Another Hierarchical Topic Model

Jason D. M. Rennie
jrennie@gmail.com

October 27, 2005

Abstract

We describe a hierarchical topic model. We assume that there are various levels of specificity in a document collection. For example, a collection of mailing list posts might be organized according to sentence, paragraph, post and thread. We describe a model that captures the structure at each level of the hierarchy. We use a trace norm penalty on a matrix composed of natural parameters for the multinomial model.

1 The Basic Model

We consider a probabilitistic model of text. We assume that a set of documents is generated in two stages. First, a set of document models are generated according to a prior model. Then, words for each document are generated according to that document's model. We assume that a document's term frequencies are generated independently of other documents', when conditioning on the document's model. We use a trace norm to penalize document models' divergence from the prior model, effectively placing a Gaussian prior on the singular vectors of the matrix composed of stacked document model parameter vectors. We use the rest of this section to describe the model in detail.

Let ϕ be the multinomial natural parameter vector for the "prior" model; ϕ represents somewhat of a "center" from which the individual document models emanate. Each document has its own multinomial model, with a natural parameter vector, θ_i. We define the prior on the document models as a characterization of where the document models are located with respect to the prior model. Let $\Theta = \begin{bmatrix} \theta_1 - \phi \\ \theta_2 - \phi \\ \vdots \\ \theta_n - \phi \end{bmatrix}$ be that matrix of stacked difference vectors, one vector per row. Then, the prior distribution on document models is

$$P(\theta_1, \dots, \theta_n; \phi) \propto \exp\left(-\lambda \left\|\Theta\right\|_{\text{tr}}\right). \tag{1}$$

The trace norm calculates the sum of singular values of the matrix, so this prior is effectively a Gaussian prior (with mean ϕ and variance $\frac{1}{\lambda} I$) on a basis for

Θ. We assume that a document's word counts, $\vec{x}$, are generated as a simple multinomial, θ_i. That is

$$P(\vec{x}|\theta_i) \propto \prod_j \left(\frac{\exp(\theta_i)}{\sum_{i'} \exp(\theta_{i'})} \right)^{x_j}. \tag{2}$$

The joint probability of the model is the product of the prior and all of the individual document likelihoods:

$$P(\theta_1, \ldots, \theta_n, \vec{x}_1, \ldots, \vec{x}_n; \phi) \propto P(\theta_1, \ldots, \theta_n; \phi) \prod_i P(\vec{x}_i|\theta_i). \tag{3}$$

We use maximum a posteriori to optimize the parameters. In other words, we find the parameters that maximize the joint probability (3).

Note that [1] introduced the idea of using the trace norm as a surrogate to a rank constraint. [2] describe the use of the trace norm in learning and particularly in a collaborative filtering application.

2 The Hierarchical Model

Here we describe how this basic model can be extended to the case of data with hierarchical structure. The hierarchical model is a simple extension of the basic model. Whereas we assumed a single collection of documents and a single, fixed prior for the basic model, we assume multiple document collections with one prior for each collection. Furthermore, we assume that the prior for each collection is itself regularized via a prior-on-priors, or a hyper-prior.

We have already specified models for producing (1) term frequency counts, and (2) the individual document model. Left is for us to describe the process of generating the set of prior models. We use a process identical to that of generating the document models. Let ψ be our hyper-prior. Let $\Phi = \begin{bmatrix} \phi_1 - \psi \\ \phi_2 - \psi \\ \vdots \\ \phi_n - \psi \end{bmatrix}$ be a matrix of stacked difference vectors, one vector per row; each row describes a prior model's location relative to the hyper-prior. We establish a distribution on prior models utilizing the trace norm of this matrix,

$$P(\phi_1, \ldots, \phi_n; \psi) \propto \exp\left(-\lambda \left\|\Phi\right\|_{\text{tr}}\right). \tag{4}$$

Then, the joint probability is proportional to the product of all the individual collection models (3) and this hyper-prior (4). Again, we use maximum a posteriori to learn the parameters. This time, our parameters include those of the document models $\{\theta\}$ and the prior models $\{\phi\}$.

Note that the hierarchical model can be further extended by adding additional levels. Also, there is no need that the leaf nodes (corresponding to individual documents) be at the same "level" of the hierarchy. Our model simply specifies that the distribution of the set of child parameter vectors is proportional to a constant multiple of the trace norm, exponentiated.

2.1 A Note on Documents

Though we have discussed this model as though the leaf nodes correspond to individual documents, there is nothing barring us from using other segmentations of text. For example, the two-layer model discussed above might correspond to a single document, with paragraph models corresponding to the ϕ and sentence models corresponding to the θ with the $\vec{x}$ being individual sentence term frequency vectors.

3 The Sequential Model

The hierarchical model can be further extended to deal with sequential structure in data. By sequential structure, we mean that the content of a document is related to the content of neighboring documents. For example, if the documents are updates on a particular news story, then developments may be such that the last update is only generally related to the first; but, each update will compare and constrast with the previous updated. Also, if we treat each paragraph in an article as a separate "document," then we also find sequential structure; neighboring paragraphs are more likely to share content than distant ones.

To capture sequential structure, we in effect penalize model differences between adjacent documents. Our hierarchical model penalizes difference from a parent model. This sequential penalty works in conjunction with and is in addition to the hierarchical penalty. Before, our model prior included only a trace norm term,

$$P_{\text{hier}}(\theta_1, \ldots, \theta_n; \phi) \propto \exp\left(-\lambda \left\|\Theta\right\|_{\text{tr}}\right). \tag{5}$$

Now, we include additional terms for each of the neighbors that we believe have sequential structure. Consider the case that the $\theta_1, \ldots, \theta_n$ are paragraph models and assume that the paragraphs have sequential structure. We introduce L_2 penalty terms which correspond to the sequential structure:

$$P_{\text{hier}}(\theta_1, \ldots, \theta_n; \phi) \propto \exp\left(-\lambda \left\|\Theta\right\|_{\text{tr}}\right) \prod_{i=1}^{n-1} \exp\left(-\lambda \|\theta_i - \theta_{i+1}\|_2\right), \tag{6}$$

where $\|\vec{v}\|_2 = \sqrt{\sum_i v_i^2}$ is the L_2 vector norm. Note that the L_2 norm penalty is a simplification of the trace norm. The trace norm of a vector (treating it as a matrix) is simply the L_2-norm or Euclidean length of that vector. A vector interpreted as a matrix is its own only singular vector with a corresponding non-zero singular value.

4 Applications

4.1 Clustering

Documents can be clustered using their model representation, or, equivalently, using their representation as weights in a weighted sum of singular vectors.

Other nodes in the hierarchy can also be clustered. If our data collection is posts to a web bulletin-board, then we might use sentences as the data primitive ("document") and cluster the parameter vectors that correspond to posts, or whole threads.

In a more standard clustering algorithm, the number of clusters is fixed, the "location" of the clusters is unrestricted and each item or document may only be assigned to a single cluster. Other algorithms first project the data down to a lower-dimensional space (of pre-determined size) and cluster the projected points. Our model used as a clustering algorithm calculates a projection. However, the dimensionality of that projection space is not limited a-priori, but rather is determined jointly by the regularization parameter and the nature of the data. Also, whereas most projection algorithms simply limit the dimensionality of the projection space, our model penalizes the size of the smallest hyper-ellipsoid that contains the projected points—it penalizes not only the number of dimensions, but also the degree to which those dimensions are used. This is, of course, the nature of the trace norm that we employ for our model.

4.2 Segmentation

Again consider a case where the data primitive ("document") is a small unit such as the sentence. One question we might ask is whether there are any clear breaks, or abrupt changes in content or topic. Our model allows such changes to be identified. Consider a sequential model where a level of the hierarchy represents paragraphs—each node corresponds to a single paragraph. Our sequential model penalizes differences between adjacent paragraph models. But, if there is sufficient evidence in the data, the paragraph models may show a large difference. Using these differences, we can identify paragraphs that exhibit a large topic from the previous paragraph. One simple algorithm for this purpose is to establish a threshold and identify all paragraphs with a difference larger than the threshold.

4.3 Anomaly Detection

Our model can also easily be applied to the problem of anomaly detection—identifying documents that are outliers or don't "fit." Consider the simplest version of our model, where there is a "root" node and a single layer of children models, each child representing the model for a document. These document models are a projection of the documents onto a restricted space; documents that are outliers will have corresponding models that are far (in terms of Euclidean distance) from other document models. Thus, we can identify outliers as those with the largest nearest neighbor distance.

5 The Trace Norm Prior

We have discussed the use of the following distribution on (multinomial natural) parameters:

$$P(\theta_1, \ldots, \theta_n; \phi) \propto \exp\left(-\lambda \|\Theta\|_{\mathrm{tr}}\right). \tag{7}$$

Worth noting is the fact that if we marginalize out the parent variable, ϕ, we are left with a joint distribution that cannot be written as a product of marginal distributions over the individual variables. I.e.

$$P(\theta_1, \ldots, \theta_n) \neq \prod_i P(\theta_i). \tag{8}$$

Even when the parent node, ϕ, is left unspecified, the distribution of the children, $\theta_1, \ldots, \theta_n$, is dependent on their relative locations.

References

[1] M. Fazel. *Matrix Rank Minimization with Applications.* PhD thesis, Stanford University, 2002.

[2] N. Srebro, J. D. M. Rennie, and T. S. Jaakkola. Maximum-margin matrix factorization. In *Advances in Neural Information Processing Systems 17*, 2005.