GNN

Xi Chen

xchen595@163.com

Keywords
combinatorial generalization, relational inductive biases, Graph Network

Graph Network Architectures

In general, the framework is agnostic to specific attribute representations and functional forms. (we focus mainly on deep learning architectures.)

Graph networks support highly flexible graph representations in two ways: - 1. in terms of the representation of the attributes; - 2. in terms of the structure of the graph itself.

Attributes

The requirements of the problem will often determine what representations should be used for the attributes.

The output of a GN block can also be tailored to the demands of the task. In particular,

- An edge-focused GN uses the edges as output, for example to make decisions about interactions among entities *(Kipf et al., 2018; Hamrick et al., 2018).*
- A node-focused GN uses the nodes as output, for example to reason about physical systems *(Battaglia et al., 2016; Chang et al., 2017; Wang et al., 2018b; Sanchez-Gonzalez et al., 2018).*
- A graph-focused GN uses the globals as output, for example to predict the potential energy of a physical system *(Battaglia et al., 2016)*, the properties of a molecule *(Gilmer et al., 2017)*, or answers to questions about a visual scene *(Santoro et al., 2017).*
- both the output edge and global attributes to compute a policy over actions *(Hamrick et al. 2018).*

Graph structure

There are generally two scenarios:

- first, the input explicitly specifies the relational structure *[knowledge graphs, social networks, parse trees, optimization problems, chemical graphs, road networks, and physical systems with known interactions.].*
- second, the relational structure must be inferred or assumed *(Vaswani et al., 2017; Watters et al., 2017; Santoro et al., 2017; Wang et al., 2018c; Kipf et al., 2018).*

Definition of "Graph"

Within GN framework,

A *graph* is defined as a 3-tuple $\mathbf{G} = (\mathbf{u}, \mathbf{V}, \mathbf{E})$.

- The $\mathbf{u}$ is a global attribute
- $v = \{v_i\}_{i=1:N^v}$ is the set of nodes (N^v is cardinality), where each v_i is a node's attribute.
- $E = \{(e_k, r_k, s_k)\}_{k=1:N^e}$ is the set of edges (N^e is cardinality), where each e_k is the edge's attribute, r_k is the index of the receiver node, and s_k is the index of the sender node.

GN block inner structure

3 "update" function ϕ and 3 "aggregation" function ρ:

$$\begin{aligned} e_k^{'} &= \phi^e(e_k, v_{r_k}, v_{s_k}, u) & \bar{e}_i^{'} &= \rho^{e\to v}(E_i^{'}) \\ v_i^{'} &= \phi^v(\bar{e}_i^{'}, v_i, u) & \bar{e}^{'} &= \rho^{e\to u}(E^{'}) \\ u^{'} &= \phi^u(\bar{e}^{'}, \bar{v}^{'}, u) & \bar{v}_i^{'} &= \rho^{v\to u}(V^{'}) \end{aligned}$$

Design Principle behind GN

1. flexible representations
2. configurable within-block structure
3. composable multi-block architectures

Open Questions

- where do the grasphs come from that graph networks operate over (it is unclear the best ways to convert sensory data into more structured representations like graphs.)
- many underlying graph structures are much more sparse than a fully connected graph (how to induce sparsity)
- how to adaptively modify graph structures during the course of computation
- further explore the interpretability of the behavior of graph networks