Variational Training for Large-Scale Noisy-OR Bayesian Networks

Geng Ji[1,2] **Dehua Cheng**[2] **Huazhong Ning**[2,3] **Changhe Yuan**[2,4]
Hanning Zhou[2] **Liang Xiong**[2] **Erik B. Sudderth**[1]

[1]UC Irvine [2]Facebook AI [3]WeRide.ai [4]CUNY Queens College

Abstract

We propose a stochastic variational inference algorithm for training large-scale Bayesian networks, where noisy-OR conditional distributions are used to capture higher-order relationships. One application is to the learning of hierarchical topic models for text data. While previous work has focused on two-layer networks popular in applications like medical diagnosis, we develop scalable algorithms for deep networks that capture a multi-level hierarchy of interactions. Our key innovation is a family of constrained variational bounds that only explicitly optimize posterior probabilities for the sub-graph of topics most related to the sparse observations in a given document. These constrained bounds have comparable accuracy but dramatically reduced computational cost. Using stochastic gradient updates based on our variational bounds, we learn noisy-OR Bayesian networks orders of magnitude faster than was possible with prior Monte Carlo learning algorithms, and provide a new tool for understanding large-scale binary data.

1 INTRODUCTION

Probabilistic graphical models provide an elegant, interpretable framework for characterizing uncertainty in relationships within high-dimensional data (Koller and Friedman, 2009). For binary directed graphical models, or Bayesian networks, noisy-OR conditional distributions effectively capture higher-order dependencies for applications including medical diagnosis (Shwe et al., 1991), dimensionality reduction (Šingliar and Hauskrecht, 2006), and text mining (Liu et al., 2016). Noisy-OR conditionals assume the activity of each variable is independently influenced by each parent, allowing correlations to be modeled with cost linear (rather than exponential) in the degree of each variable node.

While the restricted noisy-OR parameterization improves the efficiency of individual inference algorithm updates, standard methods struggle with web-scale data, where graphs with thousands of variables may be used to model corpora with millions of observations. In this paper, we develop a rigorous stochastic variational inference algorithm that allows training of noisy-OR Bayesian networks whose scale is orders of magnitude larger. Our approach involves three complementary technical innovations that enable learning of deep graph structures, with many thousands of variable nodes, from very large training databases.

Our first innovation is to develop a family of variational bounds (Wainwright and Jordan, 2008) that is applicable to deep hierarchies of variable relationships. Many prior noisy-OR Bayesian networks, like the classic QMR-DT network for medical diagnosis (Shwe et al., 1991), have a bipartite structure where all hidden (unobserved) variable nodes have no parents. There is an extensive literature on inference and learning algorithms tailored to this limited model family, including (Jaakkola and Jordan, 1999; Šingliar and Hauskrecht, 2006; Gogate and Domingos, 2010; Halpern and Sontag, 2013). However, such two-layer network structures are obviously limited by the assumption that the hidden "causal" variables are mutually independent. We generalize prior variational bounds for bipartite noisy-OR networks to support arbitrary directed acyclic graphs, and thus capture hierarchical dependencies among latent topics or causes. Unlike loopy belief propagation, which may be unstable for noisy-OR networks with sparse data (Murphy et al., 1999), our variational updates are always convergent.

Our second innovation enables scalability to graphs with large numbers of variables. Most prior work has focused on models with only hundreds of latent variables, due to limitations in computational speed and memory usage. We show that a rigorous family of constrained variational bounds may be constructed via a "local model" that only explicitly includes topic nodes connected to the set of active (positive) evidence nodes. Regardless of the overall model size, our variational bound may be optimized with cost proportional to the number of active observations; for real-world applications where observations are

typically sparse, the computational savings are dramatic.

Our third innovation enables scalability to big training databases. Standard variants of the *expectation maximization* (EM) algorithm, including Monte Carlo EM algorithms (Liu et al., 2016), must process all training data to compute the expected statistics required for each maximization step. For large corpora, each iteration may then take hours or days of computational effort. Moreover, some parameter update schemes require storage of intermediate variables that scales linearly with the number of nodes and training samples (Šingliar and Hauskrecht, 2006), which may lead to very high memory usage. We instead develop a variant of the *stochastic variational inference* (Hoffman et al., 2013) algorithm that incrementally samples small batches of data from the training corpus, uses variational inference to analyze that data given the current model, and then takes a (stochastic) gradient step to improve the weight parameters defining the noisy-OR network. This approach dramatically reduces memory usage and speeds convergence, and because our local models define rigorous variational bounds, the overall stochastic variational inference scheme is guaranteed to converge. We validate our approach using datasets of scientific abstracts from DBLP (Tang et al., 2008) and restaurant reviews from Yelp, and learn effective models for hundreds of thousands of documents and topics.

2 RELATED WORK

The QMR-DT network proposed by Shwe et al. (1991) is a two-layer, bipartite graph created by domain experts capturing how about 600 major diseases influence about 4000 possible symptoms. Each disease has an independent probability of producing each symptom, as integrated via noisy-OR conditionals (Horvitz et al., 1988).

Given an observed set of symptoms, the QMR-DT model is used to infer the posterior probability of each disease. Because exact inference is computationally infeasible, Shwe et al. (1991) used the bipartite network structure to develop a stochastic simulation algorithm. Other Monte Carlo methods like (Gogate and Domingos, 2010) support more general network structures, but become slow for graphs with hundreds of nodes. Alternatively, Jaakkola and Jordan (1999) derive variational upper and lower bounds for the QMR-DT posterior marginals, which we generalize in this work.

Two-layer noisy-OR belief networks (like QMR-DT) are sometimes called BN2O models (Henrion, 1991). To learn BN2O model parameters from observed data, Šingliar and Hauskrecht (2006) propose a variational EM approach based on the bounds of Jaakkola and Jordan (1999). Halpern and Sontag (2013) propose an alternative learning algorithm based on the method of moments which avoids local optima of the data log-likelihood, but requires the network to be sufficiently sparse.

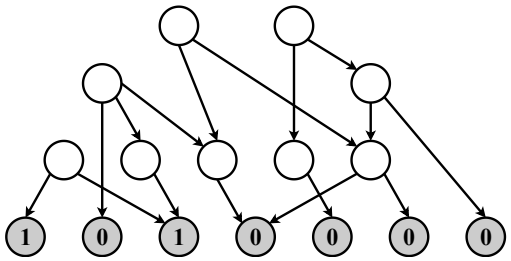

Figure 1: Graphical representation of a noisy-OR Bayesian network with binary variables. Shaded nodes are observed vocabulary tokens, and their ancestors correspond to hidden topics. The leak node is not shown.

It is attractive to generalize BN2O graph structures to deeper hierarchies capturing rich dependencies among hidden topics. Jaakkola and Jordan (2000) consider an alternative family of binary Bayesian networks with conditionals based on logistic regression. Murphy (2012, Sec. 26.5.4) briefly sketches a deep noisy-OR network used within Google to model the semantic content of text data, but provides few technical details. In this paper we generalize the variational bounds of Jaakkola and Jordan (1999) to support multi-layer noisy-OR networks, and formulate extensions that enable learning of large topic graphs from big document corpora.

Liu et al. (2016) also aim to learn general noisy-OR Bayesian networks, but instead propose a Monte Carlo method inspired by the independent cascade model (Wang et al., 2012). Some aspects of their approach are heuristic: log-likelihoods are scaled by token counts in a way that is not consistent with an underlying generative model, and no theory supports their restriction of sampling updates to document-specific subsets of the topic graph. We include comparisons to variants of their Monte Carlo inference algorithm in Sec. 6.

3 NOISY-OR BAYESIAN NETWORKS

We use binary Bayesian networks as in Fig. 1 to model vectors of binary features. For the text analysis applications that our experiments focus on, observations are indicators of whether particular tokens (words or phrases) appear in documents. Leaf nodes $j \in \mathcal{O}$ of the network correspond to the vocabulary, where $y_j = 1$ if term j appears in some document. The hidden topic nodes $i \in \mathcal{H}$ have binary variables $x_i \in \{0, 1\}$ indicating whether topics appear in that document. For notational simplicity we define a *leak node*, with index 0, that is always active ($x_0 = 1$). It allows some probability of token and topic

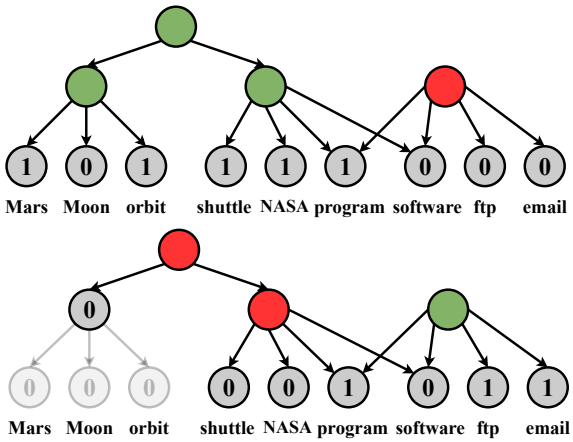

Figure 2: Local models for input queries about space science (top) and computer science (bottom). Our inference algorithm correctly infers the two different meanings of token "program" from its context. Topic nodes with variational probabilities greater than 0.5 are shaded green, and otherwise are shaded red. Some less relevant parts of the local models are not plotted to improve clarity.

activation even when other parent nodes are inactive.

Topic nodes are linked by an arbitrary directed acyclic graph, where $\mathcal{P}(i)$ are the parents of node i (excluding the leak node). Hierarchical relationships between topics are captured by the graph structure. Topic activation probabilities are defined by a noisy-OR distribution:

$$\begin{aligned} &p(x_i \mid x_{\mathcal{P}(i)}) \qquad (1)\\ &= \Big[1 - \exp\Big(-w_{0\to i} - \sum_{k\in\mathcal{P}(i)} w_{k\to i}\cdot x_k\Big)\Big]^{x_i}\\ &\times \Big[\exp\Big(-w_{0\to i} - \sum_{k\in\mathcal{P}(i)} w_{k\to i}\cdot x_k\Big)\Big]^{1-x_i}. \end{aligned}$$

Activation probabilities for tokens y_j are defined similarly. From Eq. (1) it follows that the influence of parent nodes factorizes. If parent k is active ($x_k = 1$), it activates child node i with probability $1 - \exp(-w_{k\to i})$, regardless of the states of other parents. If $x_k = 0$, parent k has no influence on the state of x_i. If all parents are inactive, the activation probability $1 - \exp(-w_{0\to i})$ is determined by the leak node.

The noisy-OR structure is useful for reasoning about cases where observations may have multiple hidden causes (Russell and Norvig, 2003): if a variable is active, then it is likely that at least one of its parents is also active. For example in medical diagnosis, it captures the fact the observed symptoms may be caused by multiple diseases. In hierarchical topic models it effectively captures polysemy, where a word or phrase may have multiple possible meanings. We provide an example in Fig. 2.

4 NOISY-OR STOCHASTIC VARIATIONAL INFERENCE

For each document d, we define a variational distribution $q(x^d)$ that factorizes over the hidden topics:

$$q(x^d) \triangleq \prod_{i\in\mathcal{H}} q(x_i^d) = \prod_{i\in\mathcal{H}} (q_i^d)^{x_i^d}(1-q_i^d)^{1-x_i^d}. \qquad (2)$$

Here q_i^d approximates the posterior probability that topic i is active in document d. As the leak node is always on, we fix $q_0^d = 1$. For any $q(x^d)$, the marginal log-likelihood of the observed tokens y^d can be lower bounded by Jensen's inequality as follows:

$$\begin{aligned} \log p(y^d) &\geq \mathbb{E}_{q(x^d)}\big[\log p(x^d, y^d) - \log q(x^d)\big] \qquad (3)\\ &= \sum_{i\in\mathcal{H}} \mathbb{E}_{q(x_i^d,\, x^d_{\mathcal{P}(i)})}\Big[\log p(x_i^d \mid x^d_{\mathcal{P}(i)})\Big]\\ &+ \sum_{j\in\mathcal{O}} \mathbb{E}_{q(x^d_{\mathcal{P}(j)})}\Big[\log p(y_j^d \mid x^d_{\mathcal{P}(j)})\Big]\\ &- \sum_{i\in\mathcal{H}} \Big[q_i^d \log q_i^d + (1-q_i^d)\log(1-q_i^d)\Big]. \end{aligned}$$

Using Eq. (1), the expectation of the noisy-OR log-probability for each topic can be decomposed as follows:

$$\begin{aligned} &\mathbb{E}_{q(x_i^d,\, x^d_{\mathcal{P}(i)})}\Big[\log p(x_i^d \mid x^d_{\mathcal{P}(i)})\Big] = q_i^d \cdot \qquad (4)\\ &\mathbb{E}_{q(x^d_{\mathcal{P}(i)})}\Big[\log\Big(1 - \exp\Big(-w_{0\to i} - \sum_{k\in\mathcal{P}(i)} w_{k\to i}x_k^d\Big)\Big)\Big]\\ &+ \Big(1-q_i^d\Big)\cdot\Big(-w_{0\to i} - \sum_{k\in\mathcal{P}(i)} w_{k\to i}q_k^d\Big). \end{aligned}$$

Due to the non-conjugate structure of the noisy-OR distribution, the expectation in the second line of Eq. (4) requires enumerating all joint states of the parent nodes, which has complexity exponential in the node degree. To simplify, we first define the *concave* function

$$f(a) \triangleq \log\big(1 - \exp(-a)\big). \qquad (5)$$

Because both $w_{0\to i}$ and $w_{k\to i}x_k^d$ are non-negative, we can use Jensen's inequality to derive a lower bound as in Jaakkola and Jordan (1999):

$$\begin{aligned} &f\big(w_{0\to i} + \sum_{k\in\mathcal{P}(i)} w_{k\to i}x_k^d\big) \qquad (6)\\ &\geq f(w_{0\to i}) + \sum_{k\in\mathcal{P}(i)} r_{k\to i}^d x_k^d\Big[f(u_{k\to i}^d) - f(w_{0\to i})\Big]. \end{aligned}$$

Here we introduce an auxiliary parameter $r_{k\to i}^d$ for each non-leak parent edge, with the constraints

$$r_{k\to i}^d \geq 0, \quad \sum_{k\in\mathcal{P}(i)} r_{k\to i}^d = 1, \qquad (7)$$

and define $u^d_{k\to i} \triangleq w_{0\to i} + w_{k\to i}/r^d_{k\to i}$. We then define a lower bound with complexity *linear* in the node degree:

$$\mathbb{E}_{q(x^d_{\mathcal{P}(i)})}\Big[f\Big(w_{0\to i} + \sum_{k\in\mathcal{P}(i)} w_{k\to i}x^d_k\Big)\Big] \geq \quad (8)$$
$$f(w_{0\to i}) + \sum_{k\in\mathcal{P}(i)} r^d_{k\to i}q^d_k\Big[f(u^d_{k\to i}) - f(w_{0\to i})\Big].$$

A similar lower bound can be constructed for token nodes' expectations of $\log p(y^d_j \mid x^d_{\mathcal{P}(j)})$ in Eq. (3). The overall variational objective for document d is then

$$\mathcal{L}_d(q^d, r^d, w) \triangleq \sum_{i\in\mathcal{H}} q^d_i \cdot \quad (9)$$
$$\Big[f(w_{0\to i}) + \sum_{k\in\mathcal{P}(i)} r^d_{k\to i}q^d_k\big(f(u^d_{k\to i}) - f(w_{0\to i})\big)\Big]$$
$$+ (1-q^d_i)\cdot\Big(-w_{0\to i} - \sum_{k\in\mathcal{P}(i)} w_{k\to i}q^d_k\Big) + \sum_{j\in\mathcal{O}} y^d_j \cdot$$
$$\Big[f(w_{0\to j}) + \sum_{k\in\mathcal{P}(j)} r^d_{k\to j}q^d_k\big(f(u^d_{k\to j}) - f(w_{0\to j})\big)\Big]$$
$$+ (1-y^d_j)\cdot\Big(-w_{0\to j} - \sum_{k\in\mathcal{P}(j)} w_{k\to j}q^d_k\Big)$$
$$- \sum_{i\in\mathcal{H}}\Big[q^d_i\log q^d_i + (1-q^d_i)\log(1-q^d_i)\Big].$$

4.1 EXPECTATION STEP

In this section, we derive closed-form update equations for local parameters q^d and r^d of each document. For notational simplicity, we omit the document index d.

4.1.1 Fixed Point Update for Edge Parameters r

For every topic and active token node i, we optimize the auxiliary parameters $r_{k\to i}$ given a fixed variational distribution q. Inactive tokens are excluded because if $y_j = 0$, the fourth line of Eq. (9) has no dependence on $r_{k\to j}$. We show in Appendix A that this optimization problem is concave and has a unique global maximum. Following Jaakkola and Jordan (1999) we derive a fixed-point algorithm by setting the partial derivative of Eq. (9) to zero after adding a Lagrange multiplier enforcing the normalization constraint of Eq. (7):

$$r_{k\to i} \propto q_k r_{k\to i} \quad (10)$$
$$\times\left[f(u_{k\to i}) - f(w_{0\to i}) - \frac{w_{k\to i}}{r_{k\to i}}\cdot f'(u_{k\to i})\right].$$

Here $f'(a) = \frac{\exp(-a)}{1-\exp(-a)}$ is the derivative of $f(a)$. Because the updates of r for different nodes are independent, the for-loop in line 15 of Alg. 1 may be easily parallelized. This iterative update monotonically increases $\mathcal{L}_d$ and rapidly converges to the global maximum.

4.1.2 Coordinate Update for Node Parameters q

To update the variational posterior q given fixed auxiliary parameters r, we cannot directly use prior work specialized to two-layer noisy-OR networks (Jaakkola and Jordan, 1999). Instead we directly optimize q by taking the partial derivative of Eq. (9) and setting to zero:

$$q_i = \frac{1}{1+\exp\big(-g(q_{\mathcal{P}(i)}, q_{\mathcal{C}(i)}, y, r, w)\big)}. \quad (11)$$

Here $\mathcal{C}(i)$ are the children of node i, and

$$g(\cdot) \triangleq f(w_{0\to i}) + w_{0\to i} + \sum_{k\in\mathcal{P}(i)} w_{k\to i}q_k \quad (12)$$
$$+ \sum_{k\in\mathcal{P}(i)} q_k r_{k\to i}\Big(f(u_{k\to i}) - f(w_{0\to i})\Big)$$
$$+ \sum_{\ell\in\mathcal{C}(i)\cap\mathcal{H}} q_\ell r_{i\to\ell}\Big(f(u_{i\to\ell}) - f(w_{0\to\ell})\Big)$$
$$- (1-q_\ell)w_{i\to\ell}$$
$$+ \sum_{m\in\mathcal{C}(i)\cap\mathcal{O}} y_m r_{i\to m}\Big(f(u_{i\to m}) - f(w_{0\to m})\Big)$$
$$- (1-y_m)w_{i\to m}.$$

The logistic function in Eq. (11) ensures $0 < q_i < 1$. The update for node i depends only on the states of its parents and children, not its full Markov blanket (which includes the childrens' parents), and is thus simpler than computing the posterior required by a Gibbs sampler.

4.1.3 Initialization of Expectation Parameters

The updates for q and r are coupled by the variational objective of Eq. (9). Our experiments initialize by setting $r_{k\to i} \propto w_{k\to i}$. In Appendix B, we show that this corresponds to the optimal solution whenever the activation probabilities q_k for all parent nodes $k \in \mathcal{P}(i)$ are equal.

4.2 NOISY-OR WEIGHT OPTIMIZATION

Given optimized local parameters for all data, previous work by Šingliar and Hauskrecht (2006) directly maximizes a (simplified, BN2O model) likelihood bound by solving a non-linear equation for each edge. This requires explicit storage of the E-step results for all documents, and thus has high computation and storage complexity scaling with the product of the number of nodes and documents. We instead employ stochastic gradient updates of the edge weights w, allowing parameter updates to be frequently interleaved with variational analyses of small batches of documents. Memory usage is also reduced because the variational posteriors for individual documents need not be explicitly stored.

4.2.1 Gradients for Non-leak Edge Weights

From Eq. (9), the partial derivative of an edge weight between (non-leak) topic node k and a topic node i is

$$\frac{\partial \mathcal{L}_d}{\partial w_{k\to i}} = q_k^d \left(\frac{q_i^d}{1 - \exp(-u_{k\to i}^d)} - 1 \right). \tag{13}$$

Similarly, if node k is linked to a token node j, then

$$\frac{\partial \mathcal{L}_d}{\partial w_{k\to j}} = q_k^d \left(\frac{y_j^d}{1 - \exp(-u_{k\to j}^d)} - 1 \right). \tag{14}$$

4.2.2 Gradient for Leak Edge Weights

For an edge between leak node 0 and topic node i,

$$\frac{\partial \mathcal{L}_d}{\partial w_{0\to i}} = q_i^d f'(w_{0\to i}) - (1 - q_i^d) + q_i^d \sum_{k \in \mathcal{P}(i)} q_k^d r_{k\to i}^d \big(f'(u_{k\to i}^d) - f'(w_{0\to i}) \big). \tag{15}$$

Similarly, if the leak node is linked to a token node j,

$$\frac{\partial \mathcal{L}_d}{\partial w_{0\to j}} = y_j^d f'(w_{0\to j}) - (1 - y_j^d) + y_j^d \sum_{k \in \mathcal{P}(j)} q_k^d r_{k\to j}^d \big(f'(u_{k\to j}^d) - f'(w_{0\to j}) \big). \tag{16}$$

Note that the gradient for non-leak edges depends only on the leak edge weight of the child node, but the gradient for leak edges depends on that child's other parents.

4.2.3 Stochastic Gradient Weight Updates

We use a variant of stochastic variational inference (Hoffman et al., 2013), where a stochastic estimate of the gradient of the variational bound is estimated from a mini-batch of sampled data. Due to the non-conjugate noisy-OR likelihood, we optimize a point estimate of the edge weights rather than a full posterior, as Paisley et al. (2012) did for logistic-normal distributions. The edge weights $w^{(t)}$ at iteration t are updated as follows:

$$w^{(t+1)} = w^{(t)} + \rho_t A \nabla \mathcal{L}_{\mathcal{D}^{(t)}}(w). \tag{17}$$

Here $\mathcal{D}^{(t)}$ is the mini-batch of data at iteration t. This stochastic scheme is guaranteed to converge to a local maximum of $\mathcal{L}$ if the learning rate ρ_t satisfies the conditions of Robbins and Monro (1951) and the preconditioner A is positive definite (Paisley et al., 2012). To ensure that all weights $w_{k\to i} > 0$, we use a projected gradient ascent algorithm that replaces any negative weights with a small constant: $w_{k\to i}^{(t+1)} \leftarrow \max(w_{k\to i}^{(t+1)}, \epsilon)$.

Our experiments use a constant learning rate ρ as in Mandt et al. (2017). While the simplest choice for the preconditioner A is the identity matrix, to accelerate convergence we scale the non-leak edges with a constant $c > 1$ so that their magnitudes are more comparable to the leak edges. Relative to more complicated scalings such as the inverse Hessian (Paisley et al., 2012) or Fisher information matrix (Hoffman et al., 2013), this simple preconditioner is more computationally efficient, while still rapidly converging to high-likelihood models.

Algorithm 1 Stochastic Variational Inference.

Input:
$w^{(t)}$: current edge weights
$\mathcal{D}^{(t)}$: data mini-batch for current iteration
$\{N_E, N_Q, N_R\}$: variational hyperparameters
$\{\rho, c\}$: weight update hyperparameters
Output:
$w^{(t+1)}$: updated edge weights

function STOCHASTICVARIATIONALUPDATE
  Initialize the gradient $\nabla \mathcal{L}_{\mathcal{D}^{(t)}} := 0$.
  # Variational Expectation Step
  for instance $d \in \mathcal{D}^{(t)}$ **do**
    Build local model as in Sec. 5.1.
    Initialize r^d as in Sec. 4.1.3.
    for $n_e := 1 \to N_E$ **do**
      # Update node parameters
      for $n_q := 1 \to N_Q$ **do**
        for $i \in \mathcal{H}_d$ **do**
          Update q_i^d using Eq. (11).
        end for
      end for
      # Update edge parameters
      for $i \in \{\mathcal{H}_d \cup \mathcal{O}_d^+\}$ **do**
        Update $r_{k\to i}^d$ using Eq. (10),
        $k \in \mathcal{P}(i)$; repeat N_R times.
      end for
    end for
    # Accumulate gradient information
    Compute $\nabla \mathcal{L}_d$ using Eqs. (13, 14, 15, 16).
    $\nabla \mathcal{L}_{\mathcal{D}^{(t)}}$ += $\nabla \mathcal{L}_d / |\mathcal{D}^{(t)}|$.
  end for
  # Stochastic Weight Optimization Step
  Apply the gradient update using Eq. (17).
  return $w^{(t+1)}$
end function

5 VARIATIONAL MODEL PRUNING

The stochastic variational inference algorithm of Sec. 4 still requires inference of all variational parameters for each document in the sampled mini-batch. For models defined by large directed graphs, this can have very high computational demands. We thus develop a more efficient algorithm that focuses only on document-specific

"local models", that contain a small subset of the nodes and edges of the full model. Computation then becomes *sub-linear* in the overall graph size, instead scaling with the number of *active* observations in each document. We first describe how to construct data-dependent local models, and then link to the variational updates of Sec. 4.

5.1 LOCAL MODEL CONSTRUCTION

Our construction of local models is motivated by the observation that real-world observations are typically *sparse*: only a small subset of token nodes are active for each document (Madsen et al., 2005). For inactive tokens, the posterior probability of their ancestor topics is typically very small. These parts of the graph have little influence on parameter updates because the absolute values of edge weight gradients, as in Eqs. (13,14), are proportional to topic activation probabilities q_k.

The goal of our local model construction process is to prune these irrelevant subsets of the graph, while still retaining the nodes that contain information crucial to the subsequent parameter update. Specifically, we construct a document-specific local model (as in Fig. 3) as follows:

1. Select $\mathcal{O}_d^+$, the set of active tokens for document d.
2. Select $\mathcal{H}_d$, the ancestors of nodes in O_d^+ excluding the leak node. We do explicit variational inference updates only for this subset of topic nodes.
3. Select the direct children of $\mathcal{H}_d$, which are a subset of the other topic nodes and the inactive tokens. Constrain their activation probabilities to zero.
4. Link the leak node to all of the other selected nodes.

5.2 LOCAL VARIATIONAL INFERENCE

We adapt the stochastic variational inference algorithm of Sec. 4 to the local model defined in Sec. 5.1, dramatically reducing computation and memory demands. Our theoretically sound approach optimizes a constrained family of variational bounds, whose optimum is similar to the original unconstrained variational bound.

5.2.1 Local Model Expectation Step

As can be verified from inspection of Eq. (9), performing an expectation step with our specified local model is equivalent to constrained variational inference on the full model where we fix $q_i^d = 0$ for all $i \in \mathcal{H} \setminus \mathcal{H}_d$. Adding constraints to the original optimization problem is equivalent to optimizing a lower bound on the original variational objective. As we verify empirically in Sec. 6, because we only apply constraints to topics that have no active descendants, the resulting local bound is a tight approximation. Note that for all $i \in \mathcal{H} \setminus \mathcal{H}_d$, fixing $q_i^d = 0$ also cancels the corresponding auxiliary variables $r_{k \to i}$ in Eq. (9), which need not be stored or updated.

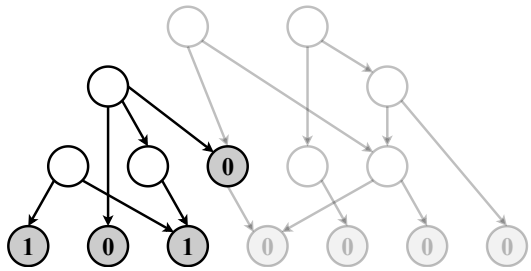

Figure 3: By selecting only the nodes most related to a sparse set of active tokens, local models may dramatically reduce the graph size. Here, lightly shaded nodes and edges are pruned. Comparing to the full model of Fig. 1, explicit inference of activation probabilities is only needed for three of nine topic nodes.

5.2.2 Local Model Weight Optimization

Although our expectation step is only explicitly performed for local models, we must ensure that gradient updates for all edge weights are still correctly computed. For edges that are included in the local model, we simply use the full model gradient updates from Sec. 4.2.

For non-leak edges outside the local model, each of their parent nodes k must satisfy $k \in \mathcal{H} \setminus \mathcal{H}_d$; if this were not true, then their children would be included in the local model. It thus follows that such parent nodes have activation probability $q_k^d = 0$, and according to Eqs. (13,14), the resulting gradient will also be exactly zero.

For leak edges outside the local model, the gradient of the edge weights can be shown to equal -1. We verify this by considering two cases. First, if the edge's child node j is a token, it must be inactive ($y_j^d = 0$). All terms scaled by y_j^d in Eq. (16) then cancel, and only the -1 remains. Alternatively, if the edge's child node i is a topic, then $i \in \mathcal{H} \setminus \mathcal{H}_d$ and $q_i^d = 0$. The partial gradient in Eq. (15) then simplifies to -1, reducing the prior activation probabilities for topics with no active descendants.

6 EXPERIMENTS

We now evaluate our variational training algorithm on datasets of various scales (see Table 1). Using a small corpus of newsgroup data where training with the full model is computationally feasible, we illustrate the effectiveness of our local model, the similarity of our variational estimates to expensive Monte Carlo approxima-

Table 1: Model Structure Statistics for Each Dataset

Dataset	# Topics	# Tokens	# Edges
Newsgroups	44	100	707
DBLP	49543	199861	1268551
Yelp	125798	117702	960419

tions, the influence of hyperparameter c on convergence speed, and qualitative features of learned topic models. Then on two larger datasets, we show that variational training with local models is the only computationally feasible option, and verify the improved efficiency of stochastic variational inference updates.

Our learning algorithm assumes the graph structure has already been determined, perhaps via external sources like knowledge bases. As we don't possess such metadata for the text data used in the experiments, we employ a greedy hierarchical clustering method that generalizes the DBScan algorithm (Ester et al., 1996). It constructs a layered graph structure recursively based on the co-occurrence statistics of token or topic pairs in the previous layer, and also prunes small edges to ensure sparsity. Our approach could be easily integrated with other, more advanced graph learning algorithms.

Unless specified otherwise, we set hyperparameters as follows: $N_E = N_Q = N_R = 10$, $\rho = 0.01$, $c = 1000$.

6.1 TINY 20 NEWSGROUPS

This dataset is a "tiny" version of the famous 20 Newsgroups corpus, with binary occurrence data for 100 words across 16,242 postings.[1] Each posting (document) is labeled with one of the four highest-level newsgroup categories. Our topic graph contains 44 topic nodes arranged in two layers, as summarized in Table 1.

Variational Inference via Local Models. For this small dataset, we compute gradients using the full dataset rather than stochastic mini-batches. 70% of the documents are randomly selected for training. On the remaining 30% we evaluate the average *evidence lower bound* (ELBO), by computing the mean of Eq. (9) across all test documents; see Table 2. The inference algorithm used to evaluate test documents (VI or MCMC, full or local model) is matched to that used during training. The quality of the initialization is assessed using local-model VI. Error bars indicate variability (under the same network structure) across five random train-test splits.

ELBO values in Table 2 indicate that our variational inference algorithm, whether using full or local models, increases the log-likelihood bound per document to about -14.5 from the initialization of -24.2. As one verification of the effectiveness of our variational optimization algorithm, these ELBO values are higher than those achieved by MCMC (-15.4), which exactly computes marginal probabilities in the limit where the number of sampling iterations becomes very large (Liu et al., 2016).

Table 2: Average Held-out ELBO and Log Likelihood of Tiny 20 Newsgroups Dataset $\pm$ Two Standard Deviations

Method	ELBO	Log-Likelihood
VI full	-14.50 ± 0.06	-14.43 ± 0.07
VI local	-14.51 ± 0.08	-14.43 ± 0.07
MCMC full	-15.36 ± 0.15	-14.18 ± 0.07
MCMC local	-19.22 ± 0.47	-17.11 ± 0.12
Initialization	-24.15 ± 0.11	-21.98 ± 0.08

More importantly, we find that the difference between the variational bounds achieved by full and local model training is negligible (-14.50 vs -14.51, smaller than the variability from the train-test split). This comparison justifies our use of local models for larger datasets, where full-model variational inference is prohibitively slow.

As a baseline, we also tried MCMC training using local models constructed in the same way. Compared to using the full model, MCMC test log-likelihoods drop dramatically from -14.2 to -17.1. This deterioration is probably caused by our deterministic procedure for constructing local models, which causes the MCMC edge weight updates to be systematically biased. In contrast, for variational inference local models lead to a principled family of bounds on the overall log-likelihood.

Lower Bounds on Data Log-Likelihood. We also approximately evaluate the marginal log-likelihood of the observed test documents. We construct a simple lower bound by summing up the joint probabilities of all the unique samples drawn over one million iterations of MCMC inference. This lower bound is potentially conservative, because there are $2^{44} \approx 10^{13}$ possible configurations of the latent topic variables. Nevertheless, we verify that our variational objective does bound these approximate log-likelihoods by checking that the MCMC estimates always exceed the corresponding ELBO values in Table 2. Previous work demonstrated the accuracy of variational bounds for directed graphical models with discrete hidden variables (Beal and Ghahramani, 2006).

Convergence Speed Acceleration. The preconditioner A was set to an identity matrix when running the preceding experiments. With this choice, thousands of iterations are required for convergence. Empirically, this

[1] https://cs.nyu.edu/~roweis/data/20news_w100.mat

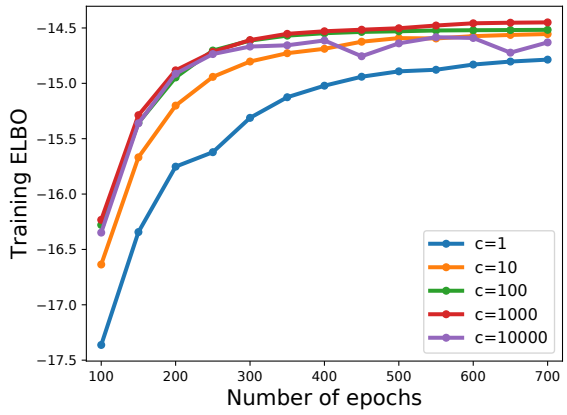

Figure 4: Accelerated convergence via hyperparameter c, the scaling of non-leak edge weights in the preconditioner A for stochastic gradient updates (see Sec. 4.2.3).

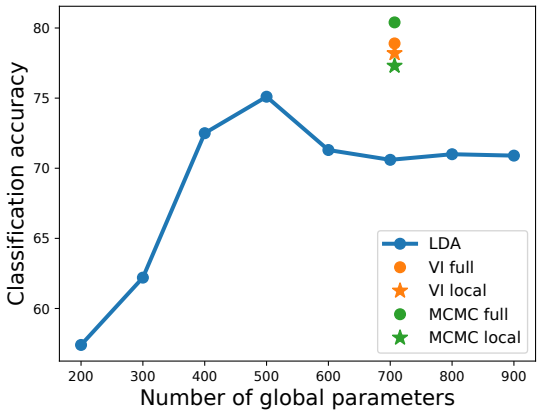

Figure 5: Classification accuracy on the tiny 20 newsgroups dataset for variants of our training algorithm, and Bernoulli LDA models with $2 \leq K \leq 9$ topics.

occurs because the gradients for non-leak edge weights have small magnitude, often about two or three orders of magnitude smaller than the gradients for leak edge weights. To better balance these gradients and improve convergence speed, we explore alternative values for the preconditioning hyperparameter c defined in Sec. 4.2.3.

As shown in Fig. 4, the learning algorithm does not converge after hundreds of epochs when $c = 1$. As we increase its value, the magnitudes of leak and non-leak gradients become better balanced, so that convergence becomes much more rapid. The fastest convergence speed is reached when $100 \leq c \leq 1000$. For larger values of c, the step size for non-leak edges becomes too large and optimization may become unstable.

Qualitative and Quantitative Analysis. Qualitatively, running inference on our trained model naturally visualizes the activated topics of input queries. Fig. 2 shows two examples where the activated topics are each related to space and computer science. In particular, as the token "program" has different meanings for each area, different topics are activated based on the context provided by other tokens. Other tokens like "software" have only one meaning, but may nevertheless be shared among multiple topics. The strength of each relationship in the topic graph is determined by the learned edge weights.

Topic models are sometimes used to define features for document classification and retrieval (Yi and Allan, 2009). We use the activation probabilities q^d of each document d as a feature representation for classification tasks. One-vs-all linear SVMs (Fan et al., 2008) are trained based on the four newsgroup labels, where the regularization parameter is selected via five-fold cross-validation. To make features more consistent across documents, activation probabilities are standarized within each document by subtracting the mean and dividing by the standard deviation. The baseline model we compare with is *latent Dirichlet allocation* (LDA, Blei et al. (2003)), where multinomial topics are replaced by Bernoulli distributions to model binary observations. For variational training, the numbers of global parameters in LDA is the product of the vocabulary size (100 in this case) and topic count K. Fig. 5 provides the results when $2 \leq K \leq 9$, which is of similar size to our model that contains 707 edges. The LDA models reach the best performance in this range when K is 4 or 5, corresponding roughly to the 4 newsgroup labels. The different variants of our graph-based learning algorithms are all superior.

6.2 DBLP PAPERS AND YELP REVIEWS

Now we evaluate our algorithm on two larger datasets. The first one comes from the DBLP bibliography of major computer science publications (Tang et al., 2008).[2] We get 430,213 training documents by extracting paper titles and abstracts in venues for database, data mining, machine learning, natural language processing, and computer vision research. The other dataset is constructed from the Yelp Open Dataset[3], where we extract reviews from the top 250 businesses in the "Restaurants" category to produce 483,448 training documents. The tokens for each document are segmented using the method of Liu et al. (2015), which removes both rare and common (stop) words, and also groups words into common phrases. We build a four-layer topic graph for each dataset, whose statistics are summarized in Table 1.

[2] http://aminer.org/billboard/aminernetwork
[3] https://www.yelp.com/dataset

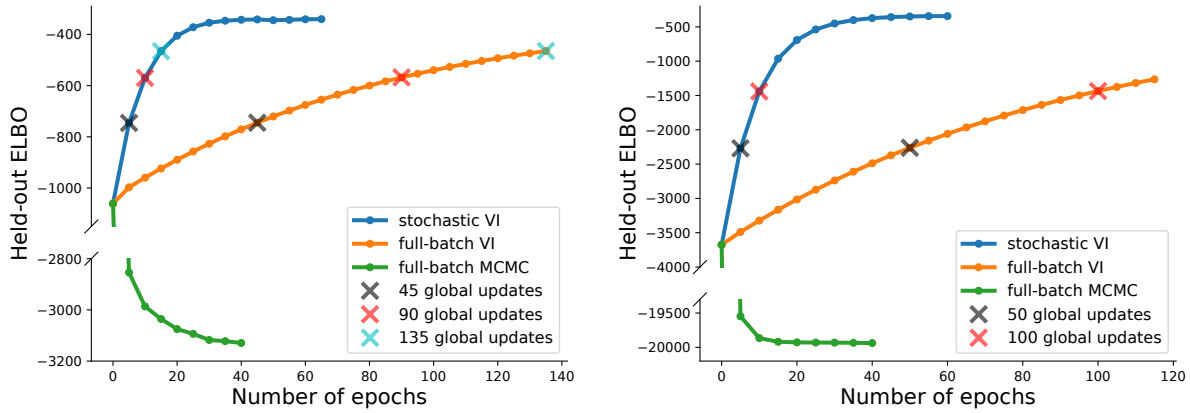

Figure 6: ELBO evaluations on the test sets of DBLP (left) and Yelp (right). In both cases, stochastic variational inference converges much faster than standard, full-batch inference. Each pair of X markers in the plots compares equal numbers of edge weight updates. Their tiny differences in ELBO values indicates that the noise in stochastic gradient updates does not have a significant impact on the convergence speed. Held-out ELBO values decrease over time for MCMC, likely due to biases caused by its heuristic use of local models. (A regularizer is added to MCMC to avoid edge weights decaying to zero; without this, its performance deteriorates further.)

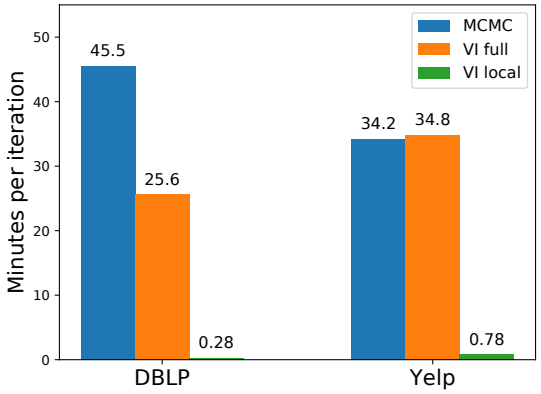

Figure 7: Time used for running one E-step iteration on the full-batch data of DBLP and Yelp using 50 CPUs. Local-model VI is the only feasible option for both datasets to run multiple inference iterations till convergence. As restaurant reviews are usually longer than paper titles and abstracts, local models for Yelp tend to be larger than DBLP, and thus need more time for inference.

For models of these scales, the only computationally feasible option is variational training on local models (Fig. 7). In Fig. 6, we compare the convergence speed of full-batch and stochastic training, with mini batches of 50,000 documents. Test documents are the same as in Liu et al. (2016), with 500 paper abstracts for DBLP and 1000 restaurant reviews for Yelp. Each point in the plot represents the average held-out ELBO evaluated using the full model. By interleaving local and global updates more frequently, stochastic training converges much faster than full-batch inference for both datasets.

As gradient-based weight updates are very fast, the variational inference updates in the expectation step dominate computation time. The overhead required by frequent stochastic weight updates is thus negligible.

7 DISCUSSION

We have developed a stochastic variational inference algorithm for training large-scale, hierarchical noisy-OR Bayesian networks. We use these models to capture high-order dependencies within the hidden topics and observed tokens in text data. By exploiting the sparsity of input data, our method creates a rigorous variational bound for each document that significantly prunes the model for fast inference. This principled algorithm scales the learning of noisy-OR networks to data and models that are orders of magnitude larger than prior work focusing on simpler, bipartite graphs. Our algorithms could potentially be used to model causal interactions within many other types of data, adapted to other model families like the noisy-AND networks used in educational assessment (Conati et al., 1997), or extended to learn graph structures jointly with their parameters.

Acknowledgements

This research supported in part by NSF CAREER Award No. IIS-1758028. The authors thank Xing Wang, Cheng Cheng, Min Li, Fangbo Tao, Tiangao Gou, and Shilin Ding for early discussions about this work. Michael Hughes and Jialu Liu provided helpful information about baseline topic models and inference algorithms.

References

Beal, M. J. and Ghahramani, Z. (2006). Variational Bayesian learning of directed graphical models with hidden variables. *Bayesian Analysis*, 1(4):793–831.

Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022.

Conati, C., Gertner, A. S., VanLehn, K., and Druzdzel, M. J. (1997). On-line student modeling for coached problem solving using Bayesian networks. In *International Conf. on User Modeling*, pages 231–242.

Ester, M., Kriegel, H.-P., Sander, J., and Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In *International Conf. on Knowledge Discovery and Data Mining*, volume 96, pages 226–231.

Fan, R.-E., Chang, K.-W., Hsieh, C.-J., Wang, X.-R., and Lin, C.-J. (2008). LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research*, 9:1871–1874.

Gogate, V. and Domingos, P. (2010). Formula-based probabilistic inference. In *Conf. on Uncertainty in Artificial Intelligence*, pages 210–219.

Halpern, Y. and Sontag, D. (2013). Unsupervised learning of noisy-OR Bayesian networks. In *Conf. on Uncertainty in Artificial Intelligence*, pages 272–281.

Henrion, M. (1991). Search-based methods to bound diagnostic probabilities in very large belief nets. In *Conf. on Uncertainty in Artificial Intelligence*, pages 142–150.

Hoffman, M. D., Blei, D. M., Wang, C., and Paisley, J. (2013). Stochastic variational inference. *Journal of Machine Learning Research*, 14(1):1303–1347.

Horvitz, E. J., Breese, J. S., and Henrion, M. (1988). Decision theory in expert systems and artificial intelligence. *International Journal of Approximate Reasoning*, 2(3):247–302.

Jaakkola, T. S. and Jordan, M. I. (1999). Variational probabilistic inference and the QMR-DT network. *Jour. of Artificial Intelligence Research*, 10:291–322.

Jaakkola, T. S. and Jordan, M. I. (2000). Bayesian parameter estimation via variational methods. *Statistics and Computing*, 10:25–37.

Koller, D. and Friedman, N. (2009). *Probabilistic Graphical Models: Principles and Techniques*. MIT Press.

Liu, J., Ren, X., Shang, J., Cassidy, T., Voss, C. R., and Han, J. (2016). Representing documents via latent keyphrase inference. In *International Conf. on World Wide Web*, pages 1057–1067.

Liu, J., Shang, J., Wang, C., Ren, X., and Han, J. (2015). Mining quality phrases from massive text corpora. In *International Conf. on Management of Data*, pages 1729–1744.

Madsen, R. E., Kauchak, D., and Elkan, C. (2005). Modeling word burstiness using the Dirichlet distribution. In *International Conf. on Machine Learning*, pages 545–552.

Mandt, S., Hoffman, M. D., and Blei, D. M. (2017). Stochastic gradient descent as approximate Bayesian inference. *Journal of Machine Learning Research*, 18(1):4873–4907.

Murphy, K. P. (2012). *Machine Learning: A Probabilistic Perspective*. MIT Press.

Murphy, K. P., Weiss, Y., and Jordan, M. I. (1999). Loopy belief propagation for approximate inference: An empirical study. In *Conf. on Uncertainty in Artificial Intelligence*, pages 467–475.

Paisley, J., Wang, C., and Blei, D. M. (2012). The discrete infinite logistic normal distribution. *Bayesian Analysis*, 7(2):235–272.

Robbins, H. and Monro, S. (1951). A stochastic approximation method. *The Annals of Mathematical Statistics*, pages 400–407.

Russell, S. J. and Norvig, P. (2003). *Artificial Intelligence: A Modern Approach*. Prentice Hall.

Shwe, M. A., Middleton, B., Heckerman, D. E., Henrion, M., Horvitz, E. J., Lehmann, H. P., and Cooper, G. F. (1991). Probabilistic diagnosis using a reformulation of the INTERNIST-1/QMR knowledge base. *Methods of Information in Medicine*, 30(4):241–255.

Šingliar, T. and Hauskrecht, M. (2006). Noisy-OR component analysis and its application to link analysis. *Journal of Machine Learning Research*, 7:2189–2213.

Tang, J., Zhang, J., Yao, L., Li, J., Zhang, L., and Su, Z. (2008). ArnetMiner: Extraction and mining of academic social networks. In *International Conf. on Knowledge Discovery and Data Mining*, pages 990–998.

Wainwright, M. J. and Jordan, M. I. (2008). Graphical models, exponential families, and variational inference. *Foundations and Trends in Machine Learning*, 1:1–305.

Wang, C., Chen, W., and Wang, Y. (2012). Scalable influence maximization for independent cascade model in large-scale social networks. *Data Mining and Knowledge Discovery*, 25(3):545–576.

Yi, X. and Allan, J. (2009). A comparative study of utilizing topic models for information retrieval. In *European Conf. on Information Retrieval*, pages 29–41.