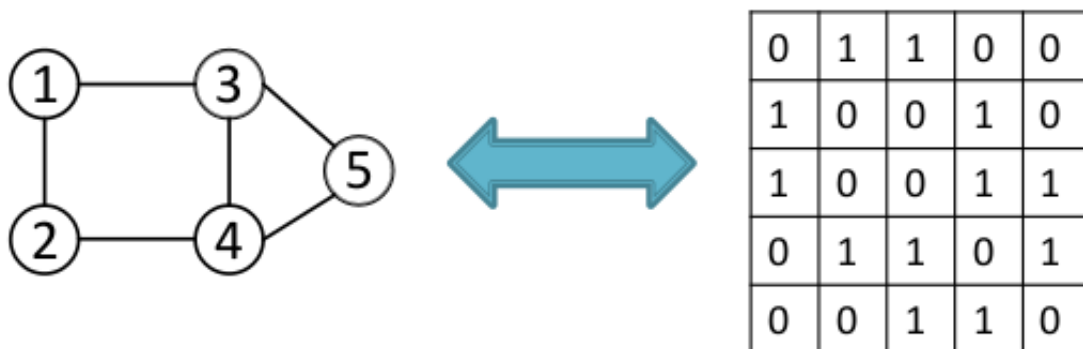


Generative Models for Graphs

在节点表示学习这节中，我们看到了几种在保留空间和网络结构相似性的同时对嵌入空间中的图进行“编码”的方法。在本节中，我们将研究如何表达图的节点和边之间的概率依赖性，以及如何通过从学习的分布中抽取样本，生成新的真实的图。这种捕获特定图形族分布的能力有许多应用。例如，从在特定图族上训练的生成模型中采样图可能会发现新配置，而且这些新配置的全局特性相同，比如，药物发现的过程。该方法的另一个应用是能够在真实世界的图形中模拟“假设”场景，以收集有关网络属性。

Challenges

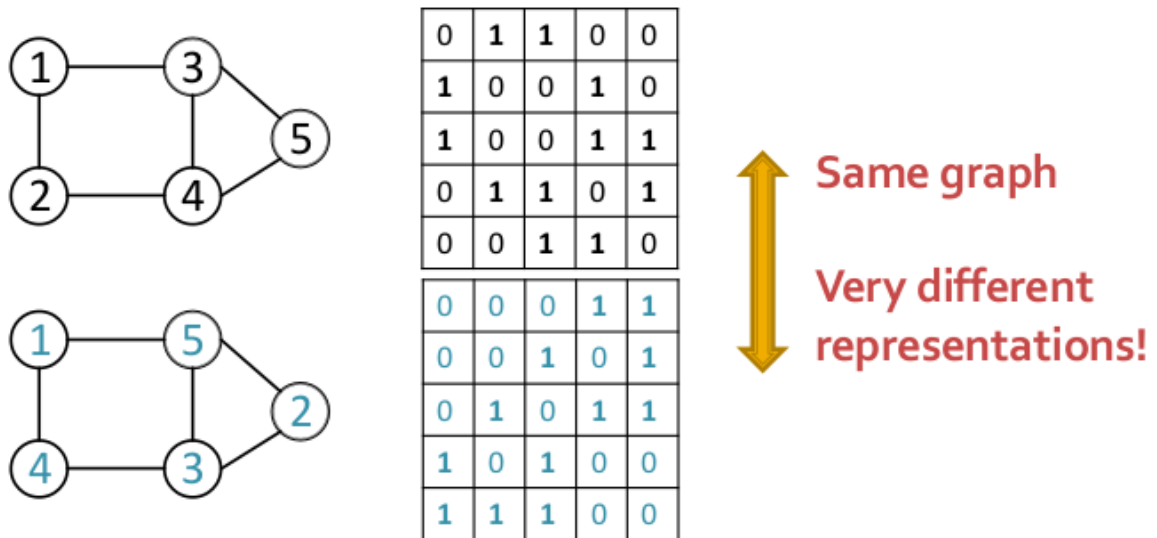
1. 对于具有 n 个节点的图可能具有 $O(n^2)$ 个可能的边。在预测图形边的时候可能会导致平方爆炸问题。



5 nodes: 25 values

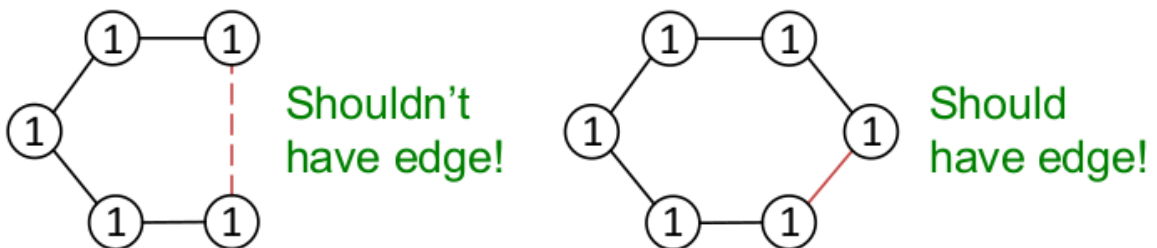
1K nodes: 1M values

2. n -节点图将会有 $n!$ 种表示方法，这使得优化目标函数非常困难，2个非常不同的邻接矩阵可能会表示相同的图结构。



3. 边缘形成可能具有长期依赖性(例如，要生成一个具有6个节点的循环的图，需要记住到目前为止的结构)

Example: Generate a ring graph on 6 nodes:



Terminology

1. $p_{data}(G)$: 给定图采样后的概率分布。
2. $p_{data}(G; \theta)$: 从 $p_{data}(G)$ 学习到的分布，参数为 θ 。

Goal: Our goal is 2-fold

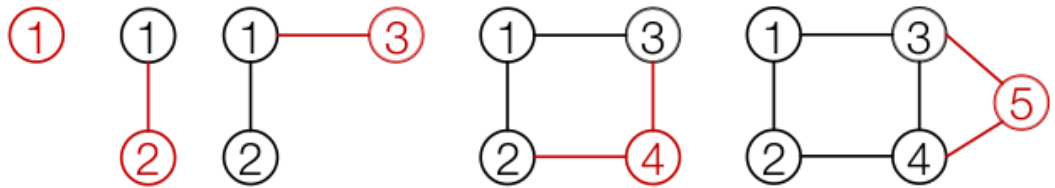
1. 确保 $p_{data}(G; \theta)$ 尽量接近 $p_{data}(G)$ (核心思想: 最大似然估计)
2. 此外，我们还需要确保可以有效地从 $p_{data}(G; \theta)$ 中进行采样(核心思想: 从噪声分布中采样并通过复杂函数转换采样噪声以生成图)

GraphRNN

这个想法是将图形生成的任务视为序列生成任务。给定先前的动作状态，我们想对下一个“动作”的概率分布建模。在语言建模中，“动作”是我们试图预测的词。对于图生成，“动作”则是添加节点/边。如上所述，图可以具有 $O(n!)$ 个与之相关的序列，但是我们可以通过对图的节点进行排序来绘制唯一的序列。

固定节点序列后，我们可以映射需要将相应边添加到图中的序列。因此，图生成的任务可以等效地转换为两个级别的序列生成问题，首先是节点级别，其次是边级别。由于RNN以其序列生成功能而闻名，因此我们将研究如何将其用于此任务。

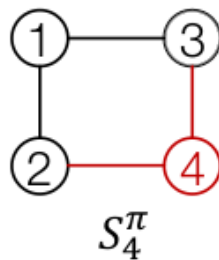
■ **Node-level:** At each step, a new node is added



$$S^\pi = (S_1^\pi, S_2^\pi, S_3^\pi, S_4^\pi, S_5^\pi)$$

“Add node 1”
...
“Add node 5”

■ **Edge-level:** At each step, add a new edge



$$S_4^\pi = (S_{4,1}^\pi, S_{4,2}^\pi, S_{4,3}^\pi)$$

“Not connect 4, 1”
“Connect 4, 2”
“Connect 4, 3”

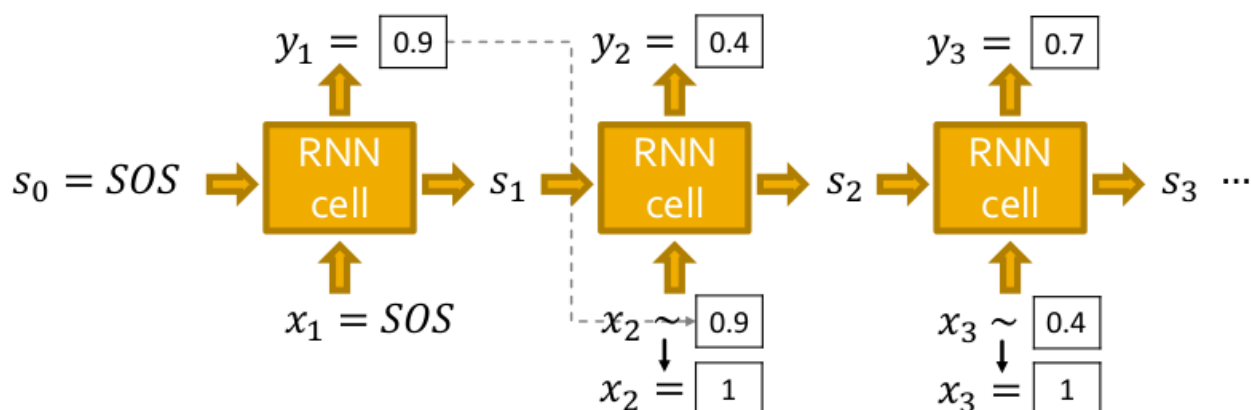
0
1
1

GraphRNN具有节点级RNN和边级RNN。这两个RNN的关系如下：

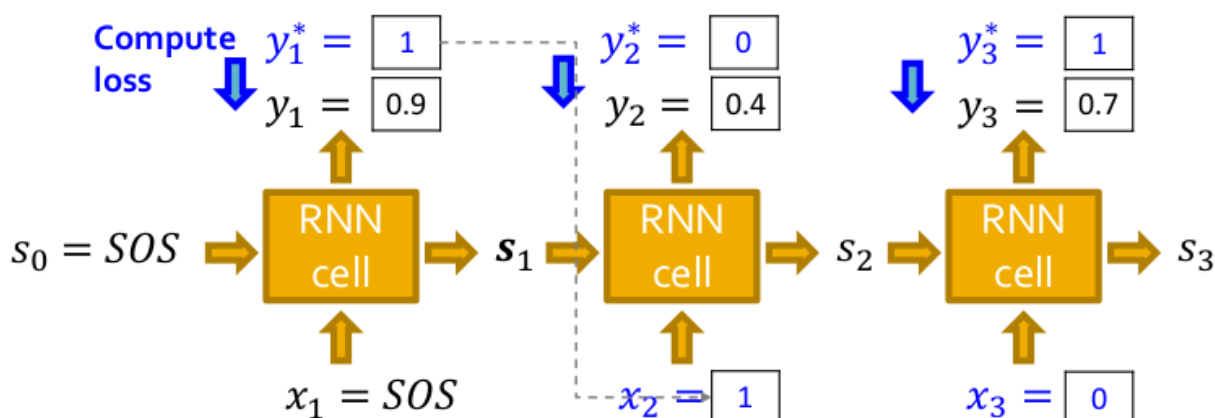
1. 节点级别的RNN为边级别的RNN生成初始状态
2. 边级别的RNN为新节点生成边，然后使用生成的结果更新节点级RNN的状态

因此有以下架构。注意，该模型是自回归的，因为当前RNN单元的输出将作为输入馈送到下一个RNN单元。此外，为使模型具有更好地表述能力并为概率分布建模，每个单元的输入是从其先前假定伯努利分布的输出单元中采样。因此，在推理时，我们只需传递特殊的“序列开始(SOS)”令牌即可开始序列生成过程，该过程一直持续到生成“序列结束(EOS)”令牌为止。

- y_t follows **Bernoulli distribution** (choice of p_{model})
- \boxed{p} means value 1 has prob. p , value 0 has prob. $1 - p$

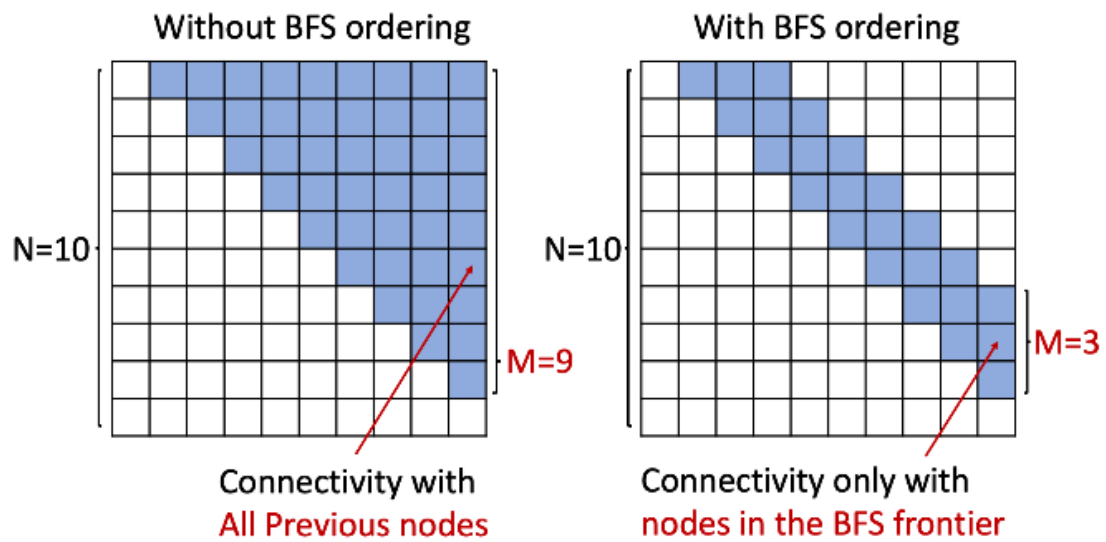


现在，我们已经知道了如何在训练好的模型的情况下生成图形。但是我们如何训练呢？我们使用 Teacher-forcing 技术训练模型，并用实际序列替换输入和输出，如下所示，并使用标准 binary cross-entropy 损失作为优化目标，并通过时间反向传播 (BPTT) 以更新模型参数。



现在，我们可以通过从模型学习的分布中进行采样来生成图。但是，主要挑战仍然存在。由于任何节点都可以连接到任何先前的节点，因此我们需要生成邻接矩阵的一半，由于二次平方爆炸的问题，邻接矩阵的效率极低。为了解决这个问题，我们以 BFS 方式生成节点序列。这将可能的节点顺序从 $O(n!)$ 降低到相对较小的 BFS 顺序。并且还减少了边生成的步骤数(因为现在该模型不需要检查所有节点的连通性，因为该节点只能连接到 BFS 树中的其前任节点) 如下图所示。

Adjacency matrices



更多阅读: [GraphRNN: Generating Realistic Graphs with Deep Auto-regressive Models \(ICML 2018\)](#)