Sparse Poisson Gamma Belief Networks for High-Dimensional Sparse Count Data

Rui Huang[1, 2], **Dian Meng**[3], **Xun Zhou**[1,4,5*], **Sikun Yang**[2,6,7,8*]
[1]Harbin Institute of Technology, Shenzhen
[2]School of Computing and Information Technology, Great Bay University, China
[3]Department of Biomedical Informatics, Yong Loo Lin School of Medicine, National University of Singapore
[4]Pengcheng Laboratory
[5]Shenzhen Loop Area Institute
[6]Great Bay Institute for Advanced Study, Great Bay University, China
[7]Guangdong Provincial Key Laboratory of Mathematical and Neural Dynamical Systems
[8]Dongguan Key Laboratory for Data Science and Intelligent Medicine, China
24b951056@stu.hit.edu.cn, dmeng@u.nus.edu, zhouxun2023@hit.edu.cn, sikunyang@gbu.edu.cn

Abstract

Bayesian networks play a crucial role in various domains for unsupervised feature extraction and data interpretation. The Poisson gamma belief networks (PGBNs), as a type of Bayesian networks, have shown promise in analyzing high-dimensional count data. However, PGBNs encounter significant challenges when applied to sparse data, particularly in achieving accurate feature extraction and avoiding overfitting during missing value prediction. In this paper, we propose the sparse Poisson gamma belief networks (SPGBNs), a Bayesian network model designed to address these limitations. By incorporating *sparse graph-structured priors* over the weight matrices between adjacent layers, the proposed SPGBNs effectively capture the inherent sparsity and graph structures of latent features. Meanwhile, SPGBNs demonstrate superior generalization on missing data prediction and enable more stable extraction of meaningful latent features compared to existing approaches. Additionally, we develop an efficient Gibbs sampling algorithm that significantly improves training stability and computational efficiency of SPGBN. Extensive experiments on real-world datasets are conducted to validate the effectiveness of our approach.

Code — https://github.com/HuangRuiiii/SPGBN

Introduction

Bayesian networks, with their inherent Bayesian foundation, transcend traditional matrix factorization by offering enhanced uncertainty quantification, intricate dependency modeling, and robust performance. The latent features extracted by Bayesian networks are critical to predict, visualize, denoise and explain patterns of interest of the data. Thus, Bayesian network models are widely used in many domains such as text mining (Blei and Lafferty 2006; Wang, Blei, and Heckerman 2008; Rudolph and Blei 2018; Acharya, Ghosh, and Zhou 2018; Dieng, Ruiz, and Blei 2019), cell genomic analysis (Levitin et al. 2019; Tong et al. 2020; Jones et al. 2023), population movement forecasting (Sheldon et al. 2013; Stuart and Wolfram 2020; Roy and Dunson 2020), and etc. The Poisson gamma belief networks (PGBNs) (Zhou, Cong, and Chen 2015) are proposed as a Bayesian network model for inferring multilayer representations of high-dimensional count vectors. Distinct from conventional deep networks (Saul, Jaakkola, and Jordan 1996; Hinton, Osindero, and Teh 2006; Salakhutdinov and Larochelle 2010) that often utilize binary units for tractable inference and require tuning both the width (number of hidden units) of each layer and the network depth (number of layers), the PGBNs employ nonnegative real hidden units and automatically infer the widths of subsequent layers given a fixed budget on the width of their first layers. Benefiting from such advantages, the PGBNs are widely used in text analysis (Wang et al. 2019), radar target recognition (Guo et al. 2020), recommender systems (Wang et al. 2022), and etc.

Despite these advantages, the PGBNs series models still encounter significant challenges in extracting latent features from high-dimensional sparse count data which is prevalent in text mining (Puurula 2016), single-cell genomic analysis (Sun et al. 2020), social network systems (Chen, Kato, and Leng 2021), and recommendation systems (Gunathilaka et al. 2025). Specifically, the latent features underlying these sparse data often exhibit sparsity themselves. The previous Bayesian network approaches often overlook the inherent sparsity of latent features. These methods typically employ densely connected architectures for feature extraction from sparse data, creating a fundamental incompatibility between model structures and data characteristics. This architectural mismatch leads to dual challenges: inaccurate feature extraction due to improper representation of sparse patterns, and poor generalization performance for missing value prediction caused by parameter overfitting. Additionally, although PGBNs can automatically prune the number of nodes at each layer, thereby adaptively adjusting the network structure, the presence of redundant parameters makes this pruning process inherently slow. As a result, both model training and inference become computationally inefficient.

To better align the model structure with the underlying data and to enable more effective extraction of latent features from sparse observations, we propose the Sparse Poisson Gamma Belief Networks (SPGBNs). Fig. 1 pro-

*Corresponding author

vides a comparative illustration of the network architectures for a single-layer PGBN and SPGBN. In these diagrams, $(x_1, x_2, \ldots, x_6)$ is observed count vector, while $(\theta_1^{(1)}, \theta_2^{(1)}, \theta_3^{(1)}, \theta_4^{(1)})$ corresponds to the first-layer latent features inferred from the data. The connections between $(x_1, x_2, \ldots, x_6)$ and $(\theta_1^{(1)}, \theta_2^{(1)}, \theta_3^{(1)}, \theta_4^{(1)})$ are determined by the factor loading matrix $\Phi^{(1)}$. In the original PGBN, this matrix is dense, meaning that most edges are preserved, even if their associated weights are negligible. In contrast, SPGBN applies a binary mask to the factor loading matrix, setting most elements to zero and thus removing the corresponding edges from the network, which leads to a sparse model structure. Notably, the sparse network structure enables the model to eliminate edges with small weights, thereby facilitating the rapid pruning of redundant latent features. For example, in the dense factor loading matrix $\Phi^{(1)}$ of PGBN in Fig. 1, the elements in the row corresponding to $\theta_4^{(1)}$ are small but nonzero, which leads to the retention of the feature node $\theta_4^{(1)}$ in the model. In contrast, in the structured factor loading matrix $\Phi^{(1)}$ of SPGBN in Fig. 1, the binary mask sets these elements to zero, resulting in the effective removal of the feature node $\theta_4^{(1)}$ during pruning. As demonstrated in Section , this accelerated pruning allows the structure of SPGBN to quickly converge to a stable state.

The main contributions of the paper include: (1) Proposing Sparse Poisson Gamma Belief Networks (SPGBNs), which incorporate structured sparsity into the factor loading matrices to better match the model architecture with the inherent sparsity of latent features in sparse data; (2) Introducing a binary masking strategy that removes insignificant edges, enabling efficient pruning of redundant latent features and resulting in a more compact network structure whose architecture rapidly stabilizes; (3) Developing an efficient Gibbs sampling algorithm tailored for SPGBNs, which significantly enhances training stability and computational efficiency; (4) Conducting comprehensive experiments that demonstrate SPGBNs not only outperform traditional dense PGBNs in extracting meaningful latent features from sparse data, but also exhibit superior generalization in missing data prediction task.

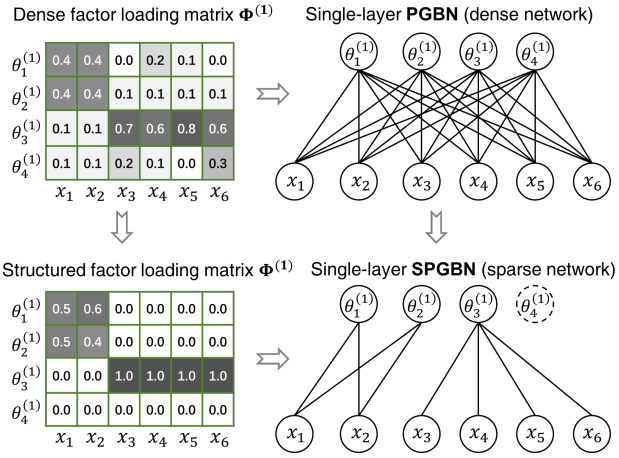

Figure 1: Comparison of the network structures between an original single-layer Poisson Gamma Belief Network (PGBN) and the proposed single-layer Sparse PGBN (SPGBN). In the PGBN, the dense factor loading matrix $\Phi^{(1)}$ preserves most edges, even those with small weights, resulting in a dense network structure. In contrast, SPGBN employs a structured factor loading matrix $\Phi^{(1)}$ in which the negligible values are set to zero, efficiently removing insignificant edges and facilitating rapid pruning of redundant feature $\theta_4^{(1)}$.

Background

Notations

In what we present below, vectors are denoted by boldfaced lowercase letters, and matrices are denoted by boldfaced capital letters. $\mathrm{Pois}(\cdot), \mathrm{Gam}(\cdot), \mathrm{NB}(\cdot), \mathrm{Dir}(\cdot), \mathrm{Ber}(\cdot)$, and $\mathrm{Mult}(\cdot)$ stand for the Poisson, gamma, negative binomial, Dirichlet, Bernoulli, and multinomial distributions, respectively. We let $\odot$ denote the Hadamard product between matrices or vectors.

Poisson Gamma Belief Networks

Suppose we have multivariate count vector $\boldsymbol{x}_j \in \mathbb{Z}^{K_0}$, where $\boldsymbol{x}_j = (x_{1j}, \ldots, x_{K_0 j})$ denotes the jth count vector whose dimension is K_0. The Poisson gamma belief networks (PGBNs) with T hidden layers factorizes the observed vector $\boldsymbol{x}_j$ at the first layer under the Poisson likelihood as

$$\boldsymbol{x}_j \sim \mathrm{Pois}\left(\boldsymbol{\Phi}^{(1)}\boldsymbol{\theta}_j^{(1)}\right), \tag{1}$$

where $\boldsymbol{\Phi}^{(1)} \in \mathbb{R}_+^{K_0 \times K_1}$ is the factor loading matrix and $\boldsymbol{\theta}_j^{(1)} \in \mathbb{R}_+^{K_1}$ is the hidden units vector for layer $t = 1$. Specifically, $\boldsymbol{\Phi}^{(1)}$ is a global parameter that remains constant regardless of the data, while $\boldsymbol{\theta}_j^{(1)}$ is the local parameter tailored for the vector $\boldsymbol{x}_j$. For the first layer $t = 1$, the model factorizes the shape parameters of the gamma distributed hidden units $\boldsymbol{\theta}_j^{(1)} \in \mathbb{R}_+^{K_t}$ of layer $t = 1$ into the product of the connection weight matrix $\boldsymbol{\Phi}^{(2)} \in \mathbb{R}_+^{K_1 \times K_2}$ and the hidden units $\boldsymbol{\theta}_j^{(2)} \in \mathbb{R}_+^{K_2}$ of layer $t = 2$.

$$\boldsymbol{\theta}_j^{(1)} \sim \mathrm{Gam}\left(\boldsymbol{\Phi}^{(2)}\boldsymbol{\theta}_j^{(2)}, \frac{p_j^{(2)}}{1 - p_j^{(2)}}\right), \tag{2}$$

where $p_j^{(2)}$ are probability parameters specified by $p_j^{(2)} \sim \mathrm{Beta}(a_0, b_0)$. For $t = 2, \ldots, T-1$, the hidden units vector $\boldsymbol{\theta}_j^{(t)}$ follows distribution as

$$\boldsymbol{\theta}_j^{(t)} \sim \mathrm{Gam}\left(\boldsymbol{\Phi}^{(t+1)}\boldsymbol{\theta}_j^{(t+1)}, c_j^{(t+1)}\right),$$

where $c_j^{(t+1)}$ are gamma rate parameters specified by $c_j^{(t)} \sim \mathrm{Gam}(e_0, f_0)$. To maintain consistent notation across layers,

the PGBSs define the relation between $p_j^{(t)}$ and $c_j^{(t)}$ by

$$p_j^{(t+1)} := \frac{-\ln\left(1-p_j^{(t)}\right)}{c_j^{(t+1)} - \ln\left(1-p_j^{(t)}\right)}. \tag{3}$$

In particular, for the second layer $t=2$, we define $c_j^{(2)} := (1-p_j^{(2)})/p_j^{(2)}$. For the top layer i.e. $t=T$, the hidden units $\boldsymbol{\theta}_j^{(T)}$ share the same vector $\boldsymbol{r}=(r_1,\cdots,r_{K_T})$ as their gamma shape parameters

$$\boldsymbol{\theta}_j^{(T)} \sim \mathrm{Gam}\left(\boldsymbol{r}, c_j^{(T+1)}\right),$$

where each element of vector $\boldsymbol{r}$ is characterized by a gamma distribution as $r_k \sim \mathrm{Gam}\left(\gamma_0/K_T, c_0\right)$). The hyperparameters γ_0 and c_0 are randomized by gamma distribution as $\gamma_0 \sim \mathrm{Gam}\left(a_0, b_0\right)$ and $c_0 \sim \mathrm{Gam}\left(e_0, f_0\right)$, respectively.

The Proposed Model

Sparse Poisson Gamma Belief Networks

In this section, we introduce the sparse Poisson gamma belief network, a novel structure designed to extract latent features from sparse count data and infer multilayer representations of high-dimensional count vectors. The model factorizes observed count vectors $\boldsymbol{x}_j \in \mathbb{Z}^{K_0}$ by decomposing them into the product of the factor loading matrix $\boldsymbol{\Phi}^{(1)} \in \mathbb{R}_+^{K_0\times K_1}$ and the hidden units factor $\boldsymbol{\theta}^{(1)} \in \mathbb{R}_+^{K_1}$, following the approach defined in Eq. 1. Specifically, the factor loading matrix $\boldsymbol{\Phi}^{(1)} = \left[\boldsymbol{\phi}_1^{(1)}, \boldsymbol{\phi}_2^{(1)}, \cdots, \boldsymbol{\phi}_{K_1}^{(1)}\right]$, where each element $\boldsymbol{\phi}_{k_1}^{(1)}$ represents a column vector of length K_0. For scale identifiability and ease of inference, each column of $\boldsymbol{\Phi}^{(1)}$ is restricted to have a unit L_1 norm. Thus the Dirichlet prior is placed over the k_1-th column of $\boldsymbol{\Phi}^{(1)}$ as

$$\boldsymbol{\phi}_{k_1}^{(1)} \sim \mathrm{Dir}\left(a_{1k_1}^{(1)}, a_{2k_1}^{(1)}, \cdots, a_{K_0k_1}^{(1)}\right),$$

where $\left[a_{1k_1}^{(1)}, a_{2k_1}^{(1)}, \cdots, a_{K_0k_1}^{(1)}\right]^{\mathrm{T}}$ is the k_1-th column of the hyperparameter matrix $\mathbf{A}^{(1)} \in \mathbb{R}_+^{K_0\times K_1}$. To introduce a *sparse graph-structured factor loading matrix*, we model the hyperparameter matrix $\mathbf{A}^{(1)} = [a_{k_0k_1}^{(1)}]_{k_0,k_1}^{K_0,K_1}$ as

$$\mathbf{A}^{(1)} = \mathbf{Z}^{(1)} \odot \mathbf{D}^{(1)},$$

where $\mathbf{Z}^{(1)} = [z_{k_0k_1}^{(1)}]_{k_0,k_1}^{K_0,K_1}$ serves as a binary mask to define the graph structure, and $\mathbf{D}^{(1)} = [d_{k_0k_1}^{(1)}]_{k_0,k_1}^{K_0,K_t}$ represents the weight relationships between the observed vectors and the hidden units of the first layer. Specifically, the weight $d_{k_0k_1}^{(1)} \in \mathbb{R}^+$ indicates the strength of the connection between the k_0-th observed count value x_{k_0j} and the k_1-th hidden unit $\theta_{k_1j}^{(1)}$ in the first layer, drawn from $d_{k_0k_1}^{(1)} \sim \mathrm{Gam}\left(\epsilon_0, \epsilon_0\right)$. The graph structure parameter $z_{k_0k_1}^{(1)} \in \{0,1\}$ is a binary variable indicating the presence of a link between the k_0-th observed count value and the k_1-th hidden unit in the first layer. The binary variable $z_{k_0k_1}^{(1)}$ is modeled using a hierarchical gamma process edge partition model (Zhou 2015; Yang and Koeppl 2018a,b; Huang, Yang, and Koeppl 2024) as

$$z_{k_0k_1}^{(1)} \sim \mathrm{Ber}\left(1-\exp\left(\sum_{l_1=1}^{L_1} u_{k_0l_1}^{(1)} v_{k_1l_1}^{(1)} \lambda_{l_1}^{(1)}\right)\right). \tag{4}$$

As shown in Eq. 4, the graph structure parameter $z_{k_0k_1}^{(1)}$ is further decomposed into L_1 latent groups. More specifically, $u_{k_0l_1}^{(1)}$ captures how strongly the k_0-th observed count value associates with the l_1-th group, and $v_{k_1l_1}^{(1)}$ captures how strongly the k_1-th hidden unit of the first layer associates with the l_1-th group. $\lambda_{l_1}^{(1)}$ is the weight of the l_1-th group between the observed count vector and the first hidden unit layer. Note that $z_{k_0k_1}^{(1)}$ can be equivalently drawn via the Bernoulli-Poisson link function as

$$\begin{aligned} z_{k_0k_1}^{(1)} &= \mathbf{1}\left(m_{k_0k_1}^{(1)} \geq 1\right), \\ m_{k_0k_1}^{(1)} &\sim \mathrm{Pois}\left(\sum_{l_1=1}^{L_1} u_{k_0l_1}^{(1)} v_{k_1l_1}^{(1)} \lambda_{l_1}^{(1)}\right), \end{aligned}$$

where $\mathbf{1}(\cdot)$ is an indication function. To ensure the model tractability, we restrict $u_{k_0l_1}^{(1)}$, $v_{k_1l_1}^{(1)}$ and $\lambda_{l_1}^{(1)}$ to be nonnegative, and thus place gamma priors over these parameters as

$$\begin{aligned} u_{k_0l_1}^{(1)} &\sim \mathrm{Gam}\left(a_0, b_0\right), \\ v_{k_1l_1}^{(1)} &\sim \mathrm{Gam}\left(e_0, f_0\right), \\ \lambda_{l_1}^{(1)} &\sim \mathrm{Gam}\left(\alpha_0/L_1, \beta_0\right), \end{aligned}$$

where $a_0, b_0, e_0, f_0, \alpha_0$ and β_0 are hyperparameters. The proposed first layer structure extend hierarchically to T layers through recursive decomposition. For layer $t(2 \leq t \leq T)$, the hidden units $\boldsymbol{\theta}^{(t-1)}$ extracted from the previous layers are similarly factorized as

$$\begin{aligned} &\boldsymbol{\theta}_j^{(1)} \sim \mathrm{Gam}\left(\boldsymbol{\Phi}^{(2)}\boldsymbol{\theta}_j^{(2)}, \frac{p_j^{(2)}}{1-p_j^{(2)}}\right), \\ &\cdots, \\ &\boldsymbol{\theta}_j^{(t)} \sim \mathrm{Gam}\left(\boldsymbol{\Phi}^{(t+1)}\boldsymbol{\theta}_j^{(t+1)}, c_j^{(t+1)}\right), \\ &\cdots, \\ &\boldsymbol{\theta}_j^{(T)} \sim \mathrm{Gam}\left(\boldsymbol{r}, c_j^{(T+1)}\right), \end{aligned}$$

where $\boldsymbol{\Phi}^{(t)} \in \mathbb{R}_+^{K_{t-1}\times K_t}$, $\boldsymbol{\theta}^{(t)} \in \mathbb{R}_+^{K_t}$, for $t=2,3,,\cdots,T$. The $p_j^{(2)}$ are probability parameters, $\{c_j^{(t)}\}_{t=3}^{T+1}$ are gamma rate parameters, and the relation between $c_j^{(t)}$ and $p_j^{(t)}$ is the same as in Eq. 3. Especially, the top layer hidden units $\boldsymbol{\theta}_j^{(T)}$ share the same vector $\boldsymbol{r}=(r_1,\cdots,r_{K_T})$ as their gamma shape parameters. The parameters $p_j^{(2)}, c_j^{(t)}$ and r_k can be

Algorithm 1: The SPGBNs Gibbs sampler

Input: Observed count vector $\boldsymbol{x}$. The number of iterations $B_t + C_T$, where B_T is burn-in time and C_T is collection time. The upper bound of the number of layers $T_{\max}$ and the fixed budget on the width of the first layer K_0. The number of latent groups L_t in each layer, and hyperparameters.
Output: A total of $T_{\max}$ jointly trained SPGBNs.

for $T = 1, 2, \ldots, T_{\max}$ **do**
  Initialize Parameters;
  for $iter = 1 : B_T + C_T$ **do**
    for $t = 1, 2, \ldots, T$ **do**
      Sample latent counts $\{x^{(t)}_{k_{t-1}jk_t}\}_{k_{t-1},k_t,j}$;
      Sample $\{u^{(t)}_{k_{t-1}l_t}\}_{k_{t-1},l_t}$,$\{v^{(t)}_{k_tl_t}\}_{k_t,l_t}$, $\{\lambda^{(t)}_{l_t}\}_{l_t}$;
      Sample $\boldsymbol{M}^{(t)}$, $\boldsymbol{Z}^{(t)}$, $\boldsymbol{D}^{(t)}$, and calculate $\boldsymbol{A}^{(t)}$;
      Sample $\boldsymbol{\Phi}^{(t)}$;
    end for
    Sample $\{p^{(2)}_j\}_j$ and calculate $\{c^{(2)}_j\}_j$;
    Sample $\{c^{(t)}_j\}_{j,t}$ and calculate $\{p^{(t)}_j\}_{j,t}$ for $t = 3, \ldots, T+1$;
    for $t = T, T-1, \ldots, 2$ **do**
      Sample $\{r_{k_T}\}_{k_T}$ if $t = T$;
      Sample $\{\boldsymbol{\theta}^{(t)}_j\}_j$;
    end for
    if $iter = B_T$ **then**
      Prune inactive hidden units k_T where $x^{(T)}_{\cdot\cdot k_T} = 0$;
    end if
  end for
  Calculate and output the posterior means of $\{\boldsymbol{\Phi}^{(t)}\}_t$ and $\{r_{k_T}\}_{k_T}$ during collection.
end for

drawn as

$$p^{(2)}_j \sim \mathrm{Beta}\left(a_0, b_0\right),$$
$$c^{(t)}_j \sim \mathrm{Gam}\left(e_0, f_0\right),$$
$$r_k \sim \mathrm{Gam}\left(\frac{\gamma_0}{K_T}, c_0\right),$$

where the hyperparameters γ_0 and c_0 are randomized by the gamma distribution as $\gamma_0 \sim \mathrm{Gam}\left(a_0, b_0\right)$ and $c_0 \sim \mathrm{Gam}\left(e_0, f_0\right)$, respectively. The factor loading matrix $\boldsymbol{\Phi}^{(t)}$ is constructed and sampled using the same hierarchical structure as the first layer, and thus omitted here for brevity.

The proposed sparse Poisson gamma belief networks are not fully conjugate. Nonetheless, tractable-yet-efficient Gibbs sampling algorithms are developed to perform posterior simulation. The full derivation of the inference procedure is presented in Appendix B.

The Layer-Wise Training Strategy

To learn the width of each hidden layer instead of fixing the number of hidden units, the original PGBNs developed a greedy layer-wise training strategy. The proposed SPGBNs inherit the training methodology of PGBNs. Under a fixed budget on the width of the first layer, SPGBNs retain the advantages of the nonparametric Bayesian shrinkage mechanism, enabling effective pruning of inactive hidden units at each layer. For more details on the nonparametric Bayesian shrinkage mechanism, readers are referred to (Zhou, Cong, and Chen 2015), the discussion is omitted here for brevity. Notably, thanks to the model inherent sparsity, SPGBNs can prune redundant hidden units more efficiently, resulting in more stable training and accelerated inference. Experimental results supporting these claims are presented in the experimental section. The detailed algorithm is summarized in the Algorithm 1.

Model	Classification Accuracy		
	MR	TREC	SUBJ
LDA	54.4±0.8	45.5±1.9	68.2±1.3
DocNADE	54.2±0.8	62.0±0.6	72.9±1.2
DPFA	56.1±0.9	62.0±0.6	78.5±1.4
PGBN	57.0±0.5	67.9±1.5	78.3±1.2
WHAI	56.4±0.6	65.6±1.7	76.5±1.1
CPGBN	63.5±0.8	**74.4±0.6**	81.5±0.6
SPGBN	**65.0±0.8**	72.1±1.2	**85.3±0.4**

Table 1: Comparison of classification accuracy on feature vectors extracted by unsupervised methods.

Experiments

To evaluate the effectiveness of the proposed SPGBNs, we conducted experiments involving quantitative comparisons and qualitative analysis across various datasets.

20Newsgroups: Comprises $18,774$ documents from 20 different groups, with a vocabulary of $33,420$ meaningful words. The resulting matrix is highly sparse, containing 99.7% zeros and only 0.3% non-zero elements.

SPL111: Single cell RNA sequencing data (Gayoso et al. 2021) measuring expression levels of $13,533$ genes in $16,828$ cells. After filtering out classes with fewer than 500 cells, $12,137$ cells across 10 classes remain.The filtered expression matrix remains 89.9% zeros.

MR: Movie reviews dataset (Pang and Lee 2005a) comprises approximately $10,600$ reviews labeled positive or negative, with a vocabulary of $20,000$ words. The resulting matrix contains 90.6% zeros.

TREC: TREC question dataset (Pang and Lee 2005b) consisting of $5,952$ questions categorized into six types (abbreviation, entity, description, human, location, or numeric) with a vocabulary of 8000 words. The sparse count matrix contains about 88.3% zeros.

SUBJ: The Subjectivity dataset (Pang and Lee 2004) contains approximately $10,000$ sentences labeled as subjective or objective opinions on movies, with a vocabulary of $22,636$ words. The matrix is roughly 83.7% sparse.

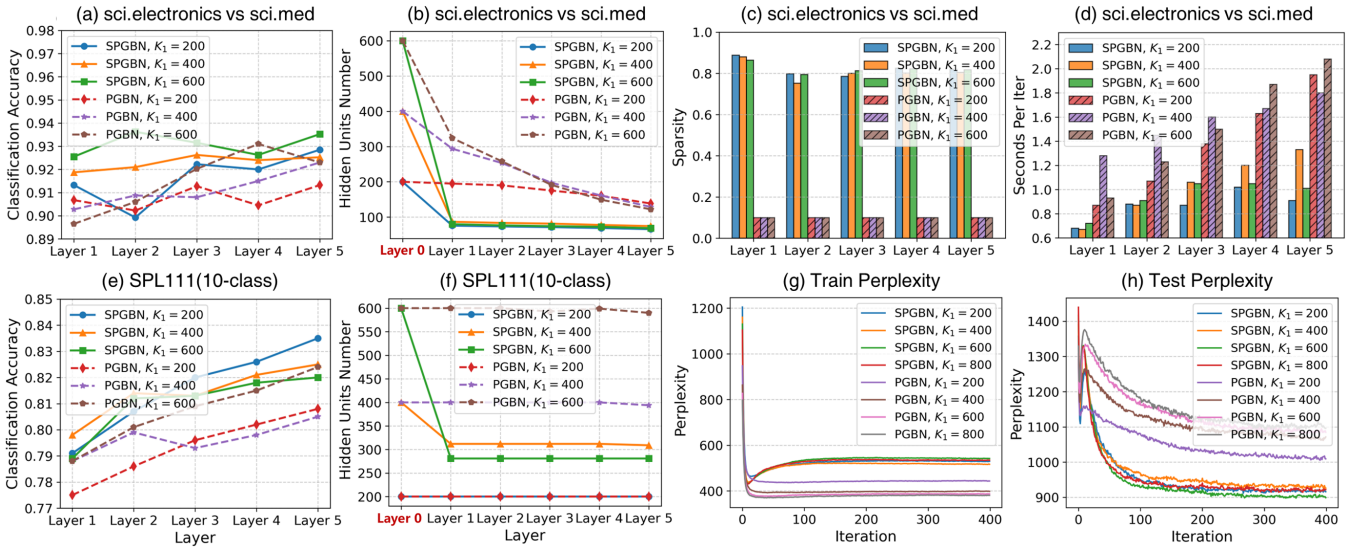

Figure 2: The proposed SPGBNs achieve comparable or superior classification accuracy to the original PGBNs for both the binary classification task (a) and 10-class classification task (e). Additionally, the proposed SPGBNs rapidly prune redundant hidden nodes, causing the number of hidden nodes to quickly stabilize at approximately 80 for the binary classification dataset (b) and 350 for the SPL111 dataset (f). Furthermore, the sparsity of the factor loading matrix $\mathbf{\Phi}^{(t)}$ in each layer reaches nearly 80% (c), significantly accelerating inference speed compared to the original PGBNs (d). Finally, figures (g) and (f) demonstrate that the perplexity value decreases as the number of iterations increases for both training set and test set, respectively.

Quantitative Analysis

We systematically evaluated the performance of unsupervised feature learning and missing value prediction. General classification accuracy evaluation were conducted over MR, TREC and SUBJ datasets. For comparison, we considered classical LDA (Blei, Ng, and Jordan 2003), several related Bayesian models including DPFA (Gan et al. 2015), PGBN (Zhou, Cong, and Chen 2015), WHAI (Zhang et al. 2018), CPGBN (Wang et al. 2019) as well as the neural topic model DocNADE (Lauly et al. 2017), as shown in Table 1. Especially, we performed an in-depth comparison between SPGBNs and PGBNs on the 20Newsgroups and SPL111 datasets, focusing on feature extraction, missing value prediction, training stability and computational efficiency, as illustrated in Fig. 2.

Unsupervised feature learning. To evaluate the quality of the features learned by the proposed unsupervised SPGBNs, we performed downstream classification tasks using the latent features from various unsupervised models. For fair comparison, we adopted the same training and test splits as (Wang et al. 2019) for the MR, TREC and SUBJ datasets. The results are shown in Table 1, where the means and error bars were obtained from five independent runs. For LDA, DocNADE, DPFA, PGBN, WHAI and CPGBN, we report the best results from (Wang et al. 2019). The proposed SPGBNs were configured with three layers and initial $K_{1\max} = 200$, consistent with the PGBN setting. The proposed SPGBNs significantly outperformed the other models on the MR and SUBJ datasets, but performed worse than CPGBN on the TREC dataset. We hypothesized that this was due to the relatively small number of documents and the imbalanced class distribution in TREC, which made it difficult for SPGBNs equipped with a sparse structure to extract sufficiently effective features.

We systematically compared the binary, 10-class and 20-class classification tasks between the proposed SPGBNs and the original PGBNs, as shown in Fig. 2(a), Fig. 2(e) and Appendix D, respectively. For the binary classification, we distinguished between the $sci.electronics$ and $sci.med$ newsgroups ($1,971$ out of $18,774$ documents in 20Newsgroups). Each dataset was partitioned into 70% training and 30% test data. The first layer width for both SPGBNs and PGBNs was set to $K_{1\max} \in \{200, 400, 600\}$. We see that the accuracy across all six experimental groups on three datasets exhibits a fluctuating upward trend as the number of model layers increases. Meanwhile, the proposed SPGBNs consistently outperformed the original PGBNs across most configurations.

Especially, as shown in Fig. 2(b), the proposed SPGBNs rapidly pruned redundant hidden nodes, and consistently stabilized at approximately 80 hidden nodes across all 5 layers, regardless of the fixed budget on the width of the layer. For the SPL111 dataset, the 200 initial hidden nodes were insufficient to capture all the features of the dataset, which was why neither the SPGBNs nor PGBNs pruned the initial 200 nodes. However, when the number of initial nodes increased to 400 and 600, the SPGBNs quickly pruned them to around 300 nodes. This indicated that the inference results of SPGBNs were primarily driven by the data itself rather than by predefined model structures. In contrast, the inference results of PGBNs were influenced by both the fixed budget in the first layer and the network depth. We specu-

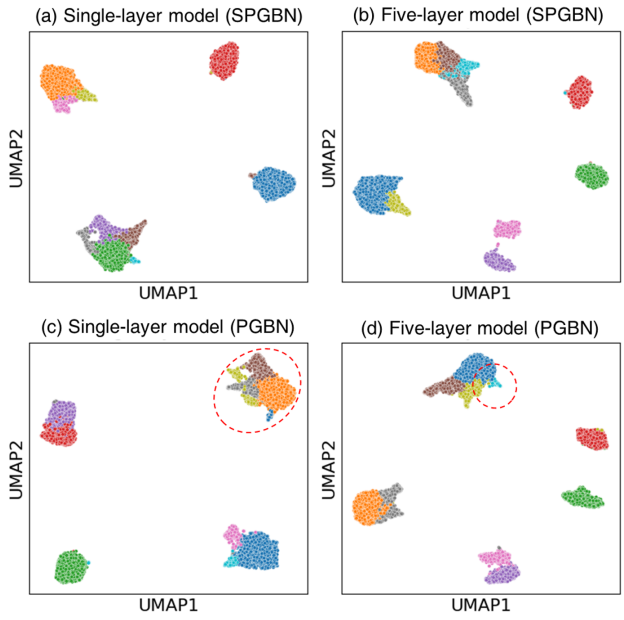

Figure 3: Using the 200-dimensional features extracted from the **SPL111** dataset, 12,137 cell samples were automatically clustered into 10 groups.

lated that this effectiveness was due to the sparse network structures of SPGBNs, which masked out much of the connectivity between adjacent layers, allowing the model to efficiently inferred important node features. Additionally, we also compared the sparsity of the global structure parameter matrix $\mathbf{\Phi}^{(t)}$ for each layer in Fig. 2(c), where higher values indicated greater sparsity. For clarity, sparsity values below 0.1 were clipped to 0.1. Due to the sparse prior, SPGBNs activated only about 20% of the parameters per layer, enabling SPGBNs consumed considerably less time per inference iteration at each layer than PGBNs, as shown in Fig. 2(d). Detailed model configurations, experimental setups, and supplementary comparative results are provided in the Appendices C and D.

Perplexities for hold out words. In addition to evaluating the performance of the proposed SPGBNs for unsupervised feature learning, we also considered a prediction task on 20Newsgroups. Specifically, we randomly selected 30% of the word tokens in each document for training and used the remaining 70% to compute per-heldout-word perplexity. The perplexity measures the uncertainty of the predictions, where lower values signify better performance, and is calculated according (Zhou et al. 2012; Zhou and Carin 2013; Zhou, Cong, and Chen 2015) as follows

$$\text{Perplexity} = \exp\left(-\frac{\sum_{i,j} n_{ij} \log f_{ij}}{\sum_{i,j} n_{ij}}\right),$$

where $f_{ij} = \frac{\phi_{ij}^{(1)}\theta_j^{(1)}}{\sum_i \phi_{ij}^{(1)}\theta_j^{(1)}}$, n_{ij} is the number of words held out at word term i in document j. All 20 topic categories were retained, with the vocabulary restricted to the $2,000$ most frequent terms. Perplexity was recorded every 5 iterations over $2,000$ iterations. Fig. 2(g) and (h) present the perplexity trajectories on the training and test set, respectively. On the training set, all models exhibited a sharp decrease in perplexity during the first 20 collections, reaching approximately 400. Subsequently, the perplexity of SPGBNs gradually increased to around 500 and then stabilized, whereas PGBNs remained largely unchanged following the initial decline—suggesting that PGBNs continued to fit the training data closely. However, on the test set, SPGBNs achieved a substantially greater reduction in perplexity compared to PGBNs. This result demonstrated that, while PGBNs performed better on the training set, they tended to overfit and thus underperformed on unseen data. In contrast, the sparsity-inducing prior in SPGBNs promoted better generalization and robustness, enabling the model to achieve lower perplexity on the test set.

Qualitative Analysis

To further assessed the feature quality extracted by SPGBNs and verified effective use of the designed prior for data structure capture, we visualized clustering results on latent features from the SPL111 and the latent graph structures inferred by SPGBNs.

Clustering for extracted latent features. Unsupervised learning models were commonly used in single-cell transcriptomics to extract cell representations for downstream analyses such as clustering (Meng et al. 2025). In our study, the $12,137$ cells were clustered based on the latent features using the Leiden algorithm (Traag, Waltman, and Van Eck 2019), and both cell embeddings and latent features were visualized in 2D with UMAP. Specifically, Fig. 3(a) (b) display SPGBN clustering results for single-layer and five-layer models, while Fig. 3(c) (d) show PGBN results under the same setting. All models successfully separated the cells into 10 distinct classes. The original single-layer PGBN exhibited significant feature coupling and poor disentanglement, while the five-layer PGBN improved disentanglement but still underrepresents some cell types. In contrast, SPGBNs achieved effective disentanglement from the first layer and yielded a more balanced, accurate feature distribution by the fifth layer.

Visualization for structural parameters. Fig. 4 illustrates the latent graph structures underlying the factor loading matrics $\mathbf{\Phi}$ between adjacent layers estimated by the proposed SPGBNs. Fig. 4(a)(b) and (c) correspond to the 2nd, 3rd and 4th layers, respectively. The 10 subplots on the left of each layer represent latent groups with dense connections between adjacent layers, while the rightmost subplot in each layer shows that most latent groups are nearly independent from each other. Deeper layers reveal increasingly distinct and well-separated latent group structures.

Visualization for network structure and topic relationship. To further compare the network structures of the original PGBN and the proposed SPGBN, and to analyze the relationships between nodes across layers, we visual-

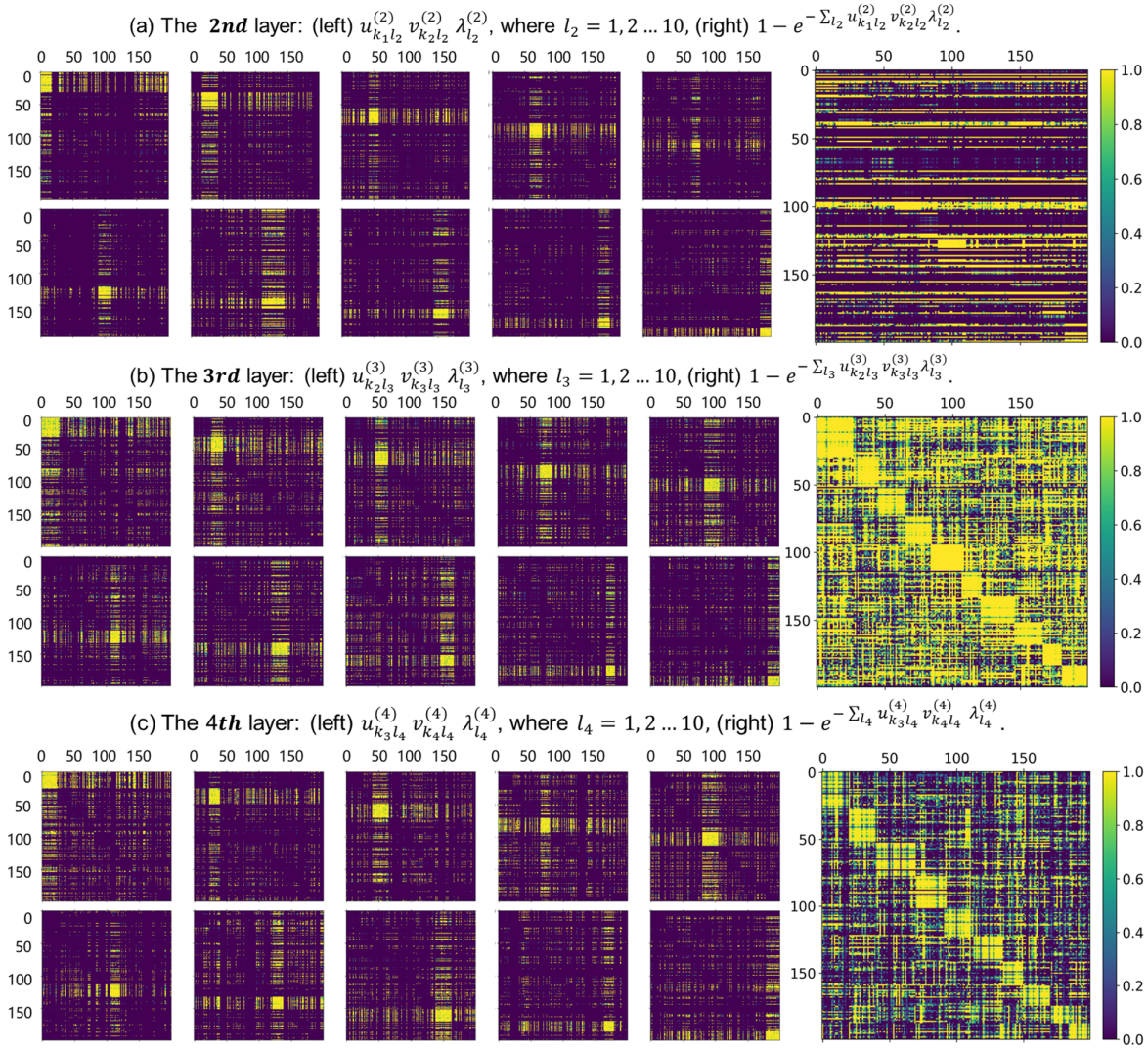

Figure 4: The latent graph structures of layer 2 (a), layer 3 (b) and layer 4 (b) inferred by the proposed SPGBN on **SPL111** dataset. The left subplots illustrate the single latent group identified in each layer, while the right plot displays the overall graph structure of each layer.

ized their network structures and topic trees on the 20Newsgroups dataset. Detailed experimental procedures and results are provided in the Appendix E.

Conclusion

In this work, we proposed the sparse Poisson gamma belief networks (SPGBNs), an advanced Bayesian network model that extended the original Poisson gamma belief networks (PGBNs). The proposed SPGBNs specifically addressed the limitations of existing approaches in accurate feature extraction from high-dimensional sparse count data and overfitting issues in missing value prediction. By integrating adaptive sparse priors and a structured graph-based mask within the network, SPGBNs effectively captured the inherent sparsity and latent structures of real-world data, enabling more accurate and interpretable feature extraction. An efficient Gibbs sampling algorithm was developed for SPGBNs, which significantly enhanced training stability and computational efficiency. Extensive experiments showed SPGBNs achieved comparable or superior classification performance to PGBNs, with significant improvements in sparsity modeling, structural stability, and generalization. Importantly, the sparsity mechanism allowed SPGBNs to rapidly prune redundant hidden units, leading to faster inference and more robust latent representations. Overall, SPGBNs provided a principled and practical approach for learning deep sparse representations from complex count data, establishing a powerful and interpretable Bayesian network model.

Acknowledgments

This work was supported by National Natural Science Foundation of China (NSFC) (Grant No.62476047), Peking Uni-

versity Mathematics Challenge Funding Program (Grant No.2024SRMC10), Guangdong Research Team for Communication and Sensing Integrated with Intelligent Computing (Project No.2024KCXTD047), the Guangdong Provincial Key Laboratory of Mathematical and Neural Dynamical Systems (Grant No.2024B1212010004), the Cross Disciplinary Research Team on Data Science and Intelligent Medicine(Grant No. 2023KCXTD054), National Natural Science Foundation of China (No. 62472125), the Natural Science Foundation of Guangdong Province, China (No. 2025A1515011258), and Shenzhen Science and Technology Programs (No. GXWD20231128102922001, ZDSYS20230626091203008).

References

Acharya, A.; Ghosh, J.; and Zhou, M. 2018. A Dual Markov Chain Topic Model for Dynamic Environments. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 1099–1108.

Blei, D. M.; and Lafferty, J. D. 2006. Dynamic topic models. In *Proceedings of the 23rd international conference on Machine learning*, 113–120.

Blei, D. M.; Ng, A. Y.; and Jordan, M. I. 2003. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan): 993–1022.

Chen, M.; Kato, K.; and Leng, C. 2021. Analysis of Networks via the Sparse -model. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 83(5): 887–910.

Dieng, A. B.; Ruiz, F. J. R.; and Blei, D. M. 2019. The Dynamic Embedded Topic Model. *ArXiv*.

Gan, Z.; Chen, C.; Henao, R.; Carlson, D.; and Carin, L. 2015. Scalable deep Poisson factor analysis for topic modeling. In *International Conference on Machine Learning*, 1823–1832. PMLR.

Gayoso, A.; Steier, Z.; Lopez, R.; Regier, J.; Nazor, K. L.; Streets, A.; and Yosef, N. 2021. Joint probabilistic modeling of single-cell multi-omic data with totalVI. *Nature methods*, 18(3): 272–282.

Gunathilaka, T. M. A. U.; Manage, P. D.; Zhang, J.; Li, Y.; and Kelly, W. 2025. Addressing sparse data challenges in recommendation systems: A systematic review of rating estimation using sparse rating data and profile enrichment techniques. *Intelligent Systems with Applications*, 25: 200474.

Guo, D.; Chen, B.; Chen, W.; Wang, C.; Liu, H.; and Zhou, M. 2020. Variational Temporal Deep Generative Model for Radar HRRP Target Recognition. *IEEE Transactions on Signal Processing*, 68: 5795–5809.

Hinton, G. E.; Osindero, S.; and Teh, Y.-W. 2006. A fast learning algorithm for deep belief nets. *Neural Comput.*, 18(7): 1527–1554.

Huang, R.; Yang, S.; and Koeppl, H. 2024. Negative-binomial randomized gamma dynamical systems for heterogeneous overdispersed count time sequences. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, 4174–4182.

Jones, A.; Townes, F. W.; Li, D.; and Engelhardt, B. E. 2023. Alignment of spatial genomics data using deep Gaussian processes. *Nature Methods*, 20(9): 1379–1387.

Lauly, S.; Zheng, Y.; Allauzen, A.; and Larochelle, H. 2017. Document neural autoregressive distribution estimation. *Journal of Machine Learning Research*, 18(113): 1–24.

Levitin, H. M.; Yuan, J.; Cheng, Y. L.; Ruiz, F. J.; Bush, E. C.; Bruce, J. N.; Canoll, P.; Iavarone, A.; Lasorella, A.; Blei, D. M.; et al. 2019. De novo gene signature identification from single-cell RNA-seq with hierarchical Poisson factorization. *Molecular systems biology*, 15(2): e8557.

Meng, D.; Feng, Y.; Yuan, K.; Yu, Z.; Cao, Q.; Cheng, L.; and Zheng, X. 2025. scMMAE: masked cross-attention network for single-cell multimodal omics fusion to enhance unimodal omics. *Briefings in Bioinformatics*, 26(1): bbaf010.

Pang, B.; and Lee, L. 2004. A sentimental education: sentiment analysis using subjectivity summarization based on minimum cuts. ACL '04, 271–es. Association for Computational Linguistics.

Pang, B.; and Lee, L. 2005a. Seeing stars: exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, 115–124. Association for Computational Linguistics.

Pang, B.; and Lee, L. 2005b. Seeing stars: exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, 115–124. Association for Computational Linguistics.

Puurula, A. 2016. Scalable Text Mining with Sparse Generative Models. arXiv:1602.02332.

Roy, A.; and Dunson, D. B. 2020. Nonparametric graphical model for counts. *The Journal of Machine Learning Research*, 21(1): 9353–9373.

Rudolph, M.; and Blei, D. 2018. Dynamic Embeddings for Language Evolution. In *Proceedings of the 2018 World Wide Web Conference*, 1003–1011.

Salakhutdinov, R.; and Larochelle, H. 2010. Efficient Learning of Deep Boltzmann Machines. In Teh, Y. W.; and Titterington, M., eds., *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, volume 9 of *Proceedings of Machine Learning Research*, 693–700. Chia Laguna Resort, Sardinia, Italy: PMLR.

Saul, L. K.; Jaakkola, T.; and Jordan, M. I. 1996. Mean field theory for sigmoid belief networks. *J. Artif. Int. Res.*, 4(1): 61–76.

Sheldon, D.; Sun, T.; Kumar, A.; and Dietterich, T. 2013. Approximate Inference in Collective Graphical Models. In *Proceedings of the 30th International Conference on Machine Learning*, 1004–1012.

Stuart, A. M.; and Wolfram, M.-T. 2020. Inverse Optimal Transport. *SIAM Journal on Applied Mathematics*, 80(1): 599–619.

Sun, Y.; Sun, Z.; Jiang, Y.; Li, Y.; and Ma, S. 2020. An integrative sparse boosting analysis of cancer genomic commonality and difference. *Statistical methods in medical research*, 29(5): 1325–1337.

Tong, A.; Huang, J.; Wolf, G.; Van Dijk, D.; and Krishnaswamy, S. 2020. TrajectoryNet: A Dynamic Optimal Transport Network for Modeling Cellular Dynamics. In *Proceedings of the 37th International Conference on Machine Learning*.

Traag, V. A.; Waltman, L.; and Van Eck, N. J. 2019. From Louvain to Leiden: guaranteeing well-connected communities. *Scientific reports*, 9(1): 1–12.

Wang, C.; Blei, D. M.; and Heckerman, D. E. 2008. Continuous Time Dynamic Topic Models. In *Conference on Uncertainty in Artificial Intelligence*.

Wang, C.; Chen, B.; Xiao, S.; and Zhou, M. 2019. Convolutional Poisson gamma belief network. In *International conference on machine learning*, 6515–6525. PMLR.

Wang, D.; Wang, C.; Chen, B.; and Zhou, M. 2022. Ordinal Graph Gamma Belief Network for Social Recommender Systems. *CoRR*, abs/2209.05106.

Yang, S.; and Koeppl, H. 2018a. Dependent Relational Gamma Process Models for Longitudinal Networks. In Dy, J.; and Krause, A., eds., *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, 5551–5560.

Yang, S.; and Koeppl, H. 2018b. A poisson gamma probabilistic model for latent node-group memberships in dynamic networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Zhang, H.; Chen, B.; Guo, D.; and Zhou, M. 2018. WHAI: Weibull Hybrid Autoencoding Inference for Deep Topic Modeling. In *6th International Conference on Learning Representations*. OpenReview.net.

Zhou, M. 2015. Infinite Edge Partition Models for Overlapping Community Detection and Link Prediction. In Lebanon, G.; and Vishwanathan, S. V. N., eds., *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics*, volume 38 of *Proceedings of Machine Learning Research*, 1135–1143. PMLR.

Zhou, M.; and Carin, L. 2013. Negative binomial process count and mixture modeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(2): 307–320.

Zhou, M.; Cong, Y.; and Chen, B. 2015. The Poisson gamma belief network. In *Proceedings of the 29th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'15, 3043–3051. Cambridge, MA, USA: MIT Press.

Zhou, M.; Hannah, L.; Dunson, D.; and Carin, L. 2012. Beta-negative binomial process and Poisson factor analysis. In *Artificial Intelligence and Statistics*, 1462–1471. PMLR.