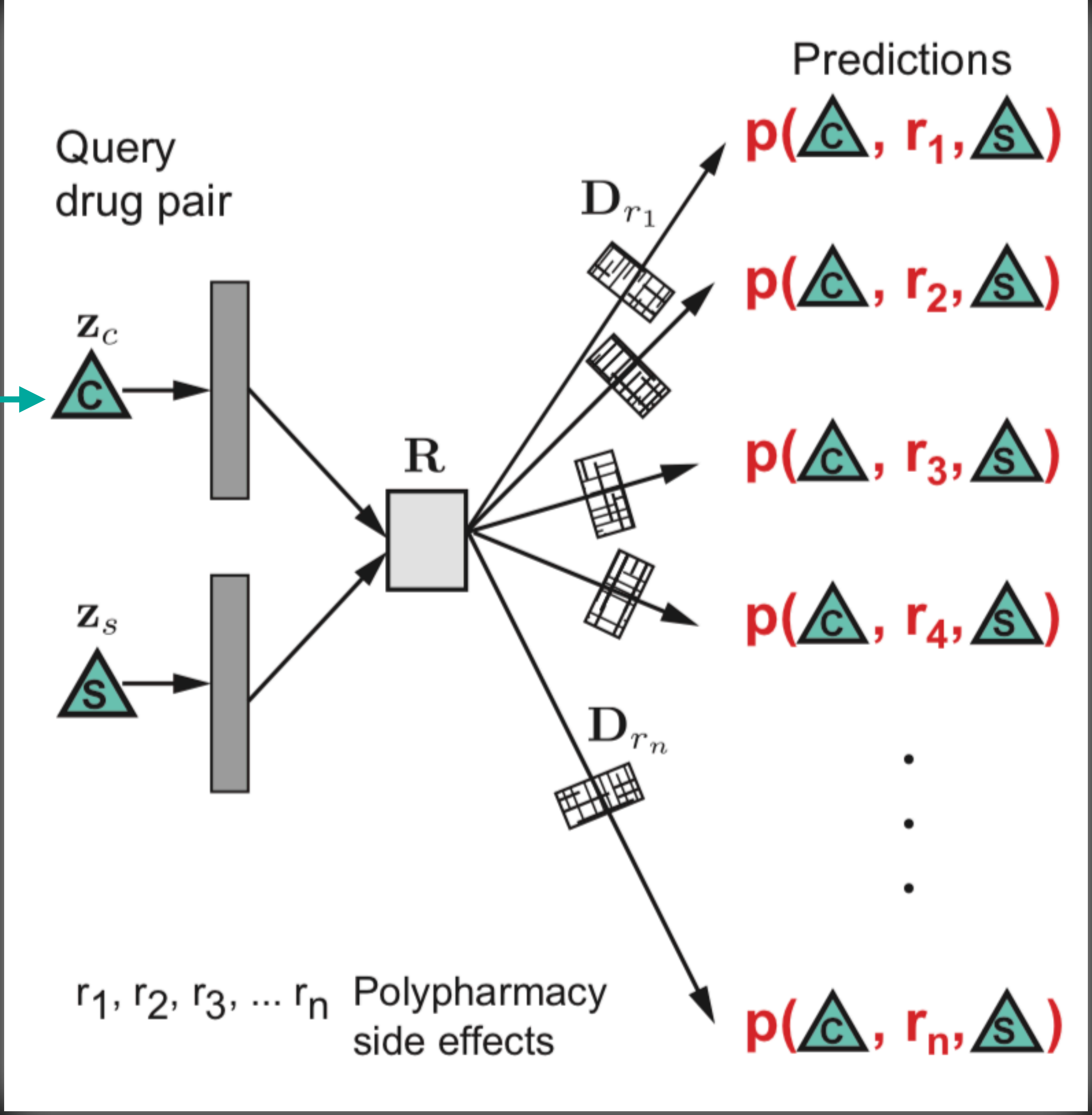
Relational Reasoning

Implementation

Encoder

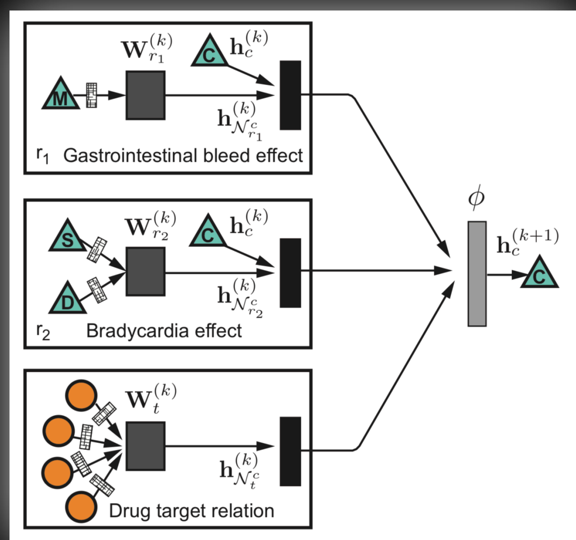


Decoder



NRI
Decagon
Relational Reasoning
Implementation

Encoder



Decoder

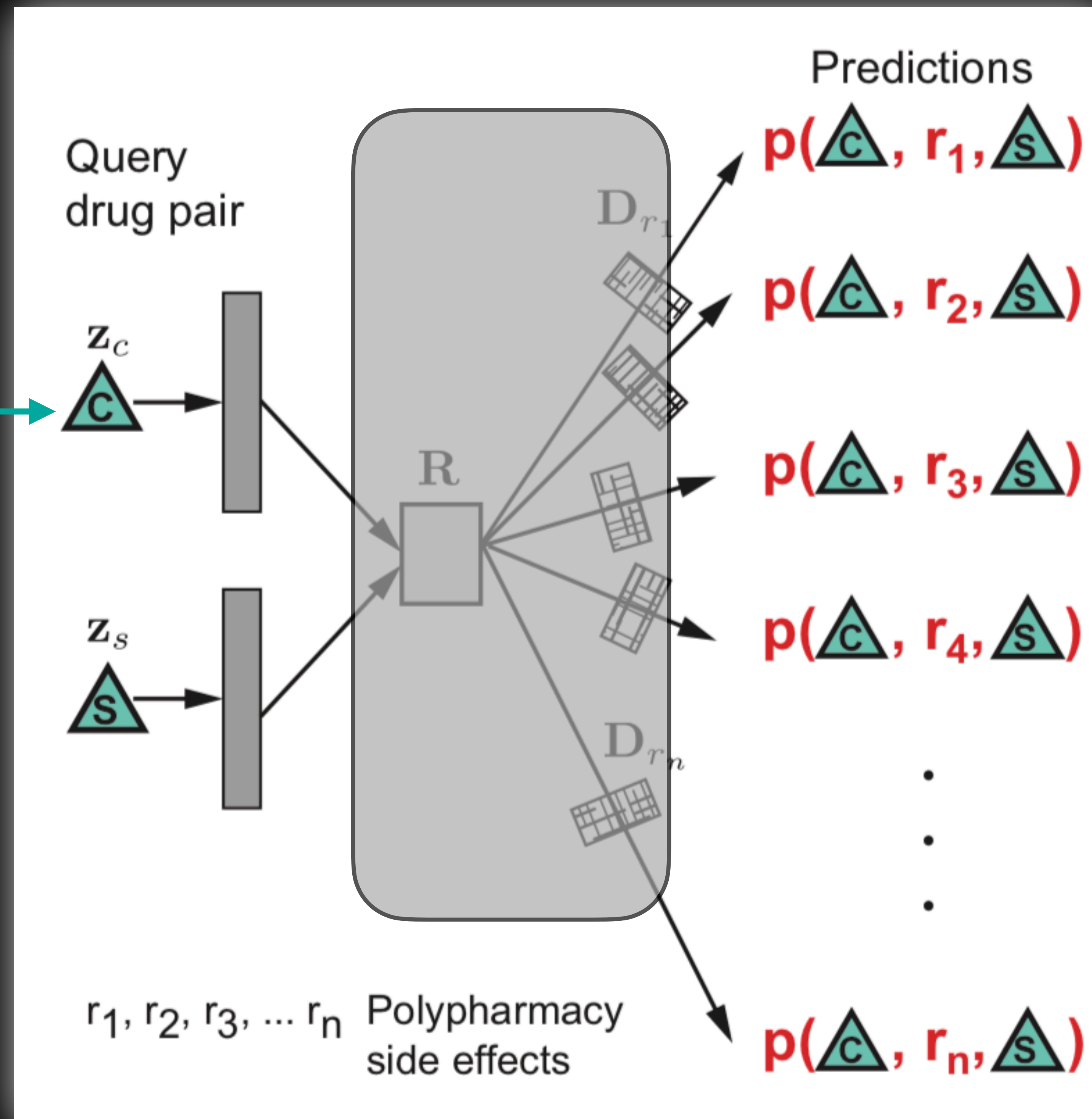


Table 2. Area under ROC curve (AUROC), area under precision-recall curve (AUPRC), and average precision at 50 (AP@50) for polypharmacy side effect prediction. Reported are average performance values for 964 side effect types.

Approach	AUROC	AUPRC	AP@50
Decagon	0.872	0.832	0.803
RESCAL tensor factorization	0.693	0.613	0.476
DEDICOM tensor factorization	0.705	0.637	0.567
DeepWalk neural embeddings	0.761	0.737	0.658
Concatenated drug features	0.793	0.764	0.712

Goal:

1) answer questions about objects in visual scenes

Goal:

1) answer questions about objects in visual scenes

Data:

1) curated images of entities of different sizes/types

NRI
Decagon
Relational Reasoning
Implementation

Goal:

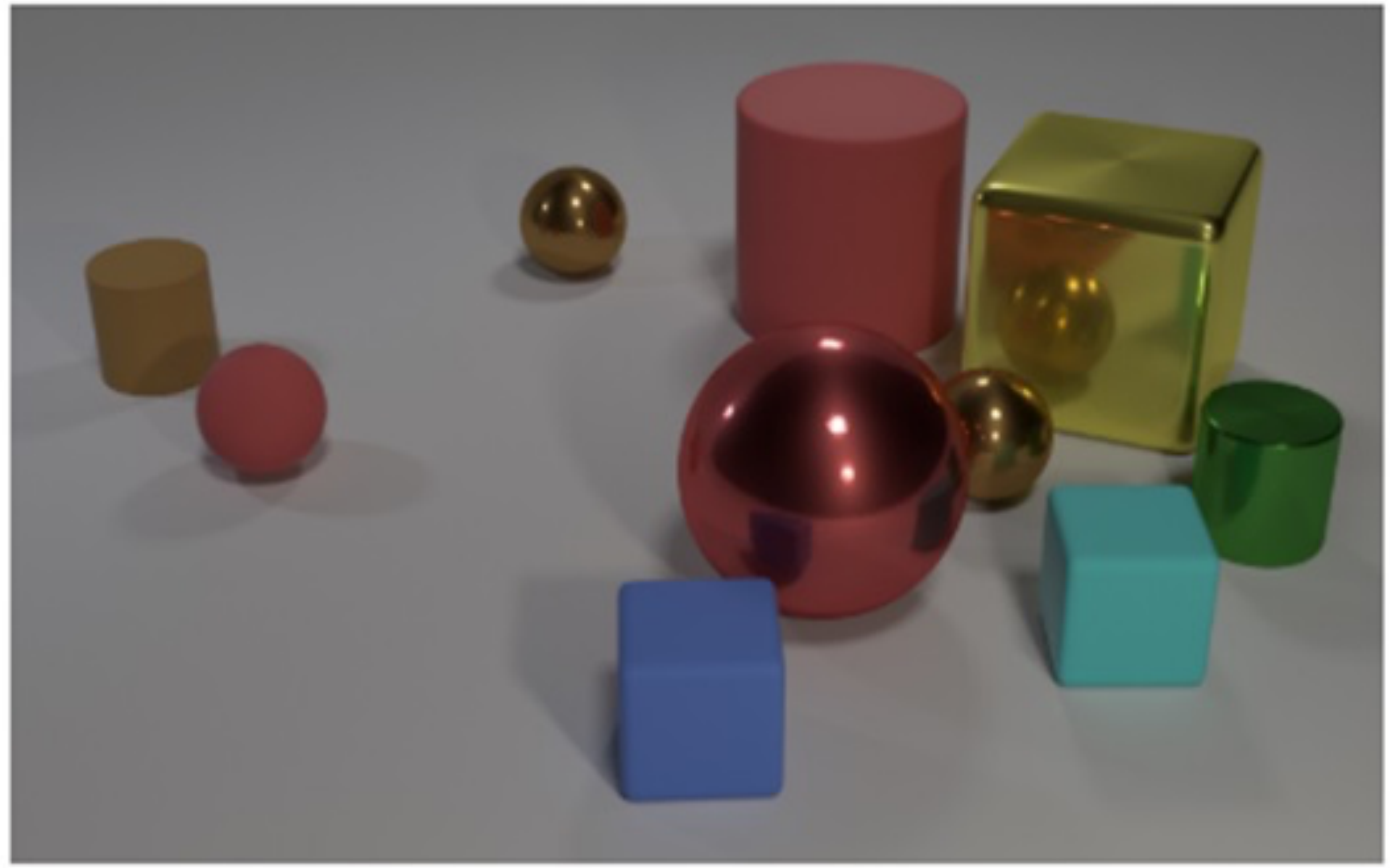
1) answer questions about objects in visual scenes

Data:

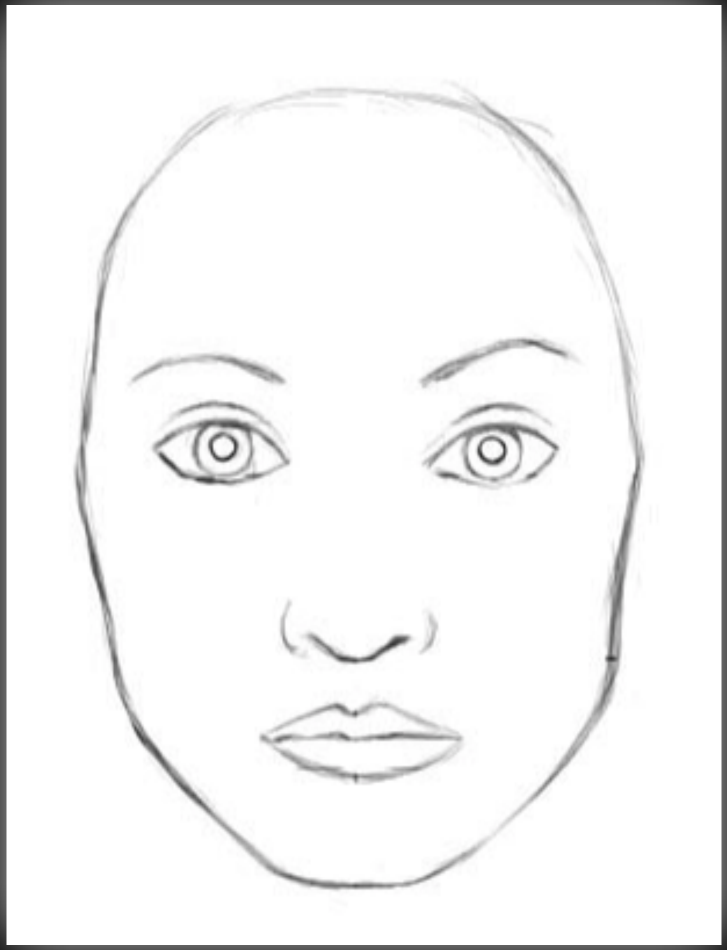
1) curated images of entities of different sizes/types

Model:

1) Varied - CNN + graph structured representations



Q: Are there an equal number of large things and metal spheres?
Q: What size is the cylinder that is left of the brown metal thing that is left of the big sphere?
Q: There is a sphere with the same size as the metal cube; is it made of the same material as the small red sphere?
Q: How many objects are either small cylinders or red things?

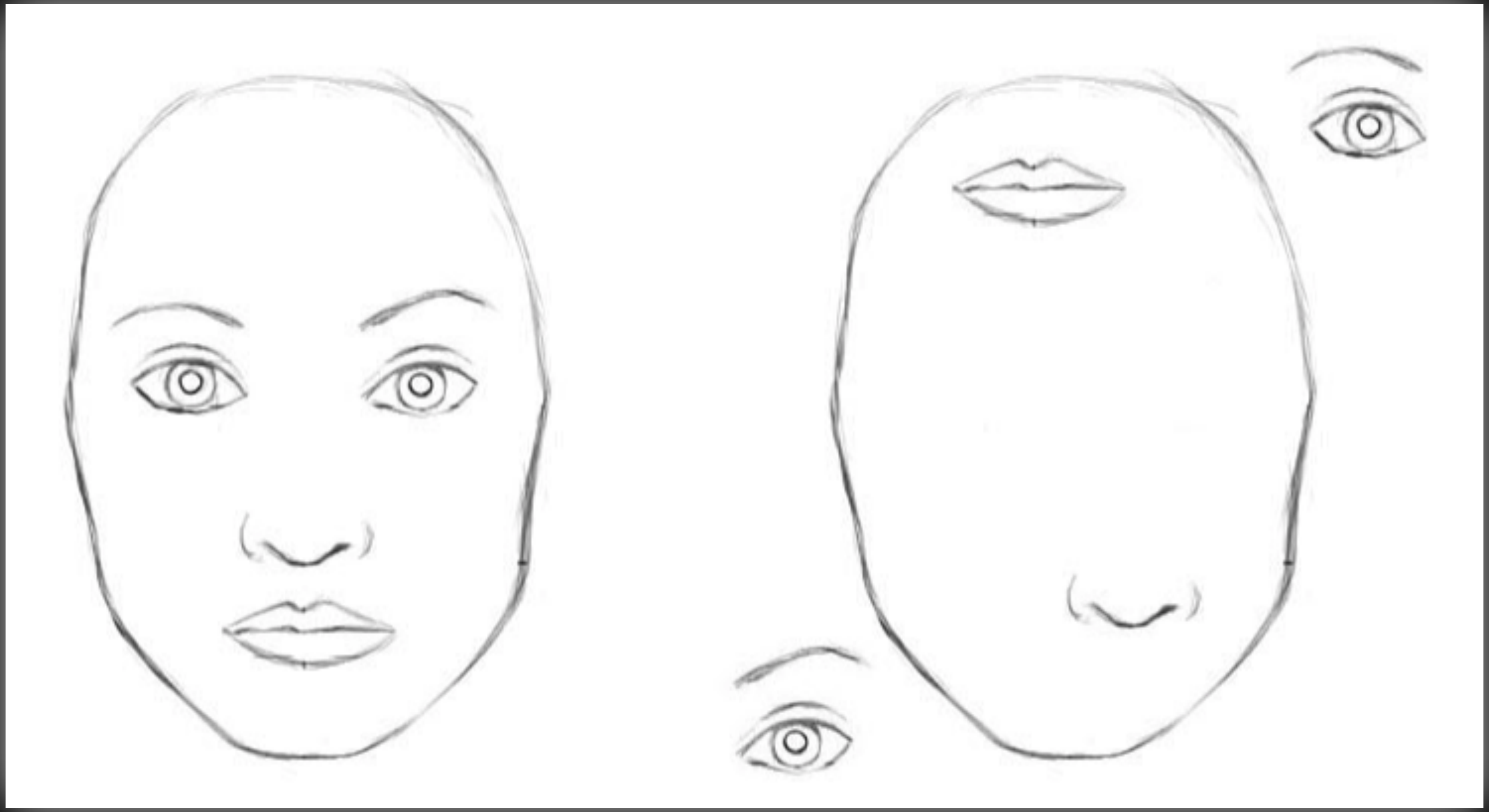


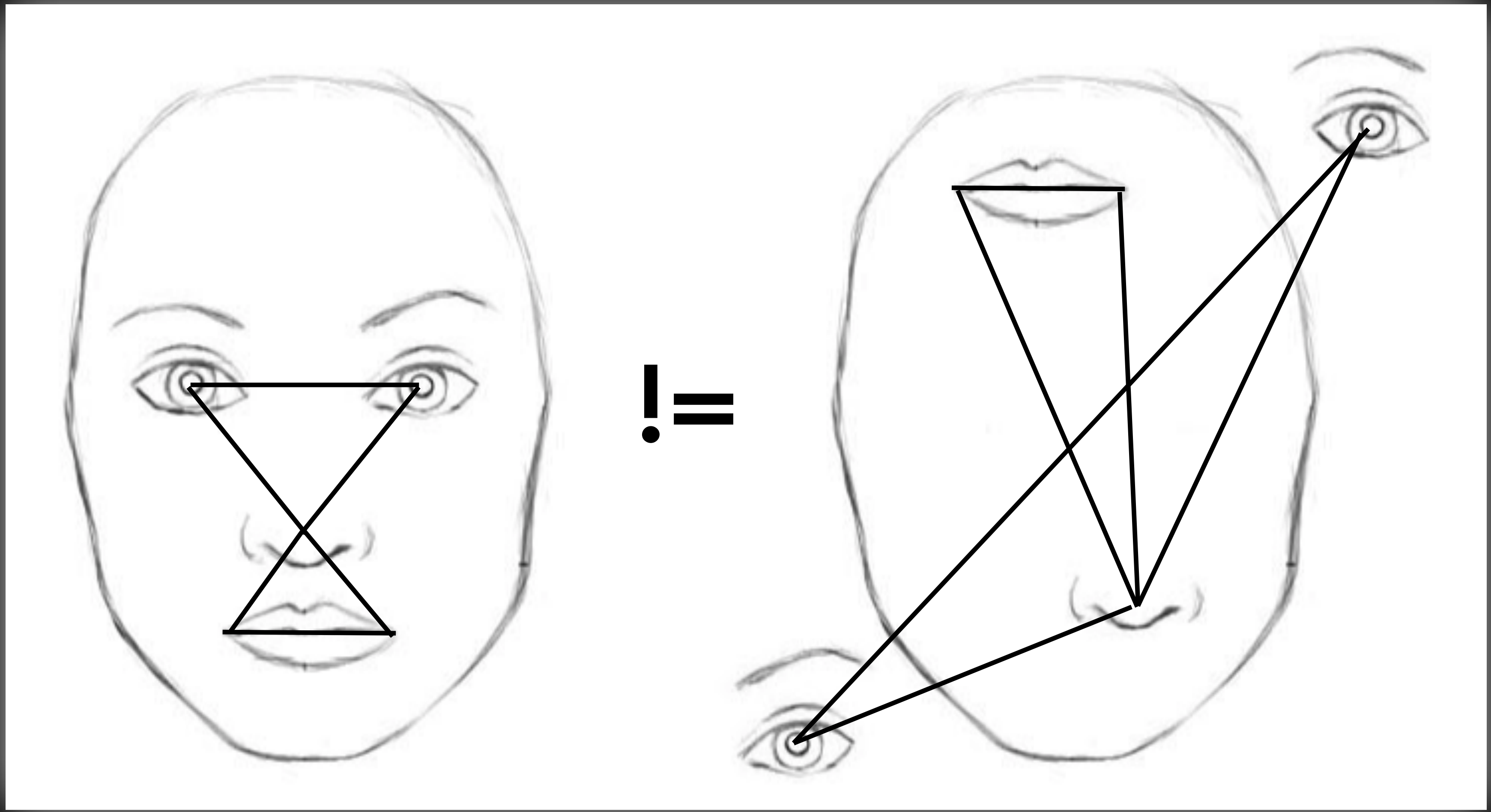
NRI

Decagon

Relational Reasoning

Implementation



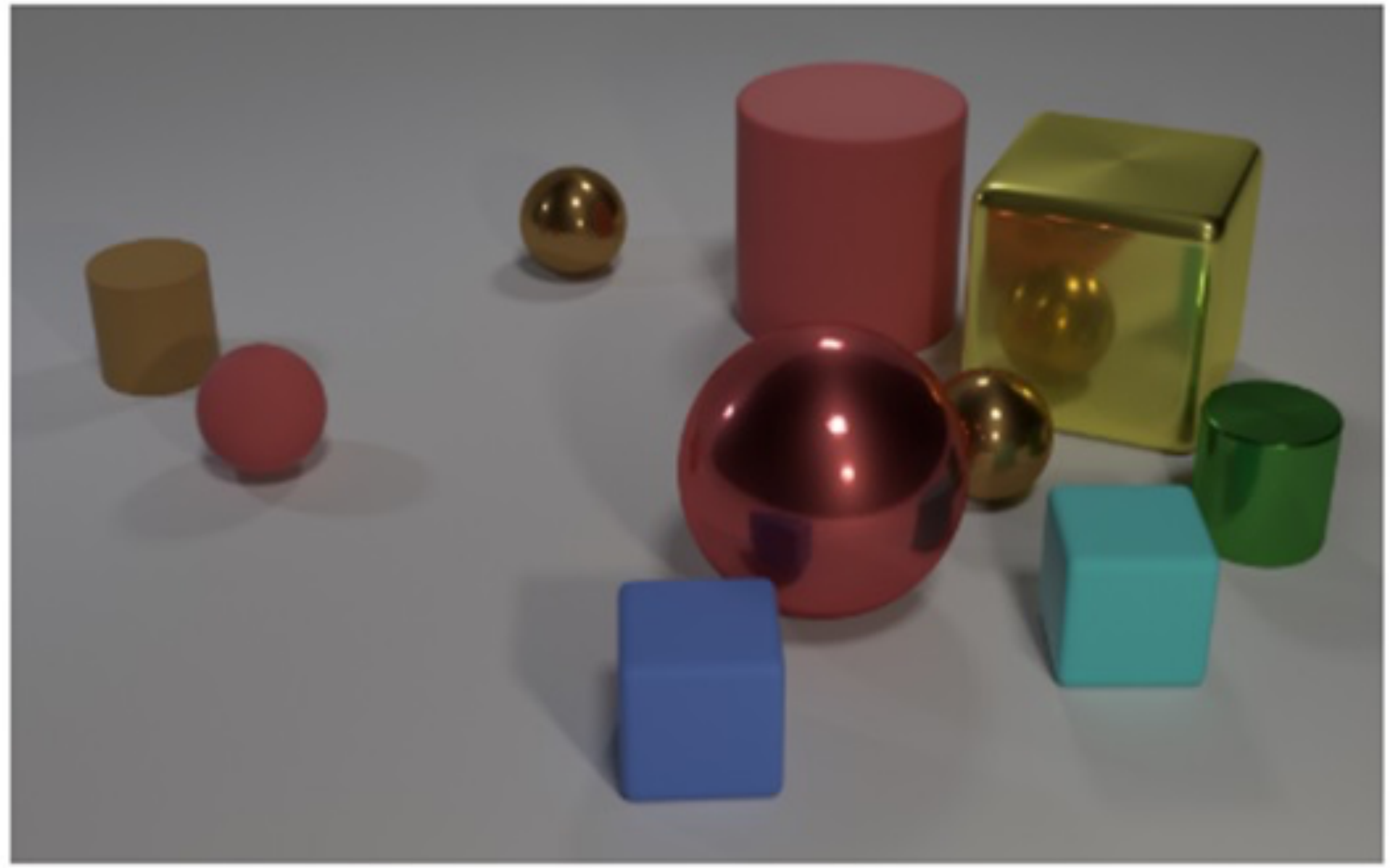


NRI

Decagon

Relational Reasoning

Implementation



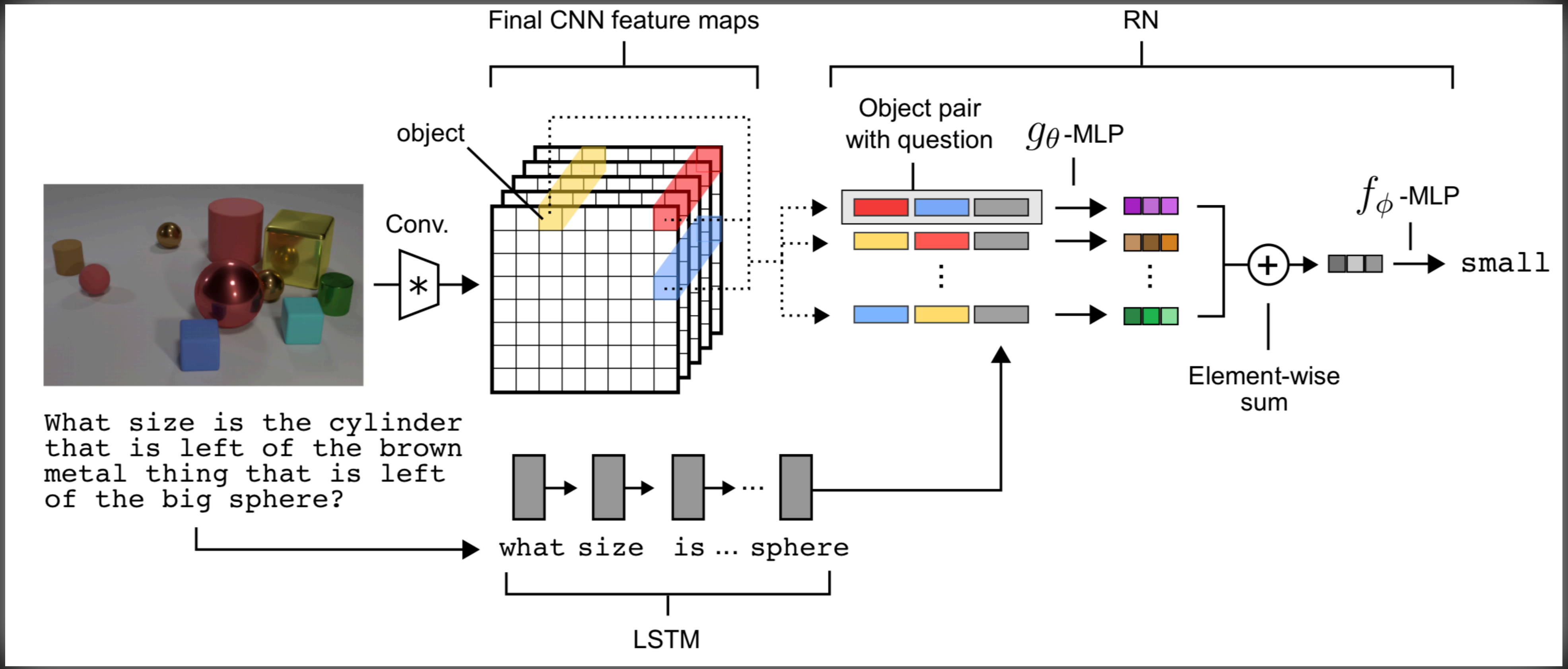
Q: Are there an **equal number** of **large things** and **metal spheres**?

Q: **What size** is the **cylinder that is left of** the **brown metal** thing **that is left of** the **big sphere**?

Q: There is a **sphere** with the **same size as** the **metal cube**; is it **made of the same material as** the **small red sphere**?

Q: **How many** objects are **either small cylinders** or **red** things?

NRI
Decagon
Relational Reasoning
Implementation



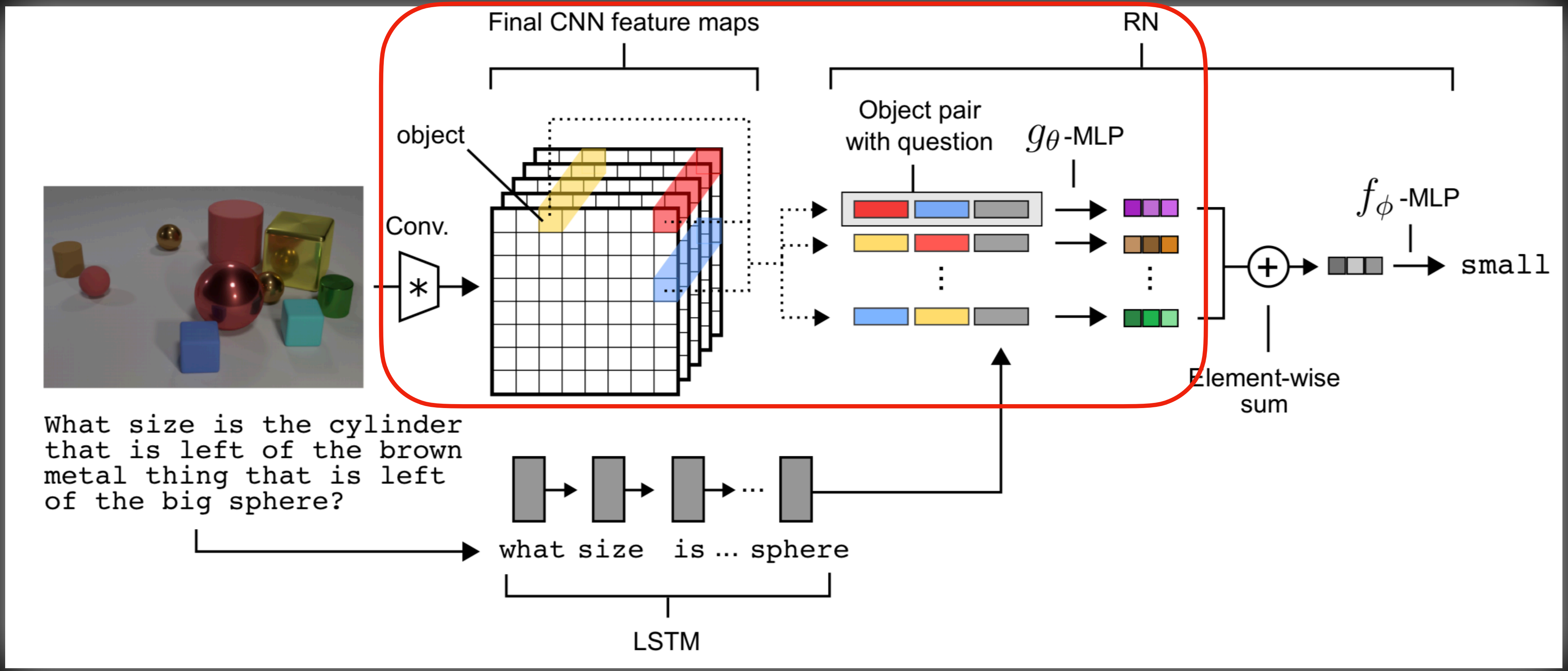
https://arxiv.org/pdf/1706.01427.pdf

NRI

Decagon

Relational Reasoning

Implementation



https://arxiv.org/pdf/1706.01427.pdf

NRI

Decagon

Relational Reasoning

Implementation

Model	**Overall**	Count	Exist	Compare Numbers	Query Attribute	Compare Attribute
Human	92.6	86.7	96.6	86.5	95.0	96.0
Q-type baseline	41.8	34.6	50.2	51.0	36.0	51.3
LSTM	46.8	41.7	61.1	69.8	36.8	51.8
CNN+LSTM	52.3	43.7	65.2	67.1	49.3	53.0
CNN+LSTM+SA	68.5	52.2	71.1	73.5	85.3	52.3
CNN+LSTM+SA*	76.6	64.4	82.7	77.4	82.6	75.4
CNN+LSTM+RN	**95.5**	**90.1**	**97.8**	**93.6**	**97.9**	**97.1**

NRI

Decagon

Relational Reasoning

Implementation

Model	**Overall**	Count	Exist	Compare Numbers	Query Attribute	Compare Attribute
Human	92.6	86.7	96.6	86.5	95.0	96.0
Q-type baseline	41.8	34.6	50.2	51.0	36.0	51.3
LSTM	46.8	41.7	61.1	69.8	36.8	51.8
CNN+LSTM	52.3	43.7	65.2	67.1	49.3	53.0
CNN+LSTM+SA	68.5	52.2	71.1	73.5	85.3	52.3
CNN+LSTM+SA*	76.6	64.4	82.7	77.4	82.6	75.4
CNN+LSTM+RN	**95.5**	**90.1**	**97.8**	**93.6**	**97.9**	**97.1**

NRI

Decagon

Relational Reasoning

Implementation

DeepMind GraphNets

https://github.com/deepmind/graph_nets

DeepMind GraphNets

https://github.com/deepmind/graph_nets

PyTorch Geometric

https://pytorch-geometric.readthedocs.io

DeepMind GraphNets

https://github.com/deepmind/graph_nets

PyTorch Geometric

https://pytorch-geometric.readthedocs.io

Open Graph Benchmark

http://ogb.stanford.edu

Open Graph Benchmark

http://ogb.stanford.edu

Open Graph Benchmark (OGB)

NRI

Decagon

Relational Reasoning

Implementation

Name	Size	Description
ogbn-proteins	100 K	Protein-protein association network linked across species
ogbn-wiki	1 M	Wikipedia hyperlinks
ogbn-products	2 M	Amazon co-purchasing network

Node Property Prediction

Name	Size	Description
ogbl-ddi	15 K	Drug-drug interaction network
ogbl-biomed	100 K	Human biomedical knowledge graph
ogbl-ppa	500 K	Protein-protein association network
ogbl-reviews	10 M	Amazon user-item review dataset
ogbl-citations	200 M	Microsoft Academic Graph citation network

Link Property Prediction

Name	Size	Description
ogbg-mol	500 K	Molecular property prediction datasets from MoleculeNet
ogbg-code	1 M	Abstract Syntax Trees of code snippets
ogbg-ppi	10 M	Protein-protein interaction network

Graph Property Prediction

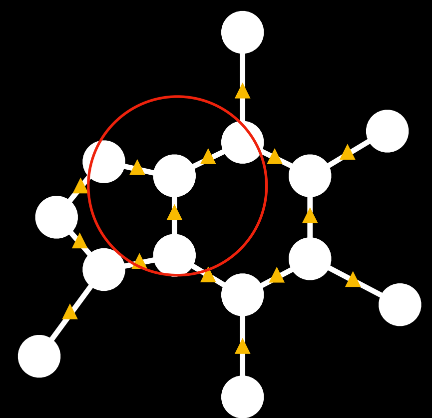
Depth	Cannot currently make "deep" GNNs
Scaling	Computational concerns
Generation	Converting sensory data into structured representations

Depth

Scaling

Generation

Problem 1: eventually we run out of nodes

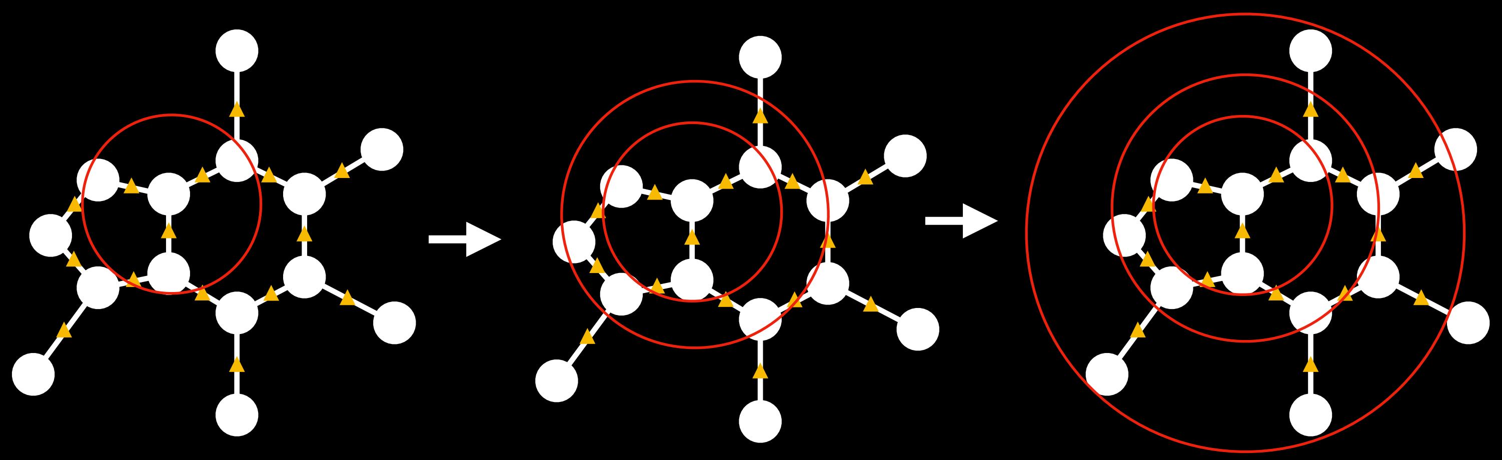


Depth

Scaling

Generation

Problem 1: eventually we run out of nodes

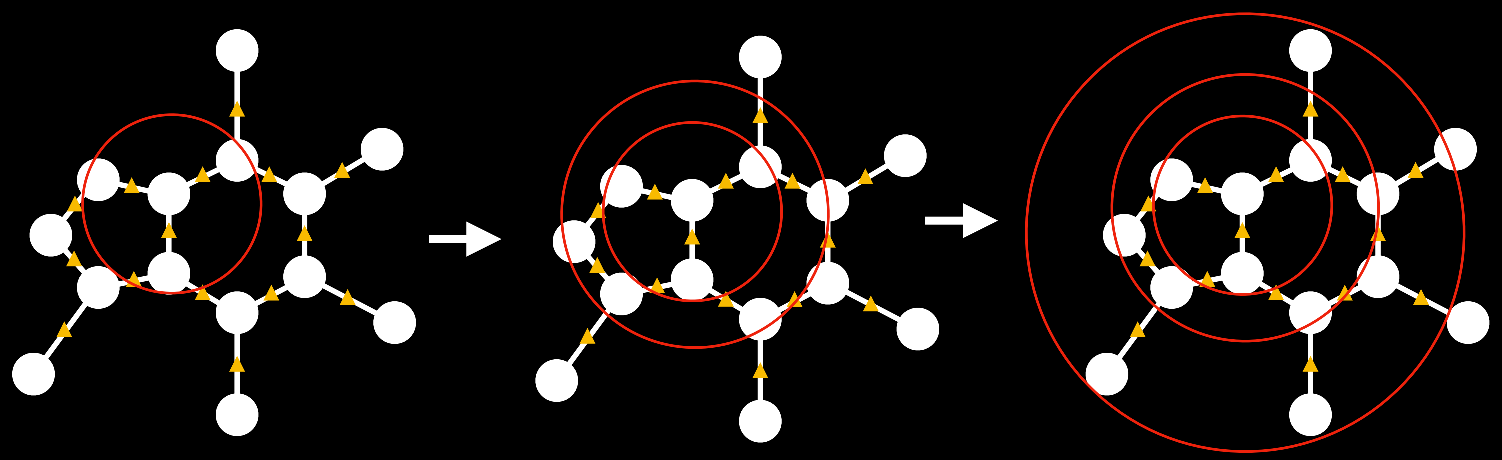


Depth

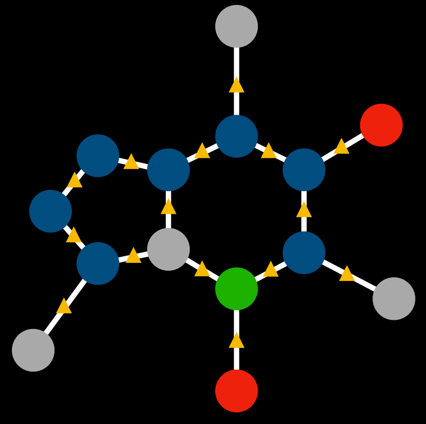
Scaling

Generation

Problem 1: eventually we run out of nodes



Problem 2: smoothing

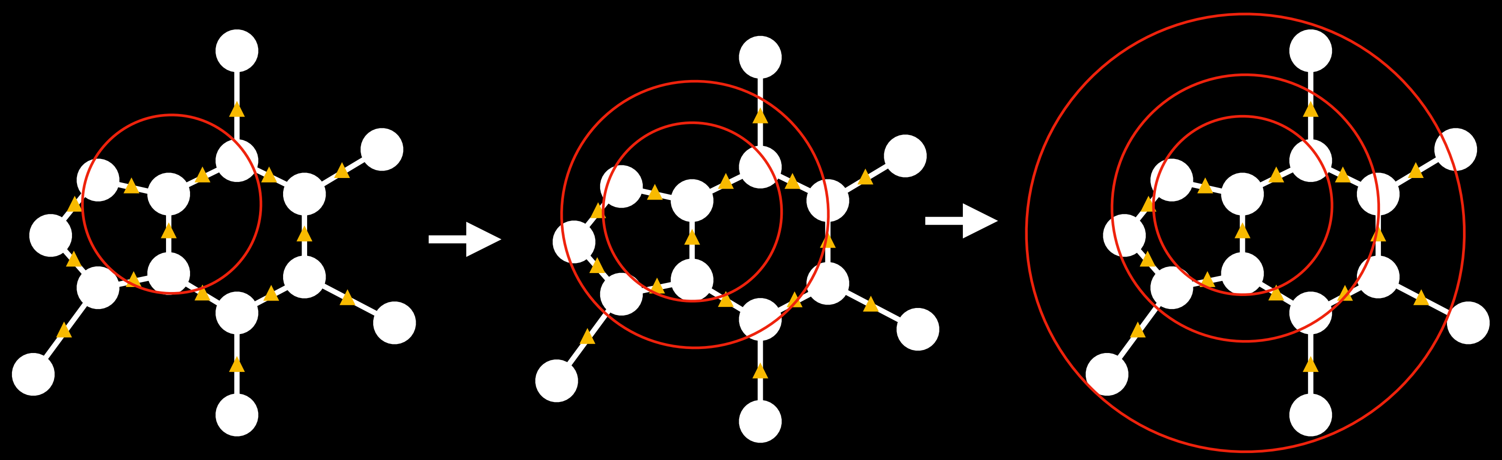


Depth

Scaling

Generation

Problem 1: eventually we run out of nodes



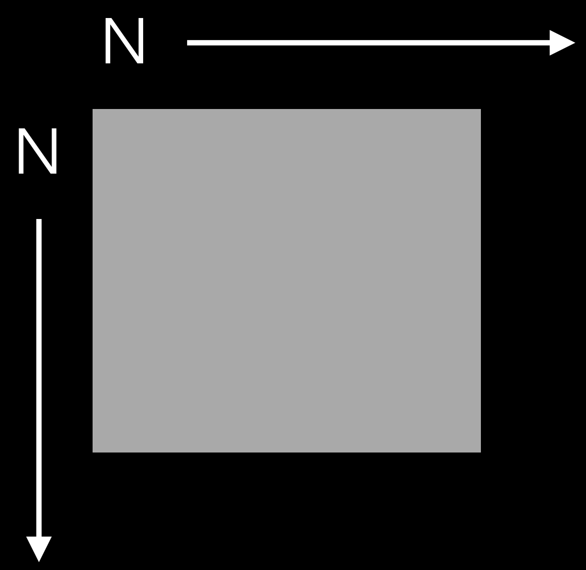
Problem 2: smoothing

Depth

Scaling

Generation

Problem 1: Current GCN formulation relies on adjacency matrix

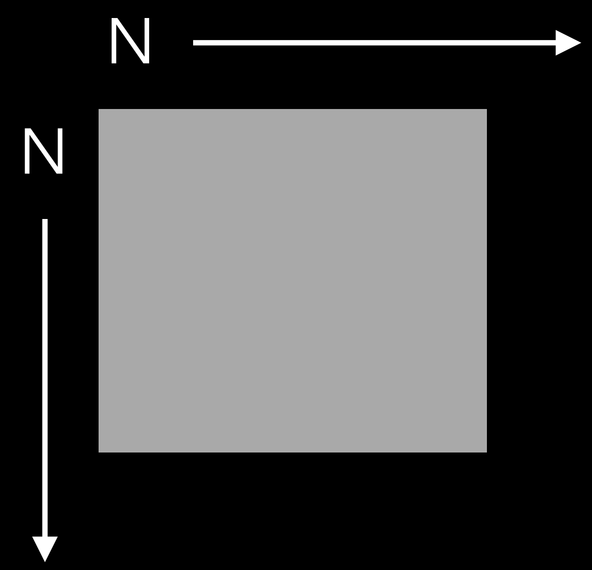


Depth

Scaling

Generation

Problem 1: Current GCN formulation relies on adjacency matrix



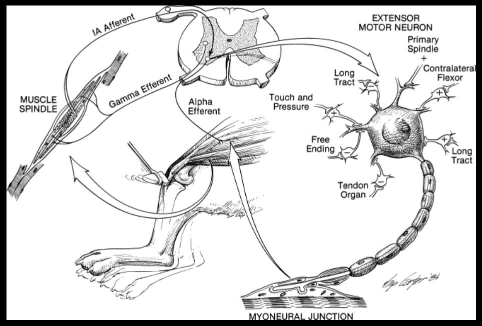
Problem 2: Models with a single network per node

Depth

Scaling

Generation

Problem 1: Huge amounts of sensory data, all "1-dimensional"

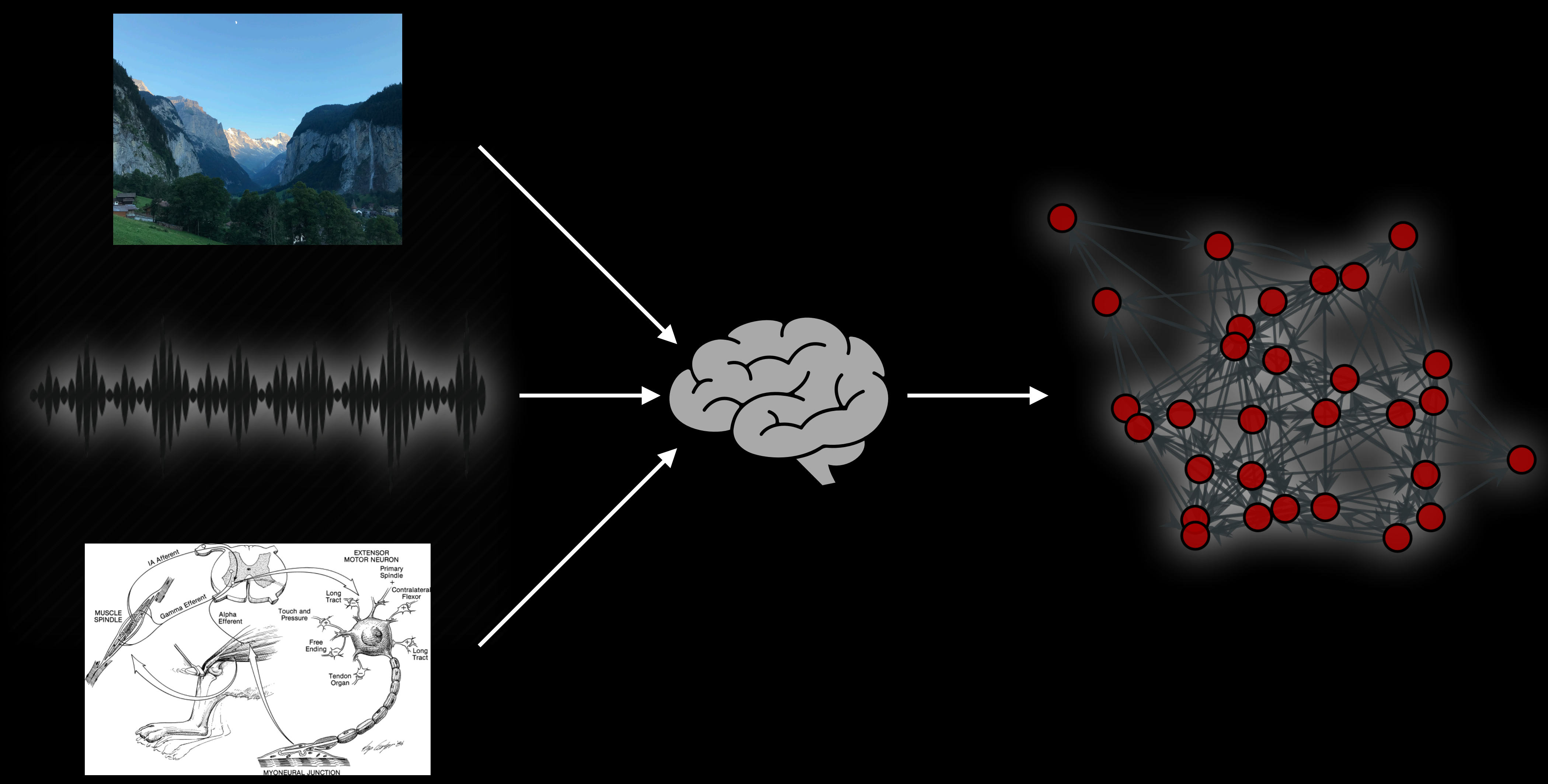


Depth

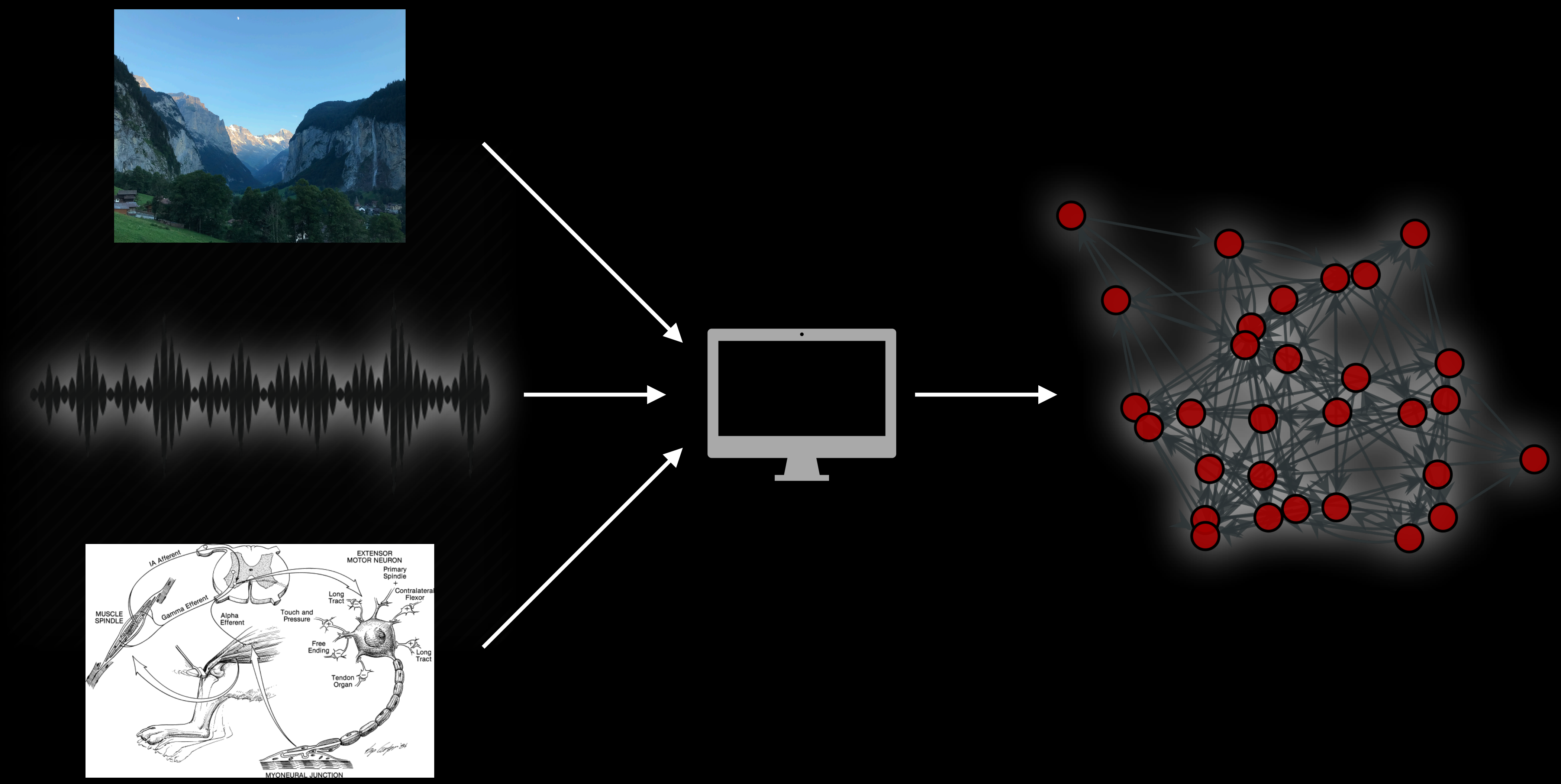
Scaling

Generation

Problem 1: Huge amounts of sensory data, all "1-dimensional"



Problem 1: Huge amounts of sensory data, all “1-dimensional”



1. Know what to use to implement a Graph Neural Network

1. Know what to use to implement a Graph Neural Network

2. Intuition for the kinds of problems in which GNNs will provide an advantage

1. Know what to use to implement a Graph Neural Network

2. Intuition for the kinds of problems in which GNNs will provide an advantage

3. Understand why structure is crucial in determining the behavior of interacting systems

1. Know what to use to implement a Graph Neural Network

2. Intuition for the kinds of problems in which GNNs will provide an advantage

3. Understand why structure is crucial in determining the behavior of interacting systems

4. Understand why **relational inductive biases** are critical for learning about interacting systems

Resources

- Battaglia, P. W., Hamrick, J. B., Bapst, V., Sanchez-Gonzalez, A., Zambaldi, V., Malinowski, M., … & Gulcehre, C. (2018). **Relational inductive biases, deep learning, and graph networks.** arXiv preprint arXiv: 1806.01261.

- Hamilton, W. L., Ying, R., & Leskovec, J. (2017). **Representation learning on graphs: Methods and applications.** IEEE Data Engineering Bulletin.

- Bronstein, M. M., Bruna, J., LeCun, Y., Szlam, A., & Vandergheynst, P. (2017). **Geometric deep learning: going beyond euclidean data.** IEEE Signal Processing Magazine, 34(4), 18-42.

- Goyal, P., & Ferrara, E. (2018). **Graph embedding techniques, applications, and performance: A survey**. Knowledge-Based Systems, 151, 78-94.

- Non-comprehensive but substantial list of geometric DL papers: https://github.com/thunlp/GNNPapers

- Graph Representation Learning @NeurIPS: https://grlearning.github.io/papers/