CSC2457 3D & Geometric Deep Learning

Relational inductive biases, deep learning, and graph networks

Peter W. Battaglia, Jessica B. Hamrick, Victor Bapst, Alvaro Sanchez-Gonzalez, Vinicius Zambaldi, Mateusz Malinowski,
Andrea Tacchetti, David Raposo, Adam Santoro1 Ryan Faulkner, Caglar Gulcehre, Francis Song, Andrew Ballard, Justin Gilmer,
George Dahl, Ashish Vaswani, Kelsey Allen, Charles Nash, Victoria Langston, Chris Dyer, Nicolas Heess,
Daan Wierstra, Pushmeet Kohli, Matt Botvinick, Oriol Vinyals, Yujia Li, Razvan Pascanu

Date: March 30th, 2021

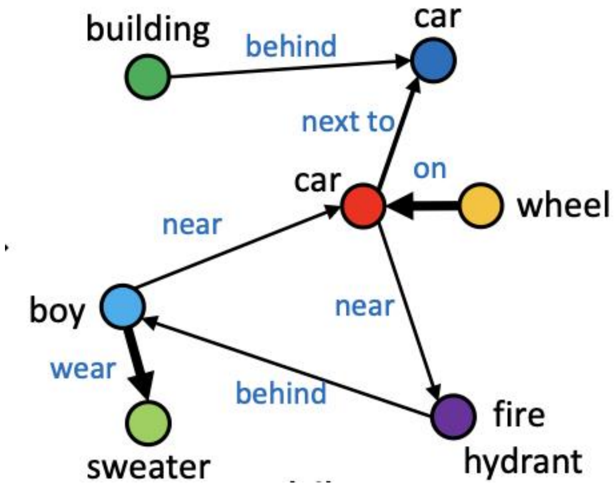
Presenter: Seung Wook Kim

Instructor: Animesh Garg

UNIVERSITY OF TORONTO

Motivation

- A key signature of human intelligence – “infinite use of finite means” (Humboldt 1836, Chomsky 1965) or combinatorial generalization



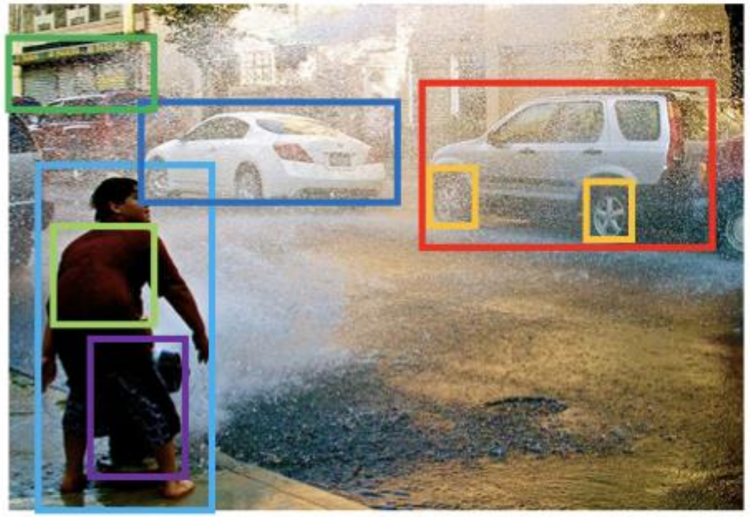


Image credit: Graph R-CNN for Scene Graph Generation (Yang et al.)

Motivation

- A key signature of human intelligence – “infinite use of finite means” (Humboldt 1836, Chomsky 1965) or combinatorial generalization

Example)

Language:

Dog bites man

Man bites dog

Motivation

- Humans solve novel problems by composing familiar skills and routines

- Humans draw analogies by aligning the relational structure between two domains

- Essentially, we understand the world in compositional terms.

Motivation

- In pre-deep learning era, machine learning community focused on structural reasoning
  - Graphical models, causal reasoning, symbolic logic
- Need structural assumptions or inductive biases to build those models
  - Wrong assumptions lead to bad models

Motivation

- Deep learning or neural network models do not need such strong structural assumptions, but had not been successful because
  - Not enough data, not enough compute

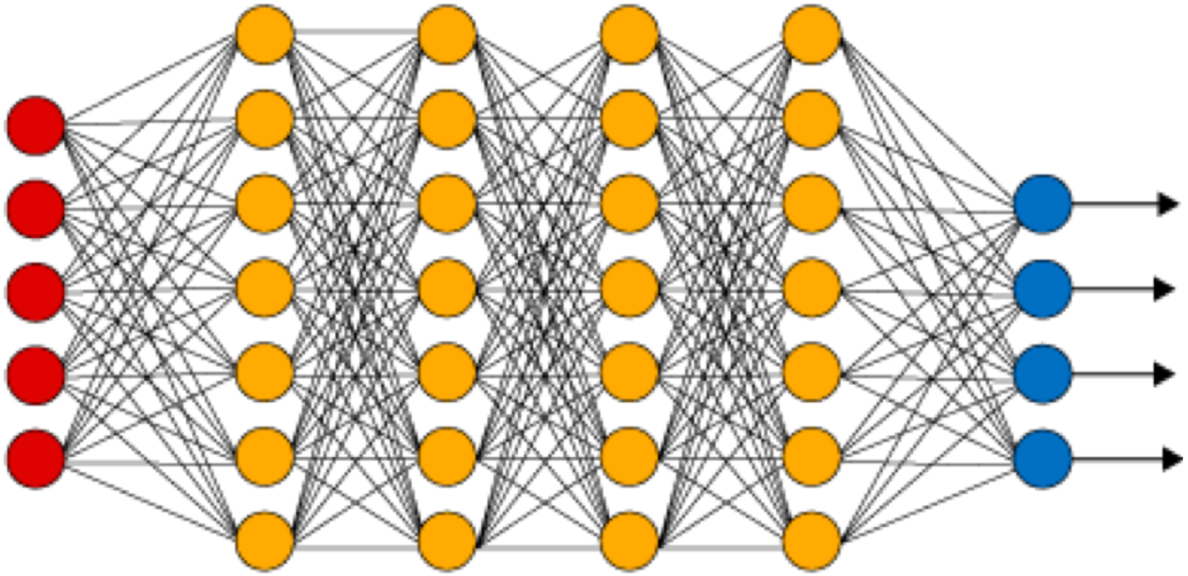


Image credit: https://niessner.github.io/I2DL/

Motivation

- With gigantic datasets and advancement in computing resources (e.g. GPU), deep learning models are thriving

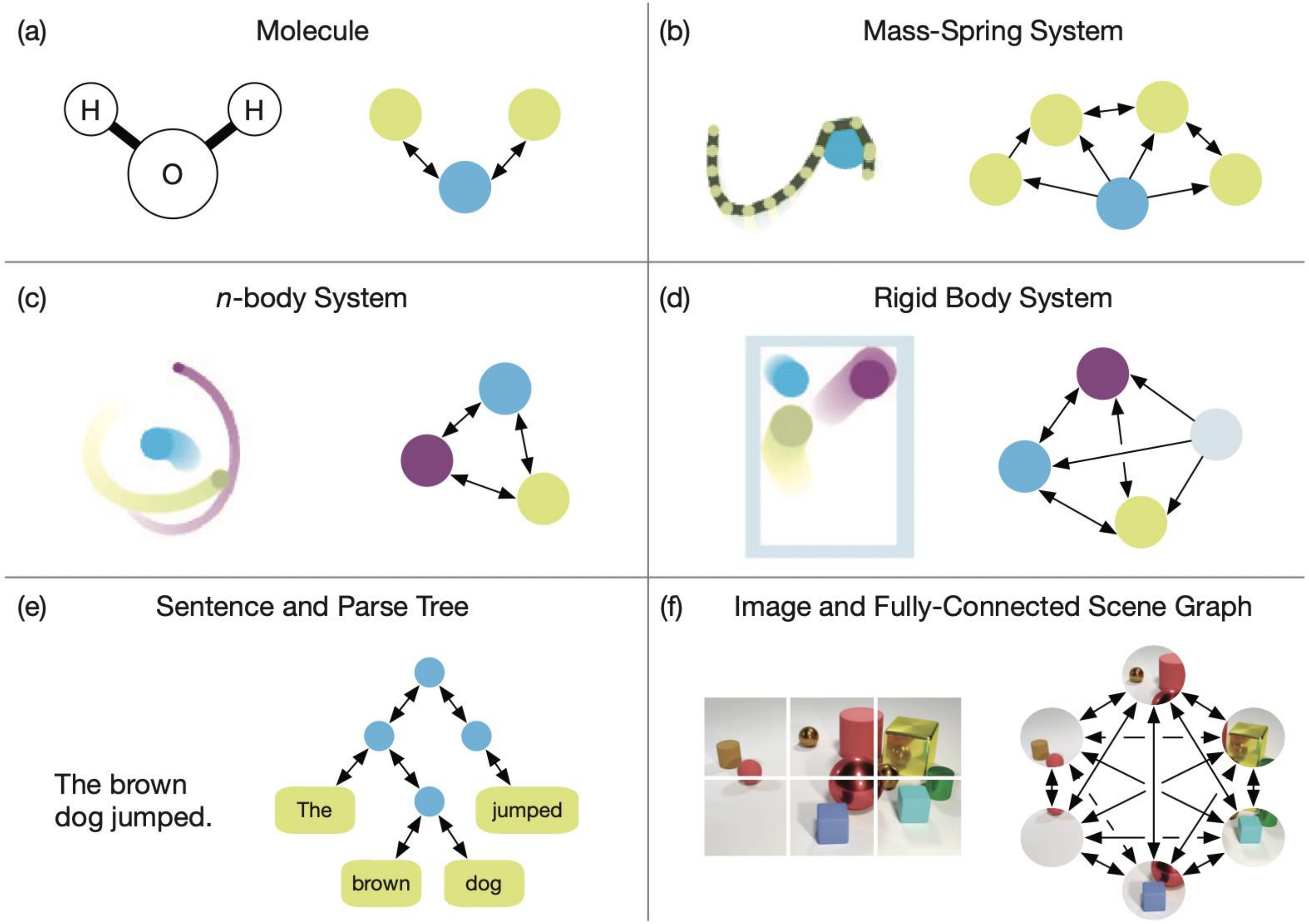
Image credit: Analyzing and Improving the Image Quality of StyleGAN (Karras et al. 2019)

Motivation

- Let's add structural assumptions to neural network models!

- In this paper, we focus on graphs + neural networks
  - Perform differentiable computations over vertices and edges
  - The representation and relations between vertices ***can be learned, not pre-defined***
  - Arbitrary pairwise relational structure
- A key signature of human intelligence – "infinite use of finite means"
  - Combination of concepts and relationship between them can be naturally represented with graphs

What can we represent with graphs?



Contributions

- This is a position paper that argues “combinatorial generalization” must be a top priority for AI to achieve human-level intelligence

- Analyzes different kinds of inductive biases in neural network models

- Proposes a general formulation of Graph Networks

Definitions

- Entity (Vertices): input data / objects / their representations

- Relations (Edges): specifies how entities are related

In Neural Nets:

Component	Entities	Relations
Fully connected	Units	All-to-all
Convolutional	Grid elements	Local
Recurrent	Timesteps	Sequential
Graph network	Nodes	Edges

Inductive Bias

- Combination of concepts and relationship between them can be naturally represented with graphs -> strong relational inductive bias

- Inductive bias allows a learning algorithm to prioritize one solution over another, independent of the observed data (Mitchell, 1980)

- E.g. Bayesian models

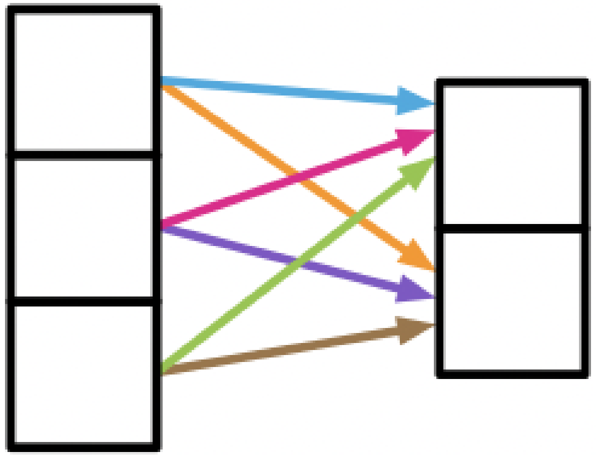
$$p(\theta|X,Y) = \frac{p(Y|X,\theta)p(\theta)}{p(Y|X)}$$

Relational Inductive Bias

- Inductive biases that impose constraints on relationships and interactions among entities in a learning process

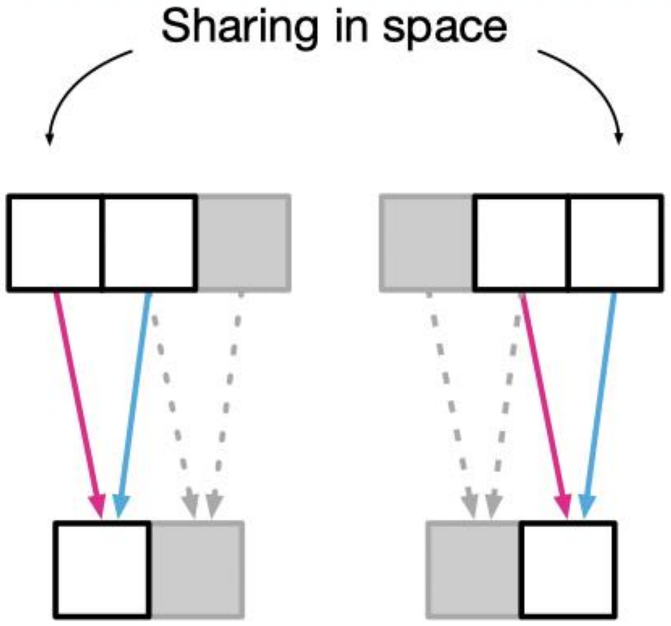
Rel. Inductive Bias in Neural Networks

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



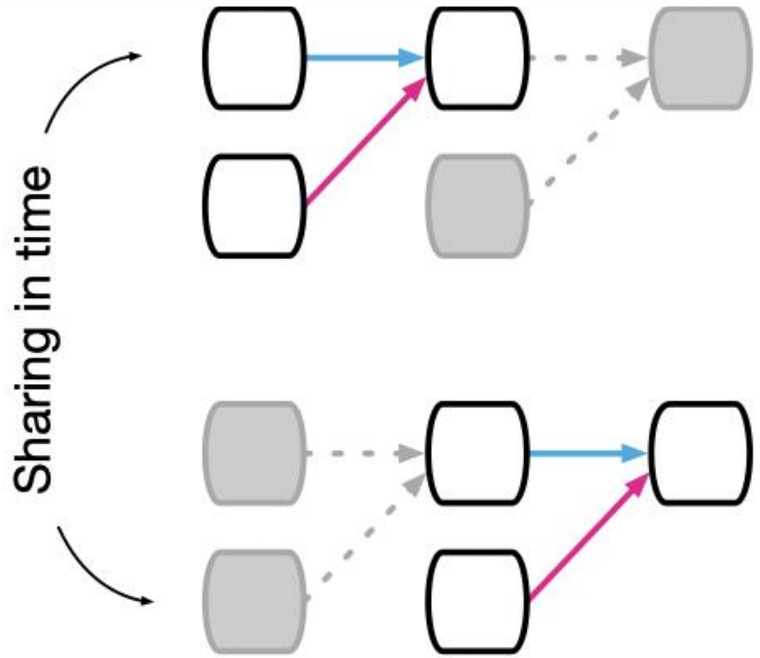
Rel. Inductive Bias in Neural Networks

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



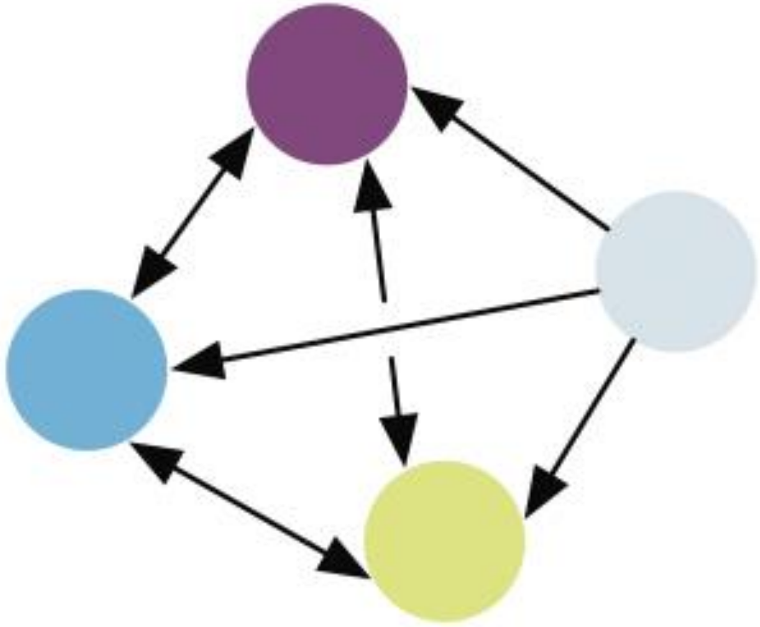
Rel. Inductive Bias in Neural Networks

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



Rel. Inductive Bias in Neural Networks

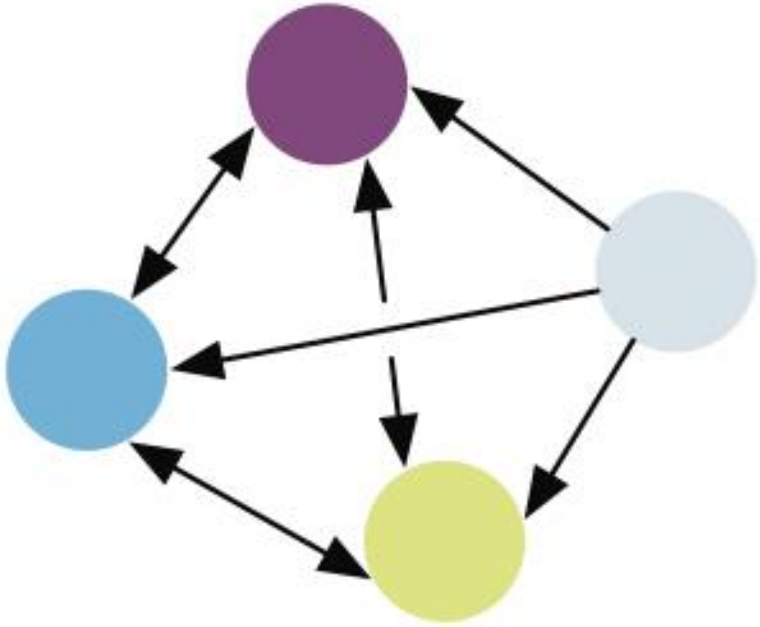
Component	Entities	Relations	Rel. inductive bias	Invariance
Fully connected	Units	All-to-all	Weak	-
Convolutional	Grid elements	Local	Locality	Spatial translation
Recurrent	Timesteps	Sequential	Sequentiality	Time translation
Graph network	Nodes	Edges	Arbitrary	Node, edge permutations



- Strong relational inductive bias beyond what other layers can provide
- Operates on arbitrary relational structure

Rel. Inductive Bias in Neural Networks

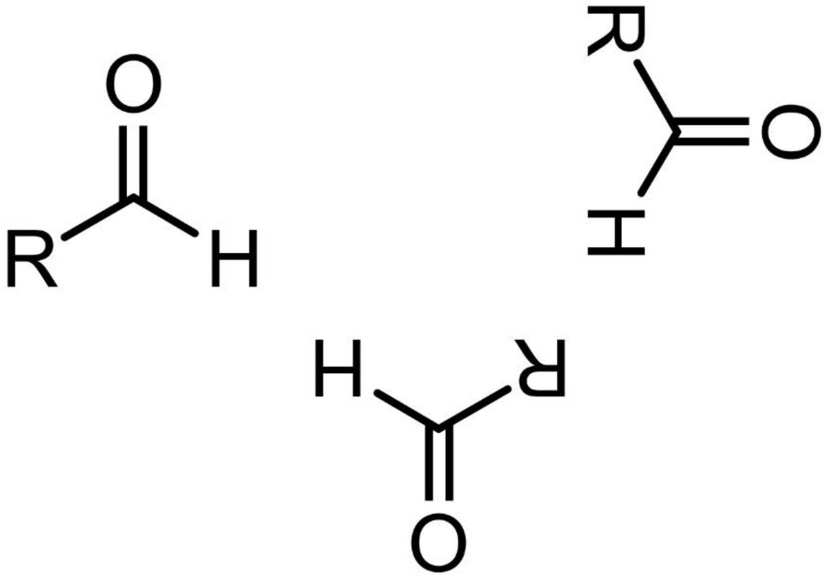
Component	Entities	Relations	Rel. inductive bias	Invariance
Fully connected	Units	All-to-all	Weak	-
Convolutional	Grid elements	Local	Locality	Spatial translation
Recurrent	Timesteps	Sequential	Sequentiality	Time translation
Graph network	Nodes	Edges	Arbitrary	Node, edge permutations



- Invariant to order of nodes
- Shared computations across all node/edges
  - > combinatorial generalization

Rel. Inductive Bias in Neural Networks

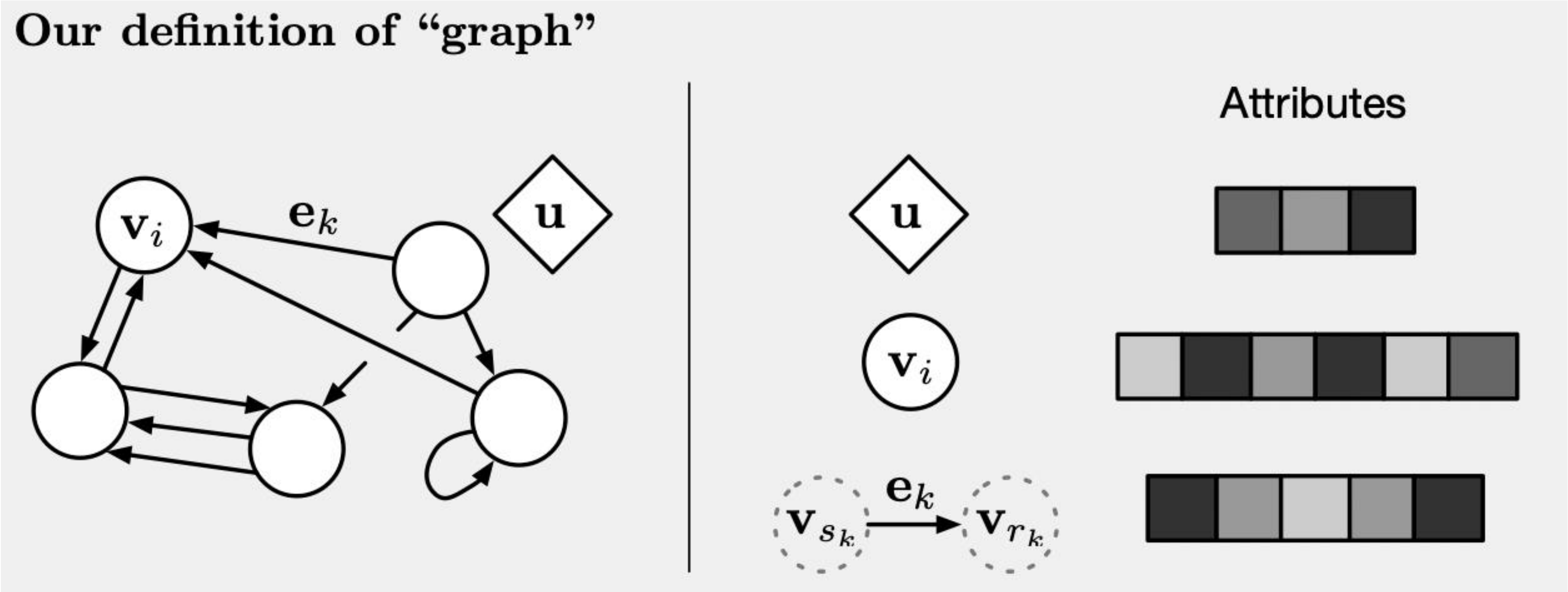
Component	Entities	Relations	Rel. inductive bias	Invariance
Fully connected	Units	All-to-all	Weak	-
Convolutional	Grid elements	Local	Locality	Spatial translation
Recurrent	Timesteps	Sequential	Sequentiality	Time translation
Graph network	Nodes	Edges	Arbitrary	Node, edge permutations



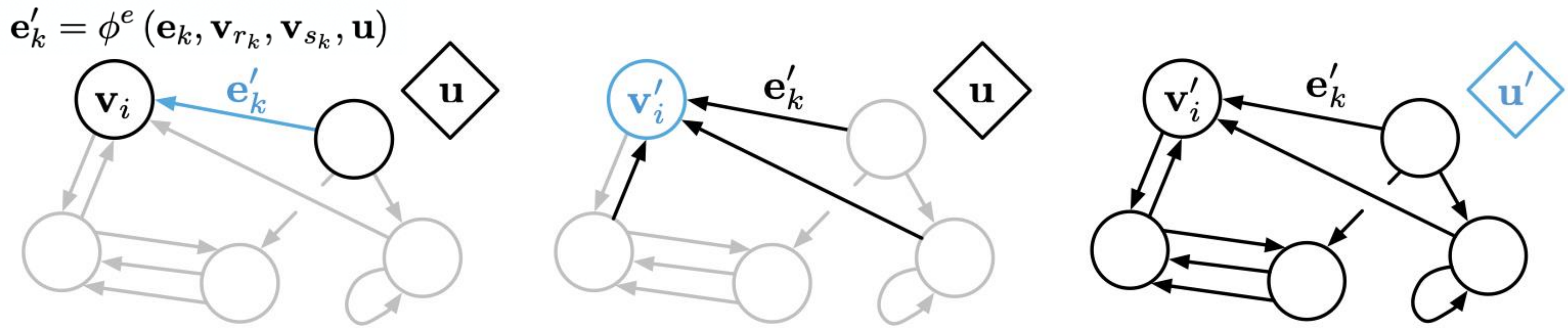
- Invariant to order of nodes
- Shared computations across all node/edges
  - > combinatorial generalization

GraphNetwork (GN) framework

$$G = (\mathbf{u}, V, E) \quad V = \{\mathbf{v}_i\}_{i=1:N^v} \quad E = \{(\mathbf{e}_k, r_k, s_k)\}_{k=1:N^e}$$



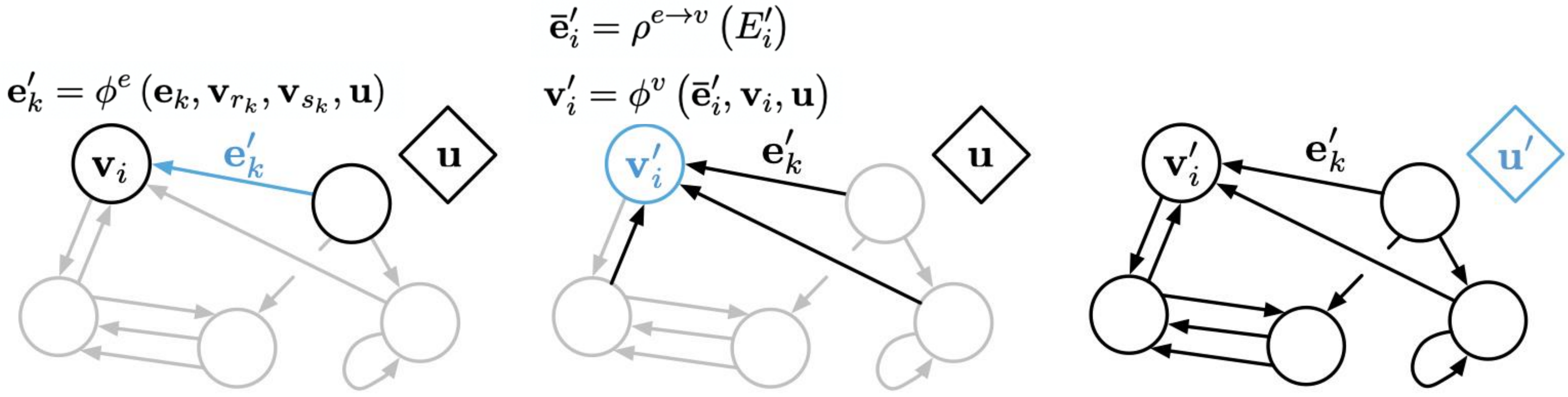
GraphNetwork (GN) framework



(a) Edge update (b) Node update (c) Global update

ϕ - Update functions per variable (e.g. node / edge)

GraphNetwork (GN) framework



(a) Edge update (b) Node update (c) Global update

- Aggregation functions ρ must be invariant to permutations of the inputs and take variable number of inputs

GraphNetwork (GN) framework

$$\mathbf{e}'_k = \phi^e\left(\mathbf{e}_k, \mathbf{v}_{r_k}, \mathbf{v}_{s_k}, \mathbf{u}\right)$$

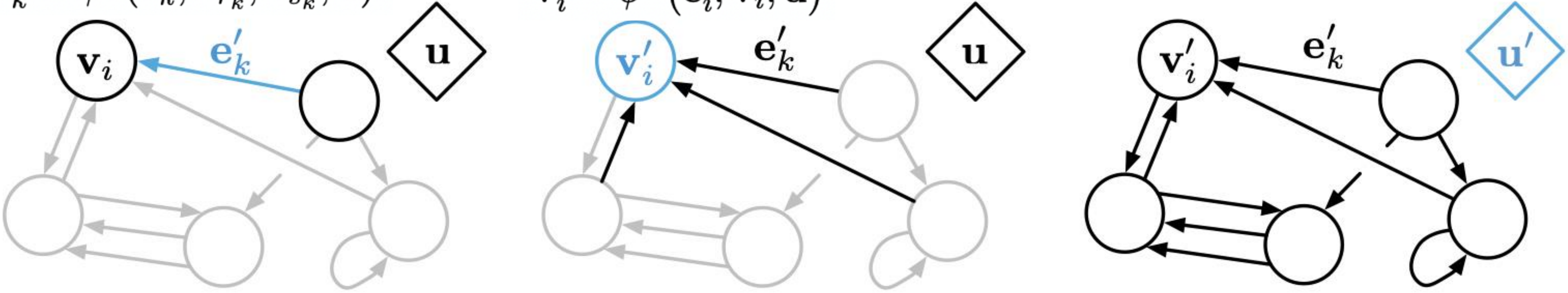
$$\bar{\mathbf{e}}'_i = \rho^{e\to v}\left(E'_i\right)$$

$$\mathbf{v}'_i = \phi^v\left(\bar{\mathbf{e}}'_i, \mathbf{v}_i, \mathbf{u}\right)$$

$$\bar{\mathbf{v}}' = \rho^{v\to u}\left(V'\right)$$

$$\bar{\mathbf{e}}' = \rho^{e\to u}\left(E'\right)$$

$$\mathbf{u}' = \phi^u\left(\bar{\mathbf{e}}', \bar{\mathbf{v}}', \mathbf{u}\right)$$



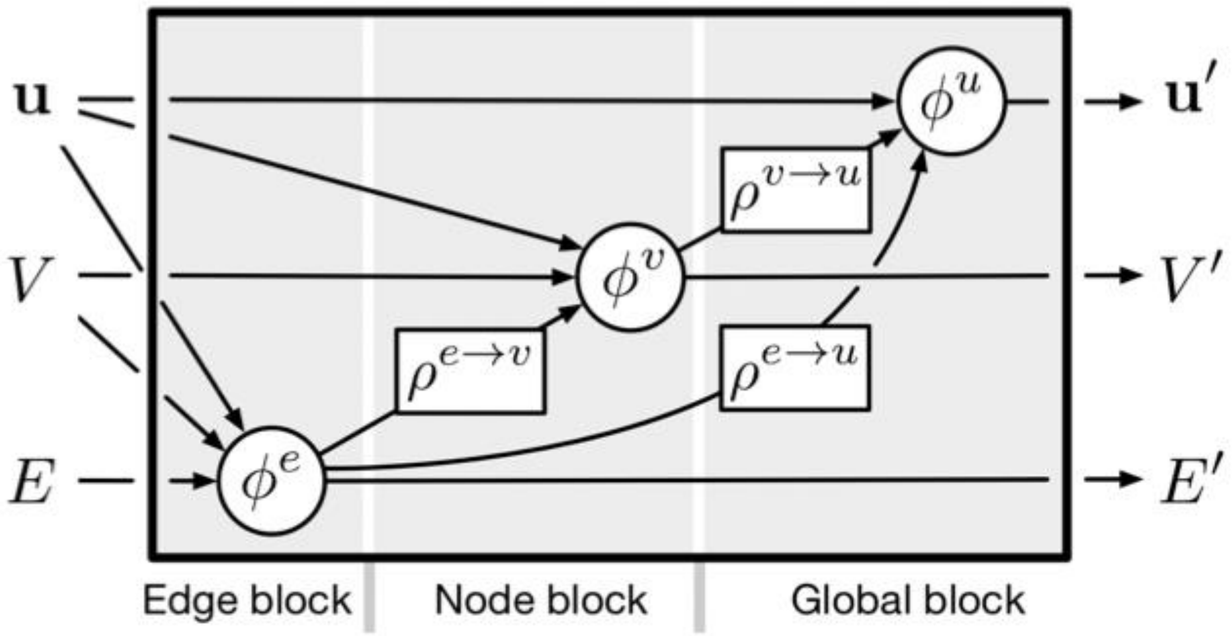
(a) Edge update (b) Node update (c) Global update

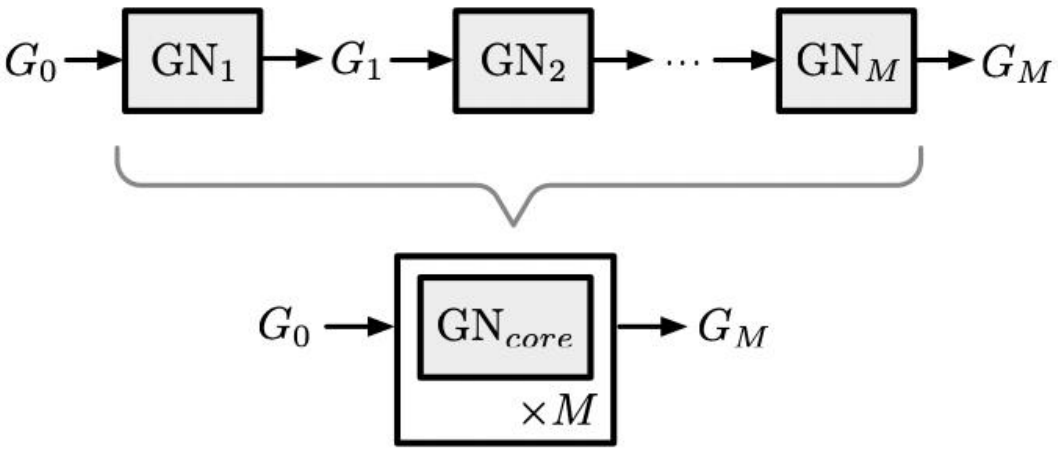
- Aggregation functions ρ must be invariant to permutations of the inputs and take variable number of inputs

GraphNetwork (GN) framework

function GRAPHNETWORK(E, V, $\mathbf{u}$)
  for $k \in \{1 \ldots N^e\}$ **do**
    $\mathbf{e}'_k \leftarrow \phi^e\left(\mathbf{e}_k, \mathbf{v}_{r_k}, \mathbf{v}_{s_k}, \mathbf{u}\right)$ ▷ 1. Compute updated edge attributes
  end for
  for $i \in \{1 \ldots N^n\}$ **do**
    let $E'_i = \{(\mathbf{e}'_k, r_k, s_k)\}_{r_k = i,\; k=1:N^e}$
    $\bar{\mathbf{e}}'_i \leftarrow \rho^{e \rightarrow v}\left(E'_i\right)$ ▷ 2. Aggregate edge attributes per node
    $\mathbf{v}'_i \leftarrow \phi^v\left(\bar{\mathbf{e}}'_i, \mathbf{v}_i, \mathbf{u}\right)$ ▷ 3. Compute updated node attributes
  end for
  let $V' = \{\mathbf{v}'\}_{i=1:N^v}$
  let $E' = \{(\mathbf{e}'_k, r_k, s_k)\}_{k=1:N^e}$
  $\bar{\mathbf{e}}' \leftarrow \rho^{e \rightarrow u}\left(E'\right)$ ▷ 4. Aggregate edge attributes globally
  $\bar{\mathbf{v}}' \leftarrow \rho^{v \rightarrow u}\left(V'\right)$ ▷ 5. Aggregate node attributes globally
  $\mathbf{u}' \leftarrow \phi^u\left(\bar{\mathbf{e}}', \bar{\mathbf{v}}', \mathbf{u}\right)$ ▷ 6. Compute updated global attribute
  return $(E', V', \mathbf{u}')$
end function

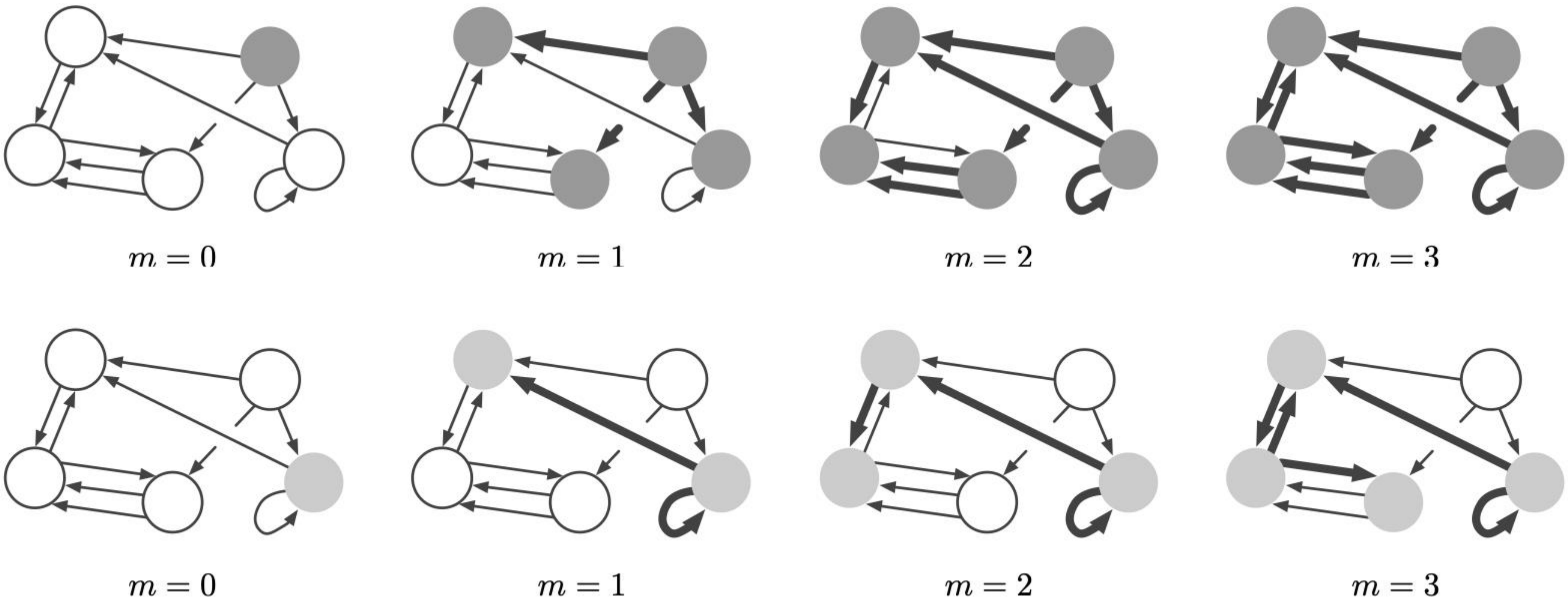
GraphNetwork (GN) framework





GraphNetwork (GN) framework

- Multi-step message passing



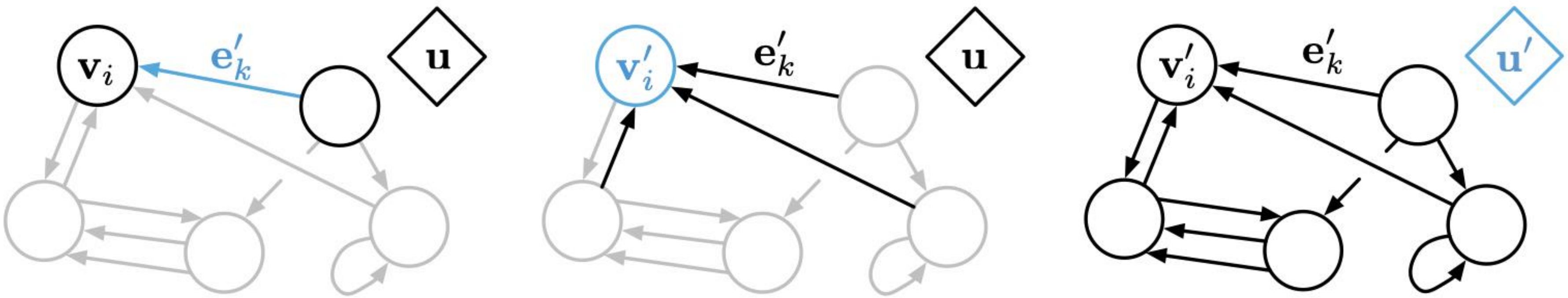
GraphNetwork (GN) framework

- Inference can be based on:

Vertices: inferring properties of each entity

Edges: inferring relationships of vertices

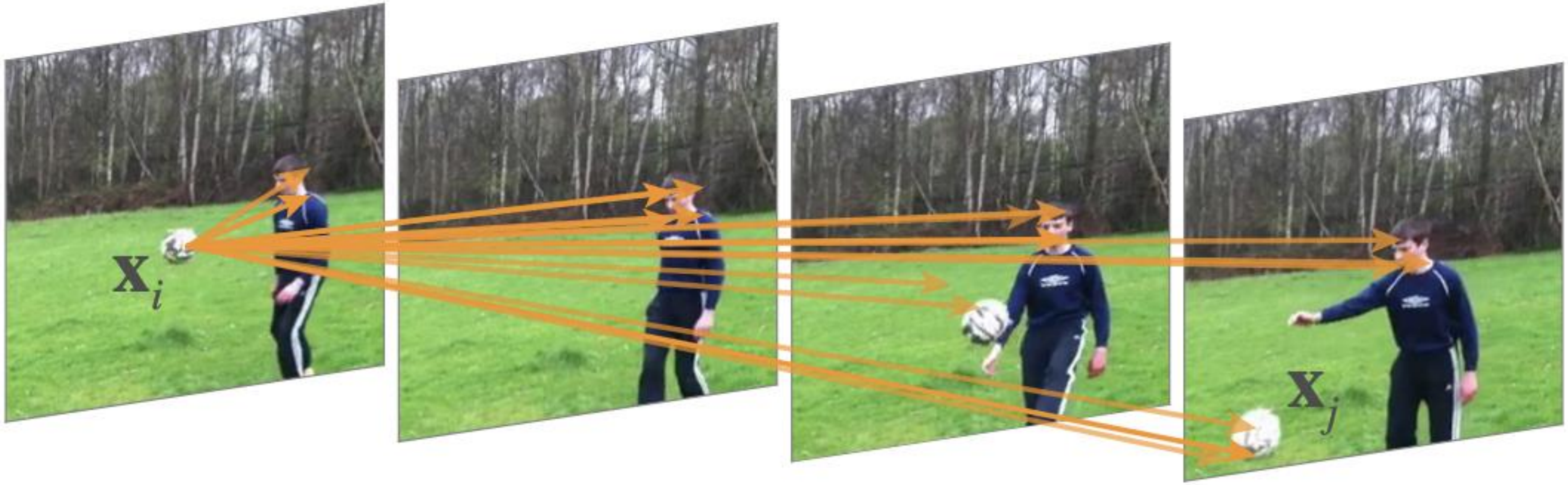
Global representation: inferring properties of the whole system



(a) Edge update (b) Node update (c) Global update

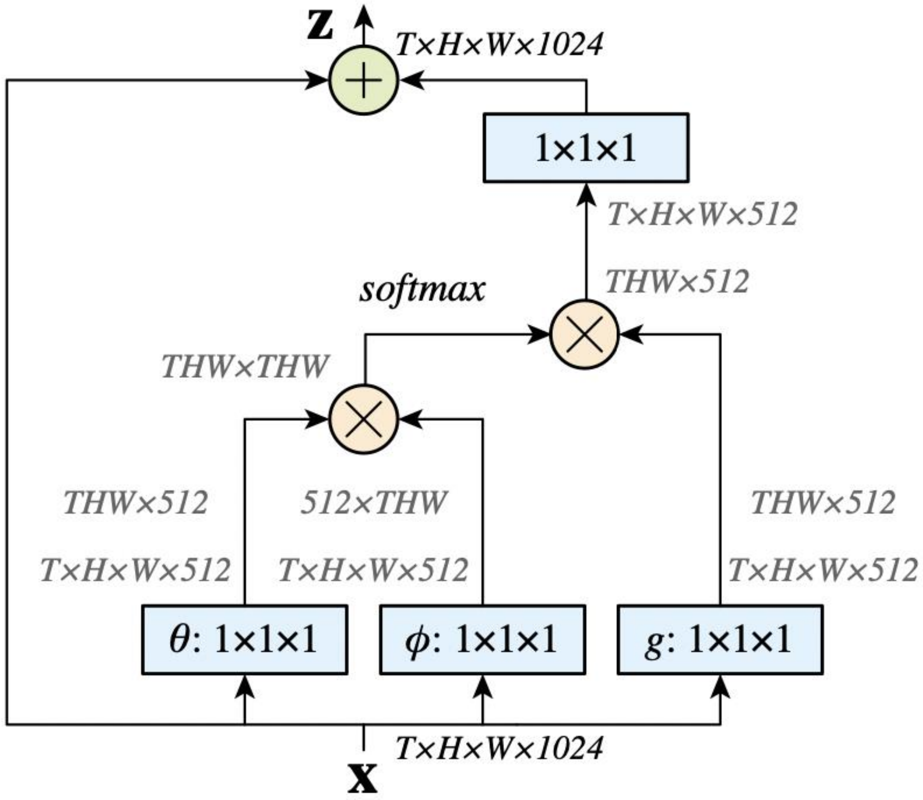
Previous works interpreted as GN framework

- Non-Local Neural Networks (Wang et al. 2018)



Previous works interpreted as GN framework

- Non-Local Neural Networks (Wang et al. 2018)



$$\phi^e\left(\mathbf{e}_k, \mathbf{v}_{r_k}, \mathbf{v}_{s_k}, \mathbf{u}\right) := f^e\left(\mathbf{v}_{r_k}, \mathbf{v}_{s_k}\right)$$

$$\phi^v\left(\bar{\mathbf{e}}'_i, \mathbf{v}_i, \mathbf{u}\right) := f^v(\bar{\mathbf{e}}'_i)$$

$$\rho^{e\to v}\left(E'_i\right) := \frac{1}{\sum_{\{k:\, r_k=i\}} a'_k} \sum_{\{k:\, r_k=i\}} a'_k \mathbf{b}'_k$$

Limitations

- Graph Networks perform well for tasks that require relational reasoning but they crucially need the edges pre-defined.
    - Not easy to modify graph structure after initialization
- No experimental results
    - Comparison of different variants would provide some insights

Limitations

- This a great review paper and it unites different graph network architectures as a single general framework. But it would have been better if there are some critical insights:
    - Why is having such a general framework beneficial?
    - What knowledge/insight do we get out of doing this unification process and how can we make GNs better?

Summary

- This is a position paper that argues “combinatorial generalization” must be a top priority for AI to achieve human-level intelligence
- Analyzes different kinds of inductive biases in neural network models
- Proposes a general formulation of Graph Networks