Hyperparameters used for GNNs

Graph Attention Neural Network (GANN)

Note: In the original Graph Attention Network by Banik et al. it is hard-coded that the number of angle convolutions = (number of edge convolutions – 1)

The best choice for num\_workers depends on your system, on some systems it gives a warning when num\_workers > 1, some systems it will crash if num\_workers != 0

|  |  |
| --- | --- |
| **parameter** | **value** |
| angle\_fea\_len | 80 |
| batch\_size | 4 |
| bond\_fea\_len | 80 |
| checkpoint\_every | 20 |
| embedding | false |
| epochs | 1000 |
| gamma | 0.5 |
| h\_fea\_angle | 256 |
| h\_fea\_edge | 256 |
| learning\_rate | 0.01 |
| n\_classification | 6 |
| n\_conv\_edge | 2 |
| n\_conv\_angle | 1 |
| neighbors | 15 |
| num\_workers | 4 |
| pin\_memory | true |
| rcut | 6 |
| resume | true |
| search\_delta | 20 |
| search\_type | global |
| step\_size | 100 |
| test\_size | 0 |
| train\_size | 9600 |
| val\_size | 600 |
| num\_MLP | 0 |

Heterogeneous Graph Attention Neural Network (het-GANN)

|  |  |
| --- | --- |
| **parameter** | **value** |
| angle\_fea\_len | 80 |
| batch\_size | 4 |
| bond\_fea\_len | 80 |
| checkpoint\_every | 20 |
| embedding | false |
| epochs | 1000 |
| gamma | 0.5 |
| h\_fea\_angle | 256 |
| h\_fea\_edge | 256 |
| learning\_rate | 0.01 |
| n\_classification | 6 |
| n\_conv\_edge | 2 |
| n\_conv\_angle | 1 |
| neighbors | 15 |
| num\_workers | 4 |
| pin\_memory | true |
| rcut | 6 |
| resume | true |
| search\_delta | 20 |
| search\_type | global |
| step\_size | 100 |
| test\_size | 0 |
| train\_size | 9600 |
| val\_size | 600 |
| num\_MLP | 0 |

GraphSAGE

|  |  |
| --- | --- |
| **parameter** | **value** |
| angle\_fea\_len | 80 |
| batch\_size | 4 |
| bond\_fea\_len | 80 |
| checkpoint\_every | 5 |
| embedding | false |
| epochs | 1000 |
| gamma | 0.5 |
| h\_fea\_angle | 128 |
| h\_fea\_edge | 128 |
| learning\_rate | 0.001 |
| n\_classification | 6 |
| n\_conv\_edge | 2 |
| n\_conv\_angle | 2 |
| neighbors | 15 |
| num\_workers | 4 |
| pin\_memory | true |
| rcut | 6 |
| resume | true |
| search\_delta | 20 |
| search\_type | global |
| step\_size | 100 |
| test\_size | 0 |
| train\_size | 9600 |
| val\_size | 600 |
| num\_MLP | 0 |

Heterogeneous GraphSAGE

|  |  |
| --- | --- |
| **parameter** | **value** |
| angle\_fea\_len | 80 |
| batch\_size | 4 |
| bond\_fea\_len | 80 |
| checkpoint\_every | 5 |
| embedding | false |
| epochs | 1000 |
| gamma | 0.5 |
| h\_fea\_angle | 128 |
| h\_fea\_edge | 128 |
| learning\_rate | 0.0001 |
| n\_classification | 6 |
| n\_conv\_edge | 2 |
| n\_conv\_angle | 2 |
| neighbors | 15 |
| num\_workers | 4 |
| pin\_memory | true |
| rcut | 6 |
| resume | true |
| search\_delta | 20 |
| search\_type | global |
| step\_size | 100 |
| test\_size | 0 |
| train\_size | 9600 |
| val\_size | 600 |
| num\_MLP | 0 |

Graph Isomorphism Network (GIN)

|  |  |
| --- | --- |
| **parameter** | **value** |
| angle\_fea\_len | 80 |
| batch\_size | 4 |
| bond\_fea\_len | 80 |
| checkpoint\_every | 20 |
| embedding | false |
| epochs | 1000 |
| gamma | 0.5 |
| h\_fea\_angle | 32 |
| h\_fea\_edge | 32 |
| learning\_rate | 0.01 |
| n\_classification | 6 |
| n\_conv\_edge | 2 |
| n\_conv\_angle | 2 |
| neighbors | 15 |
| num\_workers | 4 |
| pin\_memory | true |
| rcut | 6 |
| resume | true |
| search\_delta | 20 |
| search\_type | global |
| step\_size | 100 |
| test\_size | 0 |
| train\_size | 9600 |
| val\_size | 600 |
| num\_MLP | 0 |

Heterogeneous Graph Isomorphism Network (het-GIN)

|  |  |
| --- | --- |
| **parameter** | **value** |
| angle\_fea\_len | 80 |
| batch\_size | 4 |
| bond\_fea\_len | 80 |
| checkpoint\_every | 20 |
| embedding | false |
| epochs | 1000 |
| gamma | 0.5 |
| h\_fea\_angle | 32 |
| h\_fea\_edge | 32 |
| learning\_rate | 0.01 |
| n\_classification | 6 |
| n\_conv\_edge | 2 |
| n\_conv\_angle | 2 |
| neighbors | 15 |
| num\_workers | 4 |
| pin\_memory | true |
| rcut | 6 |
| resume | true |
| search\_delta | 20 |
| search\_type | global |
| step\_size | 100 |
| test\_size | 0 |
| train\_size | 9600 |
| val\_size | 600 |
| num\_MLP | 0 |

Relational Graph Convolutional Network (RGCN)

|  |  |
| --- | --- |
| **parameter** | **value** |
| angle\_fea\_len | 80 |
| batch\_size | 4 |
| bond\_fea\_len | 80 |
| checkpoint\_every | 1 |
| embedding | false |
| epochs | 1000 |
| gamma | 0.5 |
| h\_fea\_angle | 64 |
| h\_fea\_edge | 64 |
| learning\_rate | 0.01 |
| n\_classification | 6 |
| n\_conv\_edge | 2 |
| n\_conv\_angle | 2 |
| neighbors | 15 |
| num\_workers | 4 |
| pin\_memory | true |
| rcut | 6 |
| resume | true |
| search\_delta | 20 |
| search\_type | global |
| step\_size | 100 |
| test\_size | 0 |
| train\_size | 9600 |
| val\_size | 600 |

Note: the hidden size (h\_fea\_edge, h\_fea\_angle) and number of convolutions (n\_conv\_edge, n\_conv\_angle) are hard-coded in the RGCN and tunable in all of the other networks.