

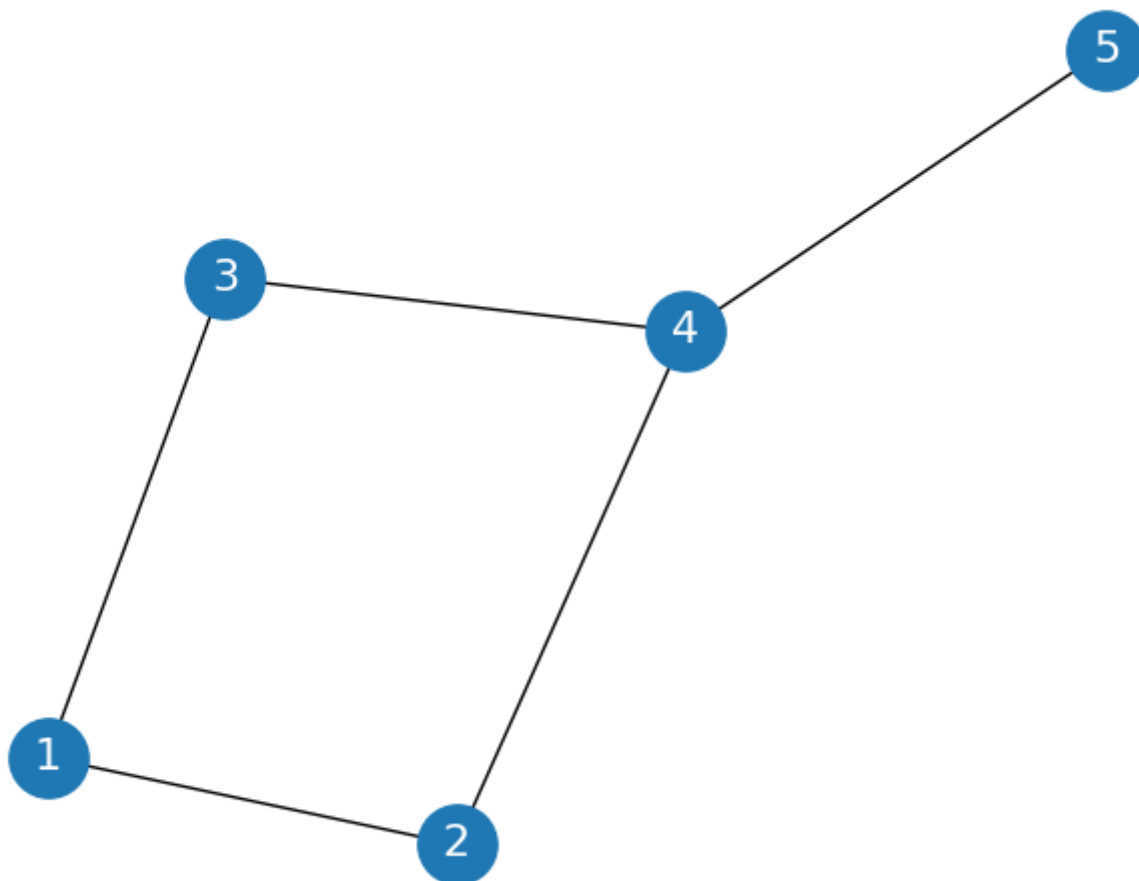
```

8   $\mathbf{h}_u^{(0)} \leftarrow x_v, \forall v \in B^{(0)};$ 
9  for  $k = K \dots 1$  do
10     for  $u \in B^{(k)}$  do
11          $\mathbf{h}_{N(u)}^{(k)} \leftarrow \text{Agg}^{(k)}\left(\left\{\mathbf{h}_{u'}^{(k-1)}, \forall u' \in N^{(k)}(u)\right\}\right);$ 
12          $\mathbf{h}_u^{(k)} \leftarrow \sigma\left(W^k[\mathbf{h}_u^{(k-1)} \parallel \mathbf{h}_{N(u)}^k]\right);$ 
13          $\mathbf{h}_u^{(k)} \leftarrow \mathbf{h}_u^{(k)} / \|\mathbf{h}_u^{(k)}\|_2;$ 
14     end
15 end
16  $\mathbf{z}_u \leftarrow \mathbf{h}_u^{(k)}, \forall u \in B$ 

```

数据准备

对于下面这样一个图：



假如我们想要节点的嵌入输出的outputdim为2维的，输入的图对应初始特征数据如下：

1. 邻居节点聚合

我们先完成第一步，邻居节点聚合。
这一步可以采用多种聚合算子，我们在此以平均聚合算子与平均池化聚合算子为例。

1.1 平均聚合算子

节点4的邻接节点为2, 3, 5
我们用平均聚合算子AVERAGE(h)对2, 3, 5这三个节点的信息进行聚合：

就是对这三个向量对应值相加取均值，得到：

由上面我们得到 为：

为了使得输出是一个 的向量，我们设置 为一个 的矩阵，比如我们初始化为：

与上面的向量相乘，得到

将其代入激活函数后进行归一化，就正式得到

对第一层的其他节点的操作也是类似的。

1.2 平均池化聚合算子

那么平均池化聚合算子呢，就是 avg_pool ，是激活函数(ReLU)
具体怎么做的呢？

我们待处理的邻接节点：

此刻我们在此添加一个dense层（全连接层）

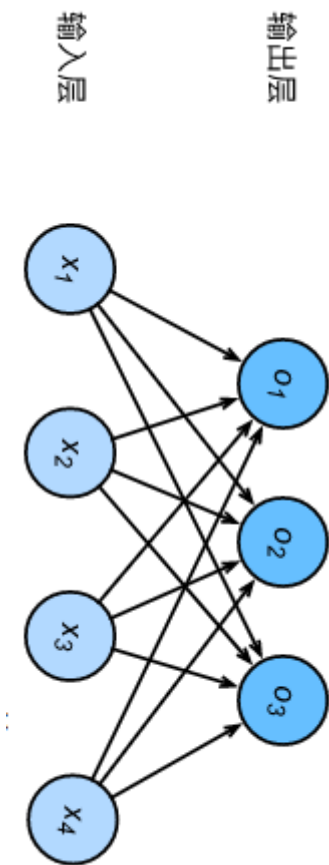
这里全连接层的意义就是对这每个节点，其向量的每个分量经过了一个全连接层。

我们知道全连接的核心是矩阵乘法，对于 \mathbf{W} ，输出是：

我们引入一个权重矩阵 \mathbf{W} ，它是input dim×hidden dim维度的，在这里input dim是4维，我们设定这个隐藏层的输出维度为3，那么W就是一个4×3的矩阵

b就是一个hidden_dim(3)维的向量。

这样的话：



权重的初始化有多种方式，我们在此用随机数初始化：

col0	col1	col2
0.17	0.75	0.53
0.4	-0.04	-0.04
-0.14	0.67	0.4
0.51	-0.18	0.77

对于2,3,5,得到的 分别为

将上述三个向量经过激活函数ReLU：
就得到(对每个分量分别取ReLU)

这样我们就完成了第一步；

由上我们得到

而

因为经过了一个平均池化聚合，邻居节点的输出变成了3维的，所以GraphSAGE对当前节点与邻居节点分别引入了一个权重矩阵：

因为我们想要输出是2维的，那么

和 就分别是

4×2维的和3×2维的；

类似上面，假如我们要在这里加一个bias的话，bias的维度应当是一个2维的向量

将计算得到的结果经过一个ReLU函数再除以它的模进行归一化进行输出，就完成了聚合操作。