Stochastic Blockmodels meet Graph Neural Networks

2025 年 3 月 20 日

研究背景

- **随机块模型（SBM）**及其变种（如 MMSB、OSBM）**广泛用于 社区发现**和 **链接预测**任务。
- **图神经网络（GNN）**例如图卷积网络，通过利用图的局部性和不变性特性来学习图中节点的强大表示（嵌入）。
- 基于图结构数据，研究开发了一个 **VAE 的稀疏变体**来统一上述两个方向，既继承了 *SBM* 的可解释性 ，又运用了 *GNN* 的良好预测性能。

研究目标

- 结合 OSBM 与 GNN，提升 **推理效率与 可解释性**。
- **Stick-Breaking 过程**使社区嵌入稀疏化，**自动学习社区数量**。
- 采用 **GCN 变分自编码器**和**随机梯度变分贝叶斯算法（SGVB）**，实现高效推理。

OSBM-基本符号

$$p(A_{nm} = 1 | z_n, z_m, W) = \sigma(z_n^T W z_m)$$

- **邻接矩阵**：$A \in \{0,1\}^{N \times N}$，其中 $A_{nm} = 1$ 表示节点 n 与 m 之间存在边。
- **节点特征矩阵**：$X \in \mathbb{R}^{N \times D}$。
- **潜在特征向量（表隶属关系）**：$z_n \in \{0,1\}^{N \times K}$。
- **社区连接概率矩阵**：$W \in \mathbb{R}^{K \times K}$。

重叠随机块模型（OSBM）

链接概率：

$$p(A_{nm}=1|z_n,z_m,W)=\sigma(z_n^TWz_m)$$

- **通过两个节点的潜在特征向量的双线性函数来定义两个节点之间的链接概率**。
- OSBM 允许 **节点归属多个社区**。
- **问题**：
  - 计算复杂度高（MCMC 推理）。
  - 难以自动学习 K（社区数量）。
  - 缺乏建模 **社区强度**。

Deep Generative OSBM

- 每条边 A_{nm} 与两个潜在嵌入 z_n 与 z_m 相关联:

$$A_{nm} \sim p_\theta(z_n, z_m),$$

p_θ 为图生成器（decoder），由嵌入节点的至少一个非线性变换层组成。

- 节点嵌入施加稀疏性:

$$z_n = b_n \odot r_n, \quad b_n \in \{0,1\}^K, \quad r_n \in R^K$$

  - b_n 为二进制向量，表示节点隶属。
  - r_n 为实值向量，表示连接强度。
  - 比起传统 OSBM 规定 z_n 为严格的二进制向量，本文框架将其建模为一个**稀疏的实值向量**，为节点提供了一个更灵活和信息更丰富的表示。

Deep Generative OSBM

- **VAE Decoder**：Stick-Breaking 生成稀疏嵌入。
- **VAE Encoder**：基于 GCN 进行高效变分推断。

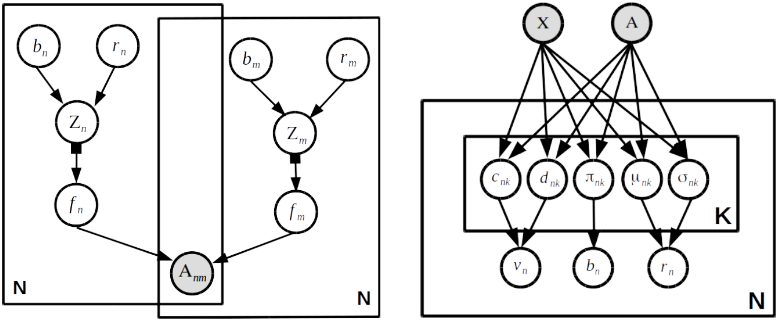

Figure 1. (Left) The generator/decoder model in plate notation. Note that the mapping from z_n to $\boldsymbol{f}_n$ is a deterministic nonlinear transformation, modeled by a deep neural network. (Right) The encoder model, defined by a graph convolutional network (Kipf & Welling, 2016a) that takes as input the network $\mathbf{A}$ and node features $\mathbf{X}$ (if available) and outputs the parameters of the variational distributions of the model parameters.

VAE Generator/Decoder

- 已知节点嵌入 $z_n = b_n \odot r_n$，解码器生成边 $A_{nm} \sim p_\theta(z_n, z_m)$。用 $f_n = f(z_n)$ 表示嵌入 z_n 的整体变换，有连接概率：

$$p(A_{nm} = 1 \mid f_n, f_m) = \sigma(f_n^T f_m)$$

  f 通过 DNN 建模，文中在每个隐藏层使用 **Leaky ReLU**。

- **IBP** 是一种非参数贝叶斯先验，用于建模具有无限多个潜在特征的对象。此处使用 IBP 的**折棍法**（stick-breaking）来生成稀疏的社区归属概率。
  - 生成 **稀疏嵌入**，自动学习社区数量 K。

$$v_k \sim \text{Beta}(\alpha, 1), k = 1, ..., K$$

$$\pi_k = \prod_{j=1}^{k} v_j, \quad b_n \sim \text{Bernoulli}(\pi_k)$$

  - 建模 **社区强度** r_n，增强表达能力。

$$r_n \sim \mathcal{N}(0, \sigma^2 I)$$

VAE Encoder

考虑模型真实后验 $p(v, b, r \mid A, X)$ 的近似：$q_\phi(v, b, r)$。使用平均场近似，**将联合分布分解为独立分布的乘积**：

$$q_\phi(v, b, r) = \prod_{k=1}^{K} \prod_{n=1}^{N} q_\phi(v_{nk}) q_\phi(b_{n,k}) q_\phi(r_{n,k})$$

VAE Encoder

使用 **GCN 编码器**输出每个节点局部变量上的变分分布:

- 输入 network A 和节点特征矩阵 X
- **GCN 每层传播规则**:

$$H^l = g(\hat{A}H^{(l-1)}W^l), \quad H^0 = X, \quad \hat{A}\text{是 A 的对称归一化矩阵}$$

- **变分后验形式**:

$$q_\phi(v_{nk}) = \mathrm{Beta}(c_{nk}, d_{nk}), k = 1, .., K$$

$$q_\phi(b_{nk}) = \mathrm{Bernoulli}(\pi_{nk}), k = 1, .., K$$

$$q_\phi(r_n) = \mathcal{N}(\mu_n, \mathrm{diag}(\sigma_n^2))$$

$$\{c_k, d_k, \pi_k, \mu_k, \sigma_k\}_{n=1}^{n=N} = GCN(A, X).$$

使用**随机梯度变分贝叶斯（SGVB）**推断参数。

补充

- 对于节点嵌入 $z_n = b_n \odot r_n$：
  - 忽略 r_n， $z_n = b_n$，变为 OSBM/LFRM
  - 忽略 b_n， $z_n = r_n$，无法推断 K，变为潜空间模型或其非线性扩展 VGAE
- 研究在学术引用网络、生物网络等真实数据上进行实验，结果证实了深度生成模型的有效性。
- 研究基于 OSBM 网络，但 SGVB 算法支持更多类型的网络：**WSBM**，**DC-SBM** 等。

https://github.com/nikhil-dce/SBM-meet-GNN

inference

We define the factorized variational posterior $q_\phi(\mathbf{v}, \mathbf{b}, \mathbf{r})$ as:

$$q_\phi(v_{nk}) = \text{Beta}(v_{nk}|c_k(\mathbf{A}, \mathbf{X}), d_k(\mathbf{A}, \mathbf{X}))$$

$$q_\phi(b_{nk}) = \text{Bernoulli}(b_{nk}|\pi_k(\mathbf{A}, \mathbf{X}))$$

$$q_\phi(r_n) = \mathcal{N}(\mu_n(\mathbf{A}, \mathbf{X}), \text{diag}(\sigma_n^2(\mathbf{A}, \mathbf{X})))$$

where c_k, d_k, π_k, μ_n, and σ_n are functions of the GCN encoder, with inputs $\mathbf{A}$ and $\mathbf{X}$.

inference

We define the loss function $\mathcal{L}$ parameterized by **inference network (encoder) parameters** (ϕ) and **generator parameters** (θ) by minimizing the **negative of the evidence lower bound (ELBO)**:

$$
\begin{aligned}
\mathcal{L} = \sum_{n=1}^{N} [\mathrm{KL}[q_\phi(b_n|v_n)||p_\theta(b_n|v_n)] + \mathrm{KL}[q_\phi(r_n)||p_\theta(r_n)]] \\
+\mathrm{KL}[q_\phi(v_n)||p(v_n)] - \sum_{n=1}^{N} \mathbb{E}_q[\log p_\theta(X_n|z_n)] \\
-\sum_{n=1}^{N}\sum_{m=1}^{N} (\mathbb{E}_q[\log p_\theta(A_{nm}|z_n, z_m)])
\end{aligned}
$$

where $\mathrm{KL}[q(\cdot)||p(\cdot)]$ is the Kullback-Leibler divergence between $q(\cdot)$ and $p(\cdot)$.
Note that here we have also included the loss from the reconstruction of the side information X_n.