Deep Exponential Families

Rajesh Ranganath
rajeshr@cs.princeton.edu
Department of Computer Science
Princeton University

Linpeng Tang
linpengt@cs.princeton.edu
Department of Computer Science
Princeton University

Laurent Charlin
lcharlin@cs.columbia.edu
Department of Computer Science
Columbia University

David M. Blei
david.blei@columbia.edu
Department of Computer Science
Columbia University

Abstract

We describe *deep exponential families*, a class of random effects models that are inspired by the hidden structures used in deep neural networks. These models allow for hierarchical dependencies between latent variables to be captured, while the language of exponential families allows for new models to be described quickly. We perform inference in these models using recent “black box” variational inference techniques. We evaluate various DEFs on text and collaborative filtering. We show both improved predictive performance along with exploratory analysis that shows that the models capture hierarchical structure found in the data.

1 Introduction

In this paper we develop deep exponential families (DEFs), a flexible family of probability distributions that reflect the intuitions behind deep unsupervised feature learning. In a DEF, observations arise from a cascade of layers of latent variables. Each layer’s variables are drawn from an exponential family that is governed by the inner product of the previous layer’s variables and a set of layer-specific latent variables (weights). As in deep unsupervised feature learning, a DEF represents hidden patterns, from coarse to fine grained, that compose with each other to form the observations. DEFs also enjoy the advantages of probabilistic modeling. Through their connection to exponential families [5], they support many kinds of data both at the observed and latent levels. Further, they can be connected and combined into more complex models.

For example, consider the problem of modeling documents. We can represent a document as a vector of term counts. In one type of DEF, each term’s count is modeled as a Poisson random variable; its rate is an inner product of a layer of latent variables (one level up from the terms) and a set of weights that are shared across documents. Loosely, we can think of the latent layer above the observations as “topics”, each of which activates a set of related terms via their inner product with the weights. These latent topic counts are, in turn, modeled the same way, conditioned on a layer above of “super topics.” Just as the topics group related terms, the super topics group related topics (again via the inner product). Note that this style of model, though with different details, has been previously studied in the topic modeling literature [17]. Figure 1 illustrates an example of a 2-level DEF uncovered from a large collection of news articles.

We emphasize that this is just one example. In a DEF, the latent variables can be from any exponential family: Bernoulli latent variables recover the classical sigmoid belief network [21]; Gamma latent variables give something akin to deep version of nonnegative matrix factorization [16]; Gaussian latent variables lead to the types of models that have recently been explored in the context of

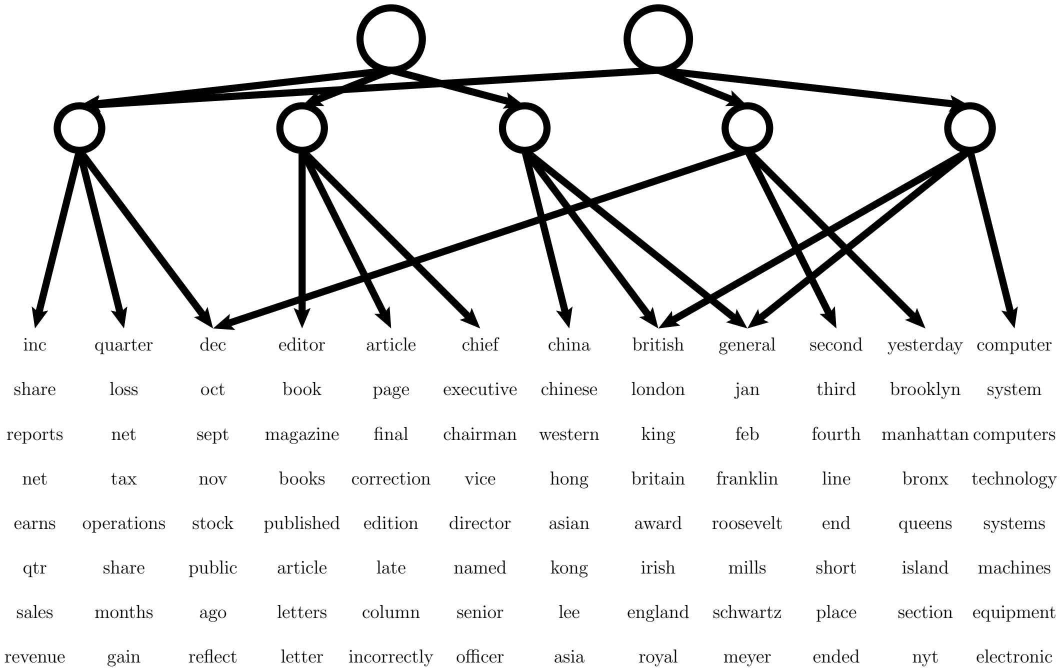

inc share reports net earns qtr sales revenue
quarter loss net tax operations share months gain
dec oct sept nov stock public ago reflect
editor book magazine books published article letters letter
article page final correction edition late column incorrectly
chief executive chairman vice director named senior officer
china chinese western hong asian kong lee asia
british london king britain award irish england royal
general jan feb franklin roosevelt mills schwartz meyer
second third fourth line end short place ended
yesterday brooklyn manhattan bronx queens island section nyt
computer system computers technology systems machines equipment electronic

Figure 1: A fraction of the three layer topic hierarchy of the *New York Times*. We choose 2 concepts and then show the top-3 super topics from each concepts, and top-3 topics from each super topics. The top words are shown for each topic. The arrows represent hierarchical groupings.

computer vision [23]. We may further change the prior on the weights and the observation model. In the language of neural networks, the prior on the weights amounts to choosing a type of regularization; the observation model amounts to choosing a type of loss. Finally, as with any probabilistic model, we can embed the DEF in a more complex model. As examples, the DEF can be made part of a multi-level model of grouped data [8], time-series model of sequential data, or a factorization model of pairwise data [26].

Below, we will define and develop deep exponential families, explain some of their properties, and situate them in the larger contexts of probabilistic models and neural networks. For example, in a DEF there is a function that maps an inner product to a natural parameter—a similar mapping exists in generalized linear models [18]—and this is a source of non-linearity in the exponential family analogue of a deep neural network. In general, the relationship between simple linear regression and Bayesian generalized linear models is mirrored in the relationship between deep neural networks and deep exponential families.

We will develop algorithms for using DEFs to solve real-world problems. Given a data set of observations, we estimate the conditional distribution of the latent layers of hidden variables and the latent parameters. Unfortunately DEFs are a nonconjugate model—their local conditional distributions are difficult to compute—and so they are not amenable to simple inference algorithms like Gibbs sampling [9] or variational message passing [10]. We will leverage recent results in inference for nonconjugate models [31, 27, 22] to develop general-purpose variational algorithms. On both document models and recommendation system data, we demonstrate that our algorithms let us easily explore and compare a particular Gamma DEFs for modern data analysis problems.

2 Deep exponential families

Exponential families [5] are a class of distributions with convenient mathematical properties. They take the following form.

$$p(x) = h(x) \exp(\eta^\top T(x) - a(\eta)),$$

where h is the base measure, η are the natural parameters, T are the sufficient statistics, and a is the log normalizer. The expectation of the sufficient statistics of an exponential family is given by the gradient of the log normalizer $\mathrm{E}[T(x)] = \nabla_\eta a(\eta)$. Exponential families are completely specified by their sufficient statistics and base measure; different choices of h and T lead to different distributions. For example, for the normal distribution the base measure is $h = \sqrt{(2\pi)}$ and the sufficient statistics are $T(x) = [x, x^2]$; and for the Beta distribution, a distribution with support over $[0, 1]$, the base measure is $h = 1$ and sufficient statistics are $T(x) = [\log x, \log 1 - x]$.

To construct deep exponential families, we will chain exponential families together in a hierarchy, where a draw from one layer controls the natural parameters of the next. We formalize this using the notation $\text{EXPFAM}(x, \eta)$, where the first argument is the variable and the second is the natural parameter of the exponential family.[1]

For each data point x_n, the model has L layers of hidden variables $z_{n,1:L}$, where each $z_{n,\ell}$ is a collection of K_ℓ random variables, $z_{n,\ell,1:K_\ell}$. We assume that $z_{n,\ell,k}$ is a scalar, but the model generalizes beyond this. Shared across data, the model has L layers of weights $W_{1:L}$, where each W_ℓ is a collection of K_ℓ vectors, each one with dimension $K_{\ell-1}$. We assume the weights have a prior distribution on them given by $p(W_\ell)$

For simplicity, we omit the data index n and describe the distribution of a single data point x. First, each latent variable is drawn conditional on the previous layer,[2]

$$p(z_{\ell,k} \mid z_{\ell+1}) = \text{EXPFAM}_\ell(z_{\ell,k}, g_\ell(z_{\ell+1}^\top W_{\ell+1,k})). \tag{1}$$

Confirm that the dimensions work: $z_{\ell,k}$ is a scalar; $z_{\ell+1}$ and $W_{\ell+1,k}$ are both $K_{\ell+1}$ vectors. Note each of the k variables in layer ℓ depends on all the variables of the higher layer. This gives the model the flavor of a neural network. EXPFAM_ℓ indicates that the exponential family may vary at each layer. We assume a prior on the top layer $p(z_L) = \text{EXPFAM}_L(\eta)$ where η is the prior parameter.

This hierarchy of latent variables defines the DEF. The data are drawn conditioned on the lower layer of the DEF, $p(x_{n,i} \mid z_{n,1})$. In this paper, we focus on count data, thus we use the Poisson distribution as the observation likelihood. The Poisson distribution with mean λ is

$$p(x_{n,i} = z) = e^{-\lambda} \frac{\lambda^x}{x!}.$$

If we let $x_{n,i}$ be the count of type i associated with observation n, then $x_{n,i}$'s distribution is

$$p(x_{n,i} \mid z_1, W_1) = \text{Poisson}(z_{n,1}^\top W_{1,i}),$$

where W_1 is a matrix of latent variables shared across observations.

Returning to the example from the introduction of modeling documents, the x_n are a vector of term counts. This means the observation weights W_1 put positive mass on groups of terms. Thus, they form “topics.” Similarly, the weights on the second layer represents “super topics.” The distribution $p(z_{n,1} \mid z_{n,2})$ represents the distribution of “topics” given the “super topics” of a document. Figure 3 depicts the compositional and sharing semantics of DEFs.

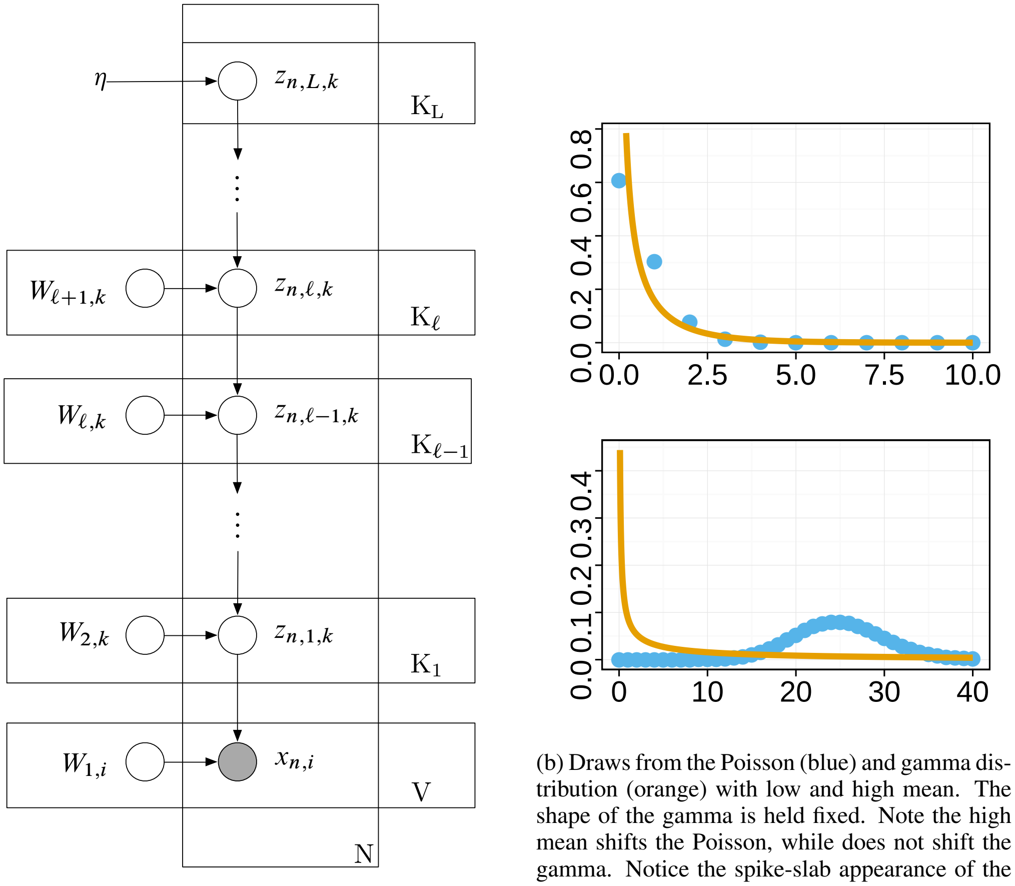
The Link Function The latent variable layers in deep exponential families are connected together via a link function, g_ℓ. Specifically the natural parameters for $z_{\ell,k}$ are specified by the inner product of $z_{\ell+1}$ with the weights $W_{\ell+1,k}$ passed through g_ℓ. Using properties of exponential families we can determine how the link function alters the distribution of the ℓth layer. The moments of the sufficient statistics of an exponential family are given by the gradient of the log normalizer $\nabla_\eta a(\eta)$. These moments completely specify the exponential family. Thus in DEFs, the mean of the next layer is controlled by the link function g_l via the gradient of the log normalizer,

$$\nabla_\eta a(g_l(z_{\ell+1}^\top W_{\ell+1,k})). \tag{2}$$

Consider the case of the identity link function, where $g_l(x) = x$. In this case, the expectation in deep exponential families gets transformed by the log-normalizer at each level. This transformation of

[1] We are loose with the base measure, h as it can be absorbed into the dominating measure.

[2] This inner product is passed through a function g_ℓ to obtain the natural parameter. We call this the link function.



(a) The deep exponential family

(b) Draws from the Poisson (blue) and gamma distribution (orange) with low and high mean. The shape of the gamma is held fixed. Note the high mean shifts the Poisson, while does not shift the gamma. Notice the spike-slab appearance of the gamma distribution.

Figure 2

the expectation is one source of non-linearity in DEFs. It parallels the non-linearities used in neural networks.

To illustrate the potential of deep exponential families, we focus on a single example: the sparse gamma DEF. We note that we are just beginning to explore the possibilities of DEFs. As such Gamma DEFs provide a proof of concept architecture. We are interested in exploring other latent exponential-family representations as well as combinations of different exponential-family representations for latent and visible units. Specifically, we are currently investigating three additional DEF architectures. First, sigmoid belief networks which have been widely used [21, 19]. We are also exploring Poisson DEFs which can be seen as an more general version of sigmoid belief nets. Finally, we are exploring compositions of multiple DEFs. Such compositions allow us to explore deep extensions of common models (e.g., matrix factorization for collaborative filtering). Note that although still preliminary we have obtained good empirical results using all of aforementioned DEF architectures.

Sparse Gamma DEF The gamma distribution is an exponential family distribution with support over the positive reals. The standard form of the density in exponential family form with natural parameters, α and β, is

$$p(z) = z^{-1} \exp(\alpha \log(z) - \beta z - \log \Gamma(\alpha) - \alpha \log(\beta)).$$

where Γ is the Gamma function. When α is small this distribution puts most of its mass near zero. The expectation of the gamma distribution is given by $\alpha\beta^{-1}$.

Through the link function in DEFs, the inner product of the previous layer and the weights control the natural parameters of the next layer. For sparse gamma models, we want components in a layer to control the expected activation of the next layer, while the shape at each layer remains constant.

Let α_ℓ be the shape at layer ℓ, then the link function is

$$g_1 = \alpha_\ell, \quad g_2 = \frac{\alpha_\ell}{z_{\ell+1}^\top W_{\ell+1,k}}.$$

As the expectation of gamma variables needs to be positive, we choose the gamma distribution as the prior on the weights.

The sparse gamma DEF differs in how the distribution changes given a change in the mean from common distributions such as the normal and Poisson. For example in the Poisson distribution, when the expected value is high, that draws are likely to be much larger than zero. While in the case of the sparse gamma DEF, when the expectation is high, z will either be close to zero or very large. This is akin to a soft spike-slab prior [14]. Figure 2b visually demonstrates this by plotting both the Poisson and gamma distribution in both settings.

We estimate the posterior on this DEF using one to three layers for a large text copora: *New York Times* (NYT). We find that the perplexity decreases as we increase the number of layers in our model Table 1. Figure 1 displays a portion of the topic hierarchy for the NYT corpus. We find the similar topics are grouped into "super topics" and "super topics" are grouped into higher level "concepts". This forms a semantic hierarchy that can be used for exploratory data analysis. In the appendix we present a larger chunk of the *New York Times* hierarchy. We differ the discussion of the details of the corpora and evaluation metric to Section 4.

Related Research. Graphical models and neural nets have a long and distinguished history. A full review is outside of the scope of this article, however we highlight some key results as they relate to DEFs. Deep exponential families fall into the broad class of stochastic feed forward belief networks [21], but Neal [21] only shows one example in this class, the sigmoid belief network, which is a binary latent variable model. Several existing stochastic feed forward networks are DEFs, such as latent Gaussian models [23] and the sigmoid belief network with layer-wise dependencies [19].

Undirected graphical models have also been used in inferring compositional hierarchies. Salakhutdinov and Hinton [25] proposes deep probabilistic models based on Restricted Boltzmann Machines (RBMs) [29]. RBMs are a two layer undirected probabilistic model with one layer of latent variables and one layer of observations tied together by a weight matrix. Directed models such as DEFs have the property of explaining away, where independent latent variables under the prior become

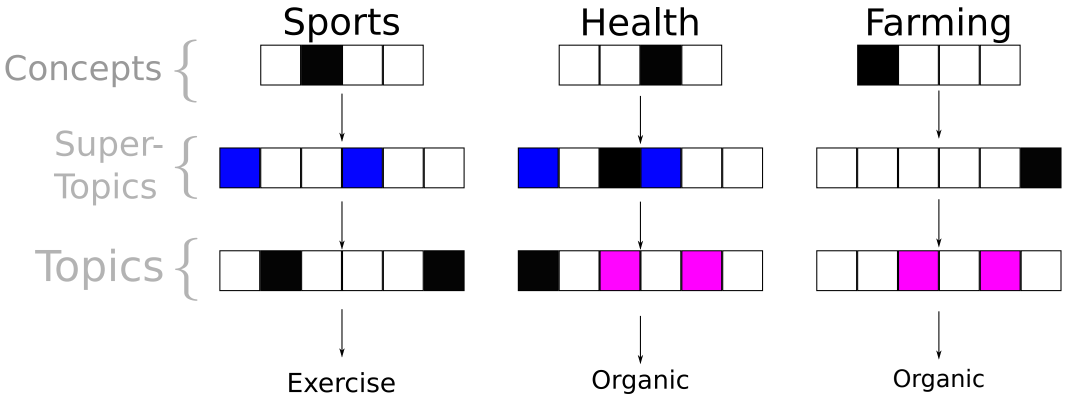


Figure 3: We depict the composition capacities of DEF through the generation of words for three different documents. Activated units are colored. The *sports* and the *health* concepts share some activated super topic (blue units). While *health* and *farming* do not yet they are able to generate the same word (*organic*) by activating similar topics (pink units).

dependent conditioned on the observations. This property makes inference harder than in RBMs, but forces a more parsimonious representation where similar features compete to explain the data rather than work in tandem [11, 1].

RBMs have been extended to general exponential family conditionals in a model called exponential family harmoniums (EFH) [30]. A certain infinite DEF with tied weights is equivalent to an EFH [13], but as our weights are not tied, deep exponential families represent a broader class of models than exponential family harmoniums (and RBMs).

The literature of latent variable models relates to DEFs through hierarchical models and Bayesian factor analysis. Latent tree hierarchies have been constructed with specific distributions (Dirichlet) [17], while Bayesian factor analysis methods such as exponential family PCA [20] and multinomial PCA [6] can be seen as a single layer deep exponential family.

3 Inference

The central computational problem for working with DEFs is posterior inference. The intractability of the partition function means posterior computations require approximations. Past work on sigmoid belief networks has proposed doing greedy layer-wise learning (for a specific kind of network) [13]. Here, instead, we develop variational methods [28] that are applicable to general DEFs. We emphasize that we propose approximating the posterior on all latent layers simultaneously.

Variational inference [15] casts the posterior inference problem as an optimization problem. Variational algorithms seek to minimize the KL divergence to the posterior from an approximating distribution q. This is equivalent to maximizing the following [2],

$$\mathcal{L}(\lambda) = \mathrm{E}_{q_\lambda(z)}[\log p(x, z, W) - \log q(z, W)].$$

The free parameter λ indexes the variational distribution. This objective function is called the Evidence Lower BOund (ELBO) because it is a lower bound on $\log p(x)$.

For the approximating distribution, q, we use the mean field variational family. In this approximating family, the distribution over the latent variables factorizes. That is with n observations, the variational family is

$$q(z, W) = \prod_{\ell=1}^{L} q(W_\ell) \prod_{n=1}^{N} q(z_{n,\ell}),$$

where $q(z_{n,\ell})$ and $q(W_\ell)$ are fully factorized. Each component in $q(z_{n,\ell})$ is

$$q(z_{n,\ell,k}) = \textsc{expfam}_\ell(\lambda_{z_{n,l,k}}),$$

where the exponential family is the same one as the model distribution p. Similarly, we choose $q(W)$ to be in the same family as $p(W)$.

To maximize the ELBO, we need to compute expectations under the approximation q. These expectations for general DEFs will not have a simple analytic form. Thus we use more recent "black box" variational inference techniques that step around computing this expectation [31, 27, 22].

Black box variational inference methods use stochastic optimization[24] to avoid the analytic intractability of computing the objective function. Stochastic optimization works by following noisy unbiased gradients. In black box variational inference [22], the gradient of the ELBO with respect to the parameters of a latent variable can be written as an expectation with respect to the variational approximation. More formally, if we let $p_{n,\ell,k}(x, z, W)$ be the terms in the log joint that contains $z_{n,\ell,k}$ (its Markov blanket), then the gradient for the variational approximation of $z_{n,\ell,k}$ is

$$\mathrm{E}_q[\nabla_{\lambda_{z_{n,\ell,k}}} \log q(z_{n,\ell,k})(\log p_{n,\ell,k}(x, z, W) - \log q(z_{n,\ell,k}))].$$

To compute the Monte Carlo estimate of the gradient, we need to be able to evaluate the Markov blanket for each latent variable, the approximating distribution, and the gradient of the log of the approximating distribution (score functions). For the sparse gamma model, the score functions we need are for the gamma distribution. The score functions for the gamma distribution are

$$\frac{\partial \log q(z)}{\partial \alpha} = -\varphi(\alpha) - \log\theta, + \log z$$
$$\frac{\partial \log q(z)}{\partial \theta} = -\alpha/\theta + z/\theta^2,$$

where φ is the digamma function.

The Markov blanket for a latent variable in the first layer is

$$\log p_{n,1,k}(x, z, W) = \log p(z_{n,1,k}|z_{n,2}) + \log p(x_n|z_{n,1}, W).$$

The Markov blanket for a latent variable in the intermediate layer is

$$\log p_{n,\ell,k}(x, z, W) = \log p(z_{n,\ell,k}|z_{n,\ell+1}, W_{\ell+1}) + \log p(z_{n,\ell-1}|z_{n,\ell}, W_\ell).$$

The Markov blanket for the top layer is

$$\log p_{n,L,k}(x, z, W) = \log p(z_{n,L,k}) + \log p(z_{n,L-1}|z_{n,L}, W_L).$$

The gradients and Markov blankets for W can be written similarly.

Stochastic optimization requires a learning rate to scale the noisy gradients before applying them to the current parameters. We use RMSProp which scales the gradient by the square root of the online average of the squared gradient.[3] RMSProp captures the varying length scales and noise through the sum of squares term used to normalize.

4 Experiments

In this section, we provide details on our predictive metric, held out perplexity, along with establishing a baseline.

Baseline As a baseline we consider Poisson factorization [7, 12]. Poisson factorization, generates the word count $x_{n,i}$ as

$$p(x_{n,i}) = \mathrm{Poisson}(\theta_n^\top \beta_i),$$

where θ_n and β_i are K dimensional iid gamma vectors. This model has been shown in Canny [7] to have superior predictive performance to the popular text model latent Dirichlet allocation [4]. Poisson factorization is equivalent to the 1-layer sparse gamma model. Thus in the perplexity table we label it as both.

Datasets We consider the *New York Times* corpora. It consists of 165,120 documents with 8,000 terms.

[3] http://www.cs.toronto.edu/~tijmen/csc321/slides/lecture_slides_lec6.pdf

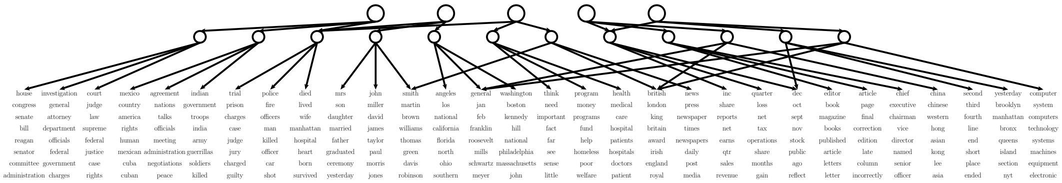

house congress senate bill reagan senator committee administration
investigation general attorney department officials federal government charges
court judge law supreme federal justice case rights
mexico country america rights human mexican cuba cuban
agreement nations talks officials meeting administration negotiations peace
indian government troops india army guerrillas soldiers killed
trial prison charges case judge jury charged guilty
police fire officers man killed officer car shot
died lived wife manhattan hospital heart born survived
mrs son daughter married father graduated ceremony yesterday
john miller david james taylor paul morris jones
smith martin brown williams thomas green davis robinson
angeles los national california florida north ohio southern
general jan feb franklin roosevelt mills schwartz meyer
washington boston kennedy hill national philadelphia massachusetts john
think need important fact far see sense little
program money programs fund help homeless poor welfare
health medical care hospital patients hospitals doctors patient
british london king britain award irish england royal
news press newspaper times newspapers daily post media
inc share reports net earns qtr sales revenue
quarter loss net tax operations share months gain
dec oct sept nov stock public ago reflect
editor book magazine books published article letters letter
article page final correction edition late column incorrectly
chief executive chairman vice director named senior officer
china chinese western hong asian kong lee asia
second third fourth line end short place ended
yesterday brooklyn manhattan bronx queens island section nyt
computer system computers technology systems machines equipment electronic

Figure 4: A fraction of the three layer topic hierarchy of the *New York Times*. The top words are shown for each topic. The arrows represent hierarchical groupings. Better seen on on screen at high resolution.

DEF Type	PF/100	100-30	100-30-15
Sparse Gamma	2396	2151	2106

Table 1: Predictive likelihood on held out collection of 1K *New York Times* documents. We find that going to deeper DEFs improves predictive likelihoods. We emphasize that the one layer sparse gamma DEF is equivalent to Poisson factorization (PF) our baseline.

Evaluation We compute perplexity on a held out set of 1,000 documents. Held out perplexity is given by

$$\exp\left(\frac{-\sum_{d\in\text{docs}}\sum_{w\in d}\log p(w\,|\,\#\text{ held out in }d)}{N_{\text{held out words}}}\right).$$

Condtional on the total number of held out words, the distribution of the held out words becomes multinomial. The mean of the conditional multinomial is given by the normalized Poisson rate in each document. We set the rates to the expected value under the variational distribution. To compute the variational distribution for the document specific latent variables, the DEF for the document, we use ten percent of the words; the other ninety percent form the held out set.

Architectures and Hyperparameters We build one, two and three layer hierarchies of the sparse-Gamma DEF, The sizes of the layers are 100, 30, and 15 respectively. We choose one hundred topics as it fall into the range of topics searched in the topic modeling literature [3]. We detail the hyperparameters for each DEF in the appendix.

Results Table 1 summarizes the predictive results on the *New York Times* corpus. We find that adding more layers improves prediction. We find similar results on a large corpus of scientific documents.

Topic hierarchy We choose 5 concepts and then show the top-3 super topics from each concepts, and top-3 topics from each super topics. The top words are shown for each topic. Note the overlapping hierarchical structure.

5 Discussion

We develop deep exponential families as a way to describe hierarchical relationships of latent variables to capture compositional semantics of data. We present several instantiations of deep exponential families and achieve improved predictive power and interpretable semantic structures for the problem of text. Our current research focuses on exploring other DEF architectures as well as on variational inference techniques. Specifically we are researching structured variational inference methods (similar to [19]) as well as methods which further exploit the properties of exponential family distributions.

References

[1] Y. Bengio, A. Courville, and P. Vincent. Representation learning: A review and new perspectives. *Pattern Analysis and Machine Intelligence*, 35(8), 2013.

[2] C. Bishop. *Pattern Recognition and Machine Learning*. Springer New York., 2006.

[3] D. Blei and J. Lafferty. A correlated topic model of Science. *Annals of Applied Stat.*, 1(1):17–35, 2007.

[4] D. Blei, T. Griffiths, M. Jordan, and J. Tenenbaum. Hierarchical topic models and the nested Chinese restaurant process. In *NIPS*, 2003.

[5] L. Brown. *Fundamentals of Statistical Exponential Families*. Institute of Mathematical Statistics, 1986.

[6] W. Buntine and A. Jakulin. Applying discrete PCA in data analysis. In *Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence*, pages 59–66. AUAI Press, 2004.

[7] J. Canny. GaP: A factor model for discrete data. In *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2004.

[8] A. Gelman and J. Hill. *Data Analysis Using Regression and Multilevel/Hierarchical Models*. Cambridge Univ. Press, 2007.

[9] S. Geman and D. Geman. Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6:721–741, 1984.

[10] Z. Ghahramani and M. Beal. Propagation algorithms for variational Bayesian learning. In *NIPS 13*, pages 507–513, 2001.

[11] I. Goodfellow, A. Courville, and Y. Bengio. Large-scale feature learning with spike-and-slab sparse coding. In *International Conference on Machine Learning (ICML)*, 2012.

[12] P. Gopalan, J. Hoffman, and D. Blei. Scalable recommendation with poisson factorization. *arXiv*, (1311.1704), 2013.

[13] G. Hinton, S. Osindero, and Y. Teh. A fast learning algorithm for deep belief nets. *Neural Comput.*, 18 (7):1527–1554, July 2006. ISSN 0899-7667.

[14] H. Ishwaran and S. Rao. Spike and slab variable selection: Frequentist and Bayesian strategies. *The Annals of Statistics*, 33(2):730–773, 2005.

[15] M. Jordan, Z. Ghahramani, T. Jaakkola, and L. Saul. Introduction to variational methods for graphical models. *Machine Learning*, 37:183–233, 1999.

[16] D. Lee and H. Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401 (6755):788–791, October 1999.

[17] W. Li and A. McCallum. Pachinko allocation:dag-stuctured mixture models of topic correlations. In *ICML*, 2006.

[18] P. McCullagh and J. Nelder. *Generalized Linear Models*. London: Chapman and Hall, 1989.

[19] A. Mnih and K. Gregor. Neural variational inference and learning in belief networks. In *ICML*, 2014.

[20] S. Mohamed, K. Heller, and Z. Ghahramani. Bayesian exponential family PCA. In *NIPS*, 2008.

[21] R. Neal. Learning stochastic feedforward networks. *Tech. Rep. CRG-TR-90-7: Department of Computer Science, University of Toronto*, 1990.

[22] R. Ranganath, S. Gerrish, and D. Blei. Black box variational inference. In *International Conference on Artifical Intelligence and Statistics*, 2014.

[23] D. Rezende, S. Mohamed, and D. Wierstra. Stochastic backpropagation and approximate inference in deep generative models. *ArXiv e-prints*, January 2014.

[24] H. Robbins and S. Monro. A stochastic approximation method. *The Annals of Mathematical Statistics*, 22(3):pp. 400–407, 1951.

[25] R. Salakhutdinov and G. Hinton. Deep boltzmann machines. In *AISTATS*, pages 448–455, 2009.

[26] R. Salakhutdinov and A. Mnih. Bayesian probabilistic matrix factorization using Markov chain Monte Carlo. In *International Conference on Machine learning*, 2008.

[27] T. Salimans and D. Knowles. Fixed-form variational posterior approximation through stochastic linear regression. *Bayesian Analysis*, 8(4):837–882, 2013.

[28] L. Saul, T. Jaakkola, and M. Jordan. Mean field theory for sigmoid belief networks. *Journal of Artificial Intelligence Research*, 4:61–76, 1996.

[29] P. Smolenksy. Information processing in dynamical systems: Foundations of harmony theory. *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, 1, 1986.

[30] M. Welling, M. Rosen-Zvi, and G. Hinton. Exponential family harmoniums with an application to information retrieval. In *Neural Information Processing Systems (NIPS) 17*, 2004.

[31] D. Wingate and T Weber. Automated variational inference in probabilistic programming. *ArXiv e-prints*, January 2013.