Message Passing Graph Neural Networks

Polina Barabanshchikova

MIPT

November 6, 2022

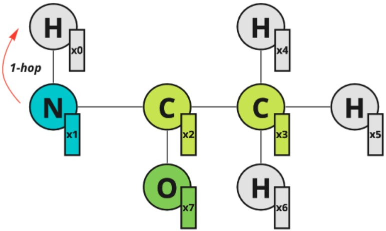
Motivation

Graph-structured data

- Spatial structure
- Node features
- Permutation invariance

Intuition

$$\underbrace{\begin{bmatrix} 1&1&0&0&0&0&0&0\\ 1&1&1&0&0&0&0&0\\ 0&1&1&1&0&0&0&1\\ 0&0&1&1&1&1&1&0\\ 0&0&0&1&1&0&0&0\\ 0&0&0&1&0&1&0&0\\ 0&0&0&1&0&0&1&0\\ 0&0&1&0&0&0&0&1 \end{bmatrix}}_{\mathbf{I+A}} \times \underbrace{\begin{bmatrix} x0\\ x1\\ x2\\ x3\\ x4\\ x5\\ x6\\ x7 \end{bmatrix}}_{\mathbf{X}} = \underbrace{\begin{bmatrix} x0+x1\\ x0+x1+x2\\ x1+x2+x3+x7\\ x2+x3+x4+x5+x6\\ x3+x4\\ x3+x5\\ x3+x6\\ x2+x7 \end{bmatrix}}_{\mathbf{X'}}$$



$$x_i' = update(x_i, aggregate([x_j, j \in N(i)]))$$

https://towardsdatascience.com

Problem statement

Given a graph $G = (V, E)$ with node features $x_v, v \in V$ and edge features $e_{vw}, (v, w) \in E$, there are two types of tasks

- Predict label or value for graph classification or regression
- Predict label or value for each node or edge in structured prediction

Method description

Message passing neural networks (MPNN)
Let h_v^t represent the node embedding for some vertex v at iteration t.
1. Initialization

$$h^{(0)}v = x_v \quad \forall v \in V$$

Method description

Message passing neural networks (MPNN)

Let h_v^t represent the node embedding for some vertex v at iteration t.

1. Initialization

$$h^{(0)}v = x_v \quad \forall v \in V$$

2. Message passing phase For $1 \leq t < T$

$$m_v^{t+1} = \sum_{w \in N(v)} M_t(h_v^t, h_w^t, e_{vw})$$

$$h_v^{t+1} = U_t(h_v^t, m_v^{t+1})$$

where $N(v)$ denotes the neighbors of v in G.

Method description

Message passing neural networks (MPNN)

Let h_v^t represent the node embedding for some vertex v at iteration t.

1. Initialization

$$h^{(0)}v = x_v \quad \forall v \in V$$

2. Message passing phase For $1 \leq t < T$

$$m_v^{t+1} = \sum_{w \in N(v)} M_t(h_v^t, h_w^t, e_{vw})$$

$$h_v^{t+1} = U_t(h_v^t, m_v^{t+1})$$

where $N(v)$ denotes the neighbors of v in G.

3. Readout phase

$$\hat{y} = R(\{h_v^T | v \in G\})$$

Message passing function

- **Concatenation**
$$M_t(h_v^t, h_w^t, e_{vw}) = (h_w^t, e_{vw})$$
- **Matrix Multiplication**
$$M_t(h_v^t, h_w^t, e_{vw}) = A_{e_{vw}} h_w^t$$
(for discrete edge types)
- **Edge Network**
$$M_t(h_v^t, h_w^t, e_{vw}) = A(e_{vw}) h_w^t,$$
where $A(e_{vw})$ is a neural network which maps the edge vector to a $d \times d$ matrix
- **Pair Message**
$$M_t(h_v^t, h_w^t, e_{vw}) = f(h_v^t, h_w^t, e_{vw}),$$
where f is a neural network

Update function

- **Sum**

$$U_t(h_v^t, m_v^{t+1}) = h_v^t + m_v^{t+1}$$

- **MLP**

$$U_t(h_v^t, m_v^{t+1}) = \sigma(H_t^{deg(v)} m_v^{t+1})$$

- **RNN**

$$U_t(h_v^t, m_v^{t+1}) = GRU(h_v^t, m_v^{t+1})$$

the same update function at each time step t

Readout function

- **Neural Network**

$$R = f(\sum_{v \in G} h_v^T)$$

$$R = \sum_{v \in G} f(h_v^T)$$

- **Neural Network with skip connections**

$$R = f\left(\sum_{v,t} \text{softmax}(W_t h_v^t)\right)$$

- **Two Neural Networks**

$$R = \sum_{v \in G} \sigma\left(f_1(h_v^T, h_v^0)\right) \odot \left(f_2(h_v^T)\right),$$

where $\odot$ denotes elementwise multiplication

Virtual graph elements

Allow information to travel long distances during the propagation phase

- **Virtual edge** for pairs of nodes that are not connected
- **Master node** which is connected to every input node in the graph with a special edge type. Master is allowed to have a separate node dimension

Application

Molecular property prediction

Experiments

- Any $T \geq 3$ works
- Complete edge feature vector (bond type, spatial distance) and explicit hydrogen work better
- Common weights and a large hidden dimension are beneficial
- The pair message function performs worse than the edge network function

Literature

[1] Justin Gilmer et al. *Neural Message Passing for Quantum Chemistry*. 2017. DOI: 10.48550/ARXIV.1704.01212. URL: https://arxiv.org/abs/1704.01212.

Questions

1. What function do virtual elements perform?
2. Describe the three stages of MPNN