TOPIC MODELLING: CASE STUDY AND CLASSIFICATION

ABHINAV JAIN, DEPARTMENT OF ELECTRICAL ENGINEERING, I.I.T KANPUR

SUPERVISER

Dr. Piyush Rai

Dr. Gaurav Pandey

ABSTRACT

In the paradigm of topic modelling, we aim to discover themes that pervade a large and otherwise unstructured collection of documents. Through topic modelling we can organize the collection according to discovered themes. We would further like to bias our model towards the class/labels that characterizes each document. Models studied are based on factor analysis through a probabilistic angle.

1. Motivation

In the realm of researchers, there is a need to explore and browse large collections of papers, journals and organize them so that only relevant documents and underlying unexplored topics can be inferred. A scientist faced with such large corpus can't possibly explore them manually. So there is a need to digitize the requirement and this what topic modelling aims at. Topic models explore the collection through the underlying topics that run through the documents.

Imagine searching and exploring documents based on themes that run through them. We might know our query for the search not through keywords but through themes that document is based on. These specific or broader themes are the ones which we are interested in. Topic models examine these documents related to a theme and lets us know the extent to which a particular theme is present in a document.

Topic models are the probabilistic models which aims at uncovering the underlying semantic structure of a document collection based on a Hierarchical Bayesian analysis. Following a bag of words model, algorithms that does topic modelling basically model documents as mixture of topics and cluster words into these topics. These algorithms can be deployed not just to textual data but to images as well. They have been used to find patterns in genetic data, images and social networks.

A lot of approaches have been developed to handle topic modelling like unigram models, mixture of unigrams, PLSI model and the popular LDA[8].

We will discuss different models which relaxes the assumptions of previously discussed models imparting flexibility and higher accuracy to topic models. Follow-up for the rest of report is as follows. Section 2 will discuss the PFA (Poisson Factor Analysis) model in particular. Section 3 and 4 will extend the PFA model by incorporating the deep architecture to infer something called meta-topics. Finally, section 5 will discuss the contribution of this report in classifying the documents against the labels that categorizes documents into classes. Last section will deal with the experiment and results.

2. Poisson Factor analysis[6]

The idea behind PFA is to model documents as arising from multiple topics, where a document is defined as a distribution over a fixed vocabulary of terms. We assume that K number of topics can be associated with a collection of documents and that each document exhibits these topics with different proportions. This is a safe assumption to make because documents in a corpus are heterogeneous which tend to combine a subset of themes that permeates the collection as a whole.

We can think of each document as a distribution over K topics and each topic as a distribution over M words. We also assume that we have a vocabulary of M words that will take into account the vastness of each document. Observed data are the word counts of each document and the hidden variables represent the topics themselves and how each document exhibits them. Given a collection, posterior distribution of the hidden variables given the observed counts determines a hidden decomposition of documents into topics.

PFA can be visualized as a generative process, an imaginary random process that is assumed to have produced the observed data.

We have document-word count matrix, $\mathbf{X} \in \mathbf{Z}_{+}^{\boldsymbol{M\ x\ N}}$ **;** where N is the number of documents present in the corpus. PFA models

$$\mathbf{X} \sim \text{Poisson}\ (\emptyset(\boldsymbol{\theta o H})) \qquad (1)$$

Where,

- $\emptyset \in \boldsymbol{R}_{+}^{\boldsymbol{M\ x\ K}}$, factor loadings matrix with K factors. Each column k represent how words from the vocabulary are distributed in topic k.
- $\boldsymbol{\theta}$n $\in \boldsymbol{R}_{+}^{\boldsymbol{K}}$, nth column represents factor intensities that is each column is the topic proportions in a document n.
- $\boldsymbol{h}n \in \{0,1\}^{K}$ hidden set of binary units indicating which factors (k) are active for observation n
- $\boldsymbol{o}$ **symbol** stands for element wise multiplication of θ and H

For each element of **X,** $x_{mn} = \sum_{k=1}^{K} x_{mnk}$: that is each count of word m in document n can be thought of as coming from topics ∈ {1, … …,K}.

x_{mnk} ~ Poisson ($\emptyset_{mk}\boldsymbol{\theta}_{kn}h_{kn}$), where $\emptyset_{mk}\boldsymbol{\theta}_{kn}h_{kn}$ is the rate parameter of the Poisson.

Having defined our model, we go fully Bayesian to infer values for these parameters. This requires our prior beliefs for the parameters that define a state of the PFA model:

- $\emptyset k$ ~ Dirichlet ($\eta \mathbf{1_M}$), where $\emptyset k$ is the kth column of $\emptyset$ and $\mathbf{1_M}$ is an M-dimensional vector of ones. (η controls for sparsity in $\emptyset$)
- θkn ~ Gamma(r_k , $\frac{1\text{-b}}{\text{b}}$)
- Assuming further a prior on r_k ~ Gamma(γ_o , $\frac{1}{c_o}$) : r_k controls for over dispersion (observed variance > theoretical variance i.e. there was more variation than predicted) in X_n via θn
- h_{kn} ~ Bernoulli(π_{kn}), where $\pi_{kn} = \pi_k \sim$ Beta($c\varepsilon$, $c(1-\epsilon)$) : This is called as **Beta-Bernoulli process.**

All these parameters are being sampled from a distribution. There are a lot of sampling methods available like Importance sampling, Rejection Sampling. Inference details are given in section 5.

3. Deep representation with SBN network[1]

The main idea that revolves around this model is that we can infer interaction between topics through deep latent binary hierarchies and this gives a deep architecture to the model where we get an intuitive sense of what is called as "Meta-Topics". These meta-topics further groups topics to give an intuition about what the document is all about in a broader sense.

$$\text{We model } \mathbf{X} \sim \text{Poisson } (\emptyset(\boldsymbol{\theta}\mathbf{o}\boldsymbol{H}^{(1)})) \qquad (2)$$

Novelty in the model comes from introducing $\mathbf{H}^{(1)}$ with prior on $h_n^{(1)}$. To construct a structured prior, we define another hidden set of units $h_n^{(2)} \in \{0,1\}^{K_2}$ placed at a layer above $h_n^{(1)}$. The layers are related through a set of weights defined by weight matrix $\boldsymbol{W}^{(1)} = [w_1^{(1)} \ldots w_{K1}^{(1)}]^{\mathrm{T}} \in \boldsymbol{R}^{K1\,x\,K2}$. The global parameters $\boldsymbol{W}^{(1)}$ are used to characterize mapping between layers for all documents. The SBN model with 2 layers has the following generative process:

$$\mathrm{p}\ (h_{k2n}^{(2)} = 1) = \sigma\left(c_{k2}^{(2)}\right) \quad \text{(prior on top layer)} \qquad (3)$$

$$\mathrm{p}\ (h_{k1n}^{(1)} = 1 \mid \boldsymbol{h}_n^{(2)}\ ) = \sigma(\left(w_{k1}^{(1)}\right)\boldsymbol{h}_n^{(2)} + \ c_{k1}^{(1)}) \text{ (logistic function)} \quad (4)$$

Similarly, we can add multiple layers to obtain a deep architecture (figure 1):

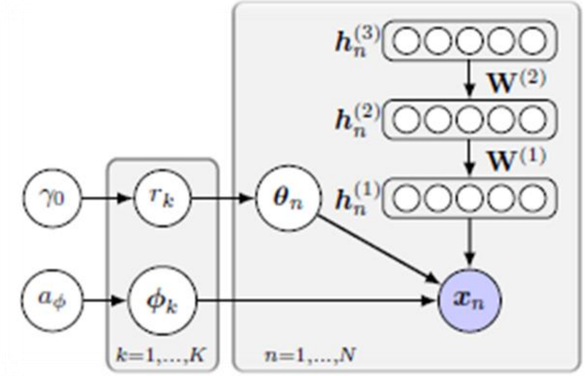

Figure 1. Graphical model for the Deep PFA with three layers of hidden binary hierarchies

$$\mathrm{p}(\boldsymbol{h}_n^{(1)}, \ldots, \boldsymbol{h}_n^{(L)}) = \mathrm{p}(\boldsymbol{h}_n^{(L)}) \textstyle\prod_{l=2}^{L} \mathrm{p}(\boldsymbol{h}_n^{(l)} | \boldsymbol{h}_n^{(l-1)}) \qquad (5)$$

Where, L is the number of layers. $\mathrm{p}(\boldsymbol{h}_n^{(L)})$ is the prior on the top layer as in (3). $\mathrm{p}(\boldsymbol{h}_n^{(l)} | \boldsymbol{h}_n^{(l-1)})$ is defined as in (4). Note that K2 can be inferred as number of meta-topics with each meta-topic having weights for each topic $k_1 \in \{1, \ldots \ldots, K_1\}$.

Inference for the SBN network using Gibbs Sampling is bit complicated. For details paper[2] can referred.

4. Deep representation with PFA modules

In PFA, we assumed that $\pi_{kn} = \ \pi_k$ which implied every document has on average the same probability of seeing a particular topic as active. It also assumed that topics are independent of each other. Both of these assumptions are restrictive and a lot of models have tried to relax these assumptions. In particular Model proposed in previous section also relaxes those assumptions by discarding the need for global Beta-Bernoulli process. This allowed documents to have individual topic activation probabilities to accommodate heterogeneity in data. Also, the model should considered the fact that topics are likely to co-occur simultaneously.

Model proposed here also follows the same rationale and relaxes those assumptions. It also has advantages over the SBN model which are discussed later in this section.

For a binary vector h_n, we can write

$$h_{kn}^{(1)} = 1\ (z_{kn}^{(2)} \geq 1) \text{ where, } z_{kn}^{(2)} \sim \text{Poisson}(\lambda_{kn}^{(2)})$$

$$\lambda_{kn}^{(2)} = \ \emptyset_k^{(2)} \theta_{kn}^{(2)} h_{kn}^{(2)}$$

Marginalizing out $z_{kn}^{(2)}$ gives $\mathbf{p}(\boldsymbol{h}_{\boldsymbol{kn}}^{(1)} = \mathbf{1}) = \mathbf{Bernoulli}(\boldsymbol{\pi}_{\boldsymbol{kn}})$, *where* $\boldsymbol{\pi}_{\boldsymbol{kn}} = \mathbf{1} - \mathbf{exp}(-\boldsymbol{\lambda}_{\boldsymbol{kn}}^{(2)})$ **(6)**

where $z_{kn}^{(2)}$ is a latent count for variable $h_{kn}^{(1)}$ parametrized by a Poisson with rate $\lambda_{kn}^{(2)}$ **. 1(.)** = 1 if the argument is true and = 0 otherwise. Rather than using logistic function as in the SBN model, we use definition (6)in this model relaxing the first assumption. This gives us the following deep model:

$$\begin{array}{ll}
\mathbf{x}_n \sim \text{PFA}\left(\mathbf{\Psi}^{(1)}, \boldsymbol{\theta}_n^{(1)}, \mathbf{h}_n^{(1)}; \eta^{(1)}, r_k^{(1)}, b^{(1)}\right), & \mathbf{h}_n^{(1)} = \mathbf{1}\left(\mathbf{z}_n^{(2)}\right), \\
\mathbf{z}_n^{(2)} \sim \text{PFA}\left(\mathbf{\Psi}^{(2)}, \boldsymbol{\theta}_n^{(2)}, \mathbf{h}_n^{(2)}; \eta^{(2)}, r_k^{(2)}, b^{(2)}\right), & \vdots \\
\vdots & \mathbf{h}_n^{(L-1)} = \mathbf{1}\left(\mathbf{z}_n^{(L)}\right), \\
\mathbf{z}_n^{(L)} \sim \text{PFA}\left(\mathbf{\Psi}^{(L)}, \boldsymbol{\theta}_n^{(L)}, \mathbf{h}_n^{(L)}; \eta^{(L)}, r_k^{(L)}, b^{(L)}\right), & \mathbf{h}_n^{(L)} = \mathbf{1}\left(\mathbf{z}_n^{(L+1)}\right),
\end{array}$$

For the top most layer:

$$z_{kn}^{(L+1)} \sim \text{Poisson}(\lambda_k^{(L+1)})$$

$$\lambda_k^{(L+1)} \sim Gamma(a, b)$$

Interpreting the model:

Assume $h_n^{(2)}$ is known. To generate $h_n^{(1)}$ we first $z_n^{(2)}$ with $z_{kn}^{(2)} \sim \text{Poisson}(\lambda_{kn}^{(2)})$ with $\lambda_{kn}^{(2)} = \emptyset_k^{(2)} \theta_{kn}^{(2)} h_{kn}^{(2)}$ where column k of $\emptyset_k^{(2)}$ corresponds to a meta-topic. $\emptyset_k^{(2)}$ is an K1 – dimensional vector denoting probability with which each of layer-1 topics are on when layer-2 "meta-topic" k is on (i.e when $h_{kn}^{(2)}$) = 1). Also, Dirichlet distribution encourages each $\emptyset_k^{(l)}$ to be sparse, thus use of a small subset $\emptyset_k^{(l-1)}$with this repeated all the way down to the data layer. Deep architecture imposes correlation across layer-1 topics and does it through use of PFA modules at all layers unlike SBN Model from section 5 which just uses PFA at the bottom layer.

Advantages of this model:

- It is a deep architecture based entirely on PFA modules and thus readily interpretable throughout all its layers; not just at base layer unlike SBN – Model which uses Gaussian distributed weight matrices within SBN modules which are hard to interpret
- PFA modules can be easily used to build discriminative topic models.
- An efficient MCMC inference procedure developed scales as a function of the number of *non-zeros* in the data and binary units. In contrast, models based on RBMs and SBNs scale with the number of hidden variables in the model.

5. Classification and Inference

For the purpose of classification, if we have document labels provided with the data-set, we would like to train a classifier. The document labels can be document IDs or key-words.

Initially we trained a separate classifier such as SVM or softmax classifier that uses local parameter $\theta_n^{(1)}$, topic distribution for each document as a feature vector against document labels. But the labels for the documents give us a sense that the cluster of documents with a same label are biased against this label. We would like to incorporate this biasing into our model.

Discriminative-DPFA: Encoded labels of each document into the Model proposed in section-5 via shared $\theta_n^{(1)}$ o $h_n^{(1)}$ to learn topics and meta-topics biased towards labels as opposed to just explaining data X_n. We prepare a Y ∈ $\{\mathbf{0},\mathbf{1}\}_+^{\boldsymbol{C\,x\,N}}$ matrix such that-

$\boldsymbol{each\ column\ of\ Y},\ \boldsymbol{y_n}$ **:** 1-hot vector with $\boldsymbol{y_n}$ (c) = 1 where, c : class label of n

$\boldsymbol{prior\ belief}, \boldsymbol{y_n} \sim$ Multinomial (normalized (λ_n)) where, λ_n = **B***$(\boldsymbol{\theta_n^{(1)}}$ **o** $\boldsymbol{h_n^{(1)}})$

B ∈ $\boldsymbol{R}_+^{\boldsymbol{C\,x\,K}}$ and normalized (λ_n) is such that $\lambda_{cn} = \frac{\lambda_{cn}}{\sum_{c=1}^{C} \lambda_{cn}}$

Inference: We used Gibbs sampling which (It is an instance of MCMC) allows us to sample from a distribution that asymptotically follows it without having to explicitly calculate the integrals.

The parameters that constitute a state in Gibbs Sampling which need to be inferred are:

	PFA	PFA+SBN	DPFA with PFA modules in all layers	Disc-DPFA
Global Parameters	$\{ \emptyset_k,$ rk, $\pi_k \}$	$\{ \emptyset_k,$ rk, $, W^{(l)}, c^{(l)} \}$	$\{\emptyset_k^l\}, \{r_k^l\}, \lambda_k^{(L+1)}$	Same as DPFA + b_k, column of B
Local Parameters	$\{\theta_{kn},$ hkn $\}$	$\{\theta_{kn},$ hkn $\}$	$\theta_{kn}^l h_{kn}^l$	Same as DPFA

The posterior inference for the parameters are:

- $\boldsymbol{x}_{mn}$ ~ Mutinomial($\rho_{mn1}, \dots \dots \dots, \rho_{mnK}$), where $\boldsymbol{x}_{mn}$ is an K-sized vector and $\rho_{mnk} = \frac{\emptyset_{mk}\theta_{kn}}{\sum_{k=1}^{K}\emptyset_{mk}\theta_{kn}}$
- $\emptyset k$ ~ Dirichlet (ŋ + $x_{1.k}, \dots \dots \dots$, ŋ + $x_{M.k}$) , where $x_{m.k} = \sum_{n=1}^{N} x_{mnk}$
- θkn ~ Gamma($x_{.nk} + r_k h_{kn}^{(1)}$, $\frac{1\text{-b}}{\text{b}}$), where $x_{.nk} = \sum_{m=1}^{M} x_{mnk}$
- r_k ~ Gamma($\gamma_o + \sum_{n=1}^{N} l_{kn}$, $\frac{1}{c_o + \sum_{n=1}^{N} h_{kn}^{(1)} \ln(1-b)}$)
- π_k ~ Beta(cε $+ \sum_{n=1}^{N} h_{kn}$, $c(1-\epsilon)$ + N - $\sum_{n=1}^{N} h_{kn}$)

For disc-DPFA:

- bk ~Dirichlet (ŋ + $y_{1.k}, \dots \dots \dots$, ŋ + $y_{M.k}$) , where $y_{c.k} = \sum_{n=1}^{N} y_{cnk}$
- θkn ~ Gamma($x_{.nk} + y_{.nk} + r_k h_{kn}^{(1)}$, $\frac{1\text{-b}}{\text{b}}$), where $x_{.nk} = \sum_{m=1}^{M} x_{mnk}$ and $y_{.nk} = \sum_{c=1}^{C} y_{cnk}$

6. Experiment and Results

Models were experimented on 20 Newsgroup corpora. 20 News is composed of 18,845 documents and 2,000 words, partitioned into a 11,315 training set and a 7,531 test set. For the model with single layer number of topics, K were chosen to be 128. For multiple layer models, total number of layers, L =2 was chosen with K2 =64 .

It should be noted that there is no pre-defined way of choosing number of topics. Although there exists non-parameterized approaches like HDP-LDA[3] that lets the model learn number of topics automatically, but we can get much better results by running a few iterations, manually inspecting the topics it produced, deciding whether to increase or decrease the topics.

After running the model proposed in section 5, we choose a meta-topic randomly and inferred top 5 topics from that meta topic using $\emptyset_k^{(2)}$ and for each topic, we obtained top 5 words using $\emptyset_k^{(1)}$. (Tabel1)

For the classification, $\theta_n^{(1)}$ was chosen as a feature vector from all three models separately and SVM, softmax classifier were trained to compare the accuracy (softmax outperformed the SVM). Number of classes were fixed to 20. Finally, disc-PFA was executed with modified posterior inferences. It should be noted that classification results can be further increased by using a separate classifier or by embedding the classifier in the model itself. Here these classifiers are used to show not just the classification results but to compare models on classification scale.

It can be observed that disc-PFA gives better results than normal PFA because of biasing the model. Further improvement in the accuracy is observed when a deep architecture is employed.

Topic 1	Topic 2	Topic 3	Topic 4	Topic 5
Sale	car	Quality	Buy	Card
Offer	dealer	Products	Company	Video
Shipping	bought	High	Price	Drivers
Price	cars	Support	Money	Windows
condition	price	product	sell	monitor

Table 1: Top 5 topics from a meta-topic with top 5 words from each topic

	PFA	Disc-PFA	DPFA
Softmax	53.33%	55.61%	57.28%
SVM	51.51%	54.75%	56.05%

Table 2: Classification of documents with labels using Softmax Classifier and SVM

References

1. Z. Gan, C. Chen, R. Henao, D. Carlson, and L. Carin. Scalable deep Poisson factor analysis for topic modeling. In *ICML*, 2015.

2. Z. Gan, R. Henao, D. Carlson, and L. Carin. Learning deep sigmoid belief networks with data augmentation. In *AISTATS*, 2015

3. Y. W. Teh, M. I. Jordan, M. J. Beal, and D. M. Blei. Hierarchical Dirichlet processes. *JASA*, 2006.

4. Ricardo Henao, Zhe Gan, James Lu and Lawrence Carin. Deep Poisson Factor Modelling. In NIPS, 2016.

5. David M.Blei .Probabilistic Topic Models – General review

6. Mingyuan Zhou , Lauren A. Hannah, David B. Dunson, Lawrence Carin. Beta-Negative Binomial Process and Poisson Factor Analysis

7. David M.Blei, Jon D. McAuliffe. Supervised Topic Models

8. D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent Dirichlet allocation. *JMLR*, 2003