A Comparison of Topic Models: Latent Dirichlet Allocation with Hierarchical Dirichlet Process
Interpretability, Inference and Uncertainty with a Probabilistic Programming Language

Submitted on 20/04/2022

Authors:
Caleb C. Aguida, Mamwouo F. Lyne M. , Messanh K. Guillaume
Supervised by
Nicolas Brunel and Benoit Lebreton

Abstract

This paper describes two models used for topic modeling: the Latent Dirichlet Allocation (LDA) and the Hierarchical Dirichlet Process (HDP). We present an inference method for each model. The LDA model infers parameters and topics using the *variational methods* and an *EM algorithm*, while the HDP model inference in based on the *Chinese Restaurant Franchise (CRF)* and the *Gibbs sampling algorithm*. Our interest is the use of these two models to estimate and to discover topics contained in a text corpora. We have simulated 700 documents from 3 topics that we have fixed, then we have tested the ability of both LDA and HDP models to discover these topics.

Keywords— Latent Dirichlet Allocation, variational inference, EM algorithm, Hierarchical Dirichlet Processes, Chinese restaurant franchise, Gibbs sampling algorithm, document, topic, corpora.

Introduction

In textual data mining, one of the most fashionable techniques is to construct groups of similar documents according to their content (topic) without prior information on the documents. The idea is to have homogeneous clusters of documents (group of documents dealing with the same topic) and heterogeneous from one cluster to another. However, before grouping the documents together, it is necessary to identify the topics covered in each document. One solution to this problem has been proposed by Blei (2004): *Latent Dirichlet Allocation (LDA) model*. The standard version of the LDA is based on the assumption of independence of the topics which is not always verified in practice. Several models have therefore been proposed to take into account the relationship between topics, including Hierarchical Dirichlet Processes (HDPs), which allow to represent a certain correlation structure. In this paper we try to explain how these two models work and to provide some practical experiments. The paper is structured as follows. Section (1) presents some models used before the LDA model. In Section (2), we present an overview of the LDA model. Section (3) introduces the HDP model. Basic technical definitions of Dirichlet Processes (DPs) and HDPs are provided. Stick-breaking and Chinese restaurant representations are presented for DPs and HDPs. Then, we present the assumptions of the HDP model and the inference method based on Chinese restaurant franchise (CRF). We report experimental results in section (4). We simulate a corpus from 3 topics and we illustrate the ability of both LDA and HDP to discover the initial topics that have been fixed.

1 Before Latent Dirichlet Allocation

The problem we are dealing with here is to model text corpora. Several methodologies have been developed over time to solve this problem. We will present some of them in this section.

1.1 TF-IDF model

TF-IDF stands for Term Frequency-Inverse Document Frequency. It is a weighting method that takes into account the frequency of a term (TF) in a given document as well as the number of documents containing this word (IDF). It then makes it possible to distinguish the differentiating elements (here the words) between one document and another. The final result here is a term-by-document matrix X whose columns contain the $tf-idf$ values for each of the documents in the corpus.
Thus, the *tf-idf* scheme reduces documents of arbitrary length to lists of fixed-length numbers. Although the *tf-idf* can identify the words that best discriminate a document in a collection, it only slightly reduces the length of the text. Moreover, this method reveals only few elements on the inter- or intra-document statistical structure. To address these shortcomings, researchers have proposed the latent semantic indexing (LSI).

1.2 LSI model

This model uses the X matrix previously obtained in $tf-idf$. The X matrix is reduced using a singular value decomposition (SVD). The SVD enables us to identify a linear subspace

in the space of $tf - idf$ features that captures most of the variance in the collection. The model works on the basis of a distributional assumption, i.e. it assumes that words with similar meanings will be found in the same type of text.
Thus, this model can capture some notions such as synonymy and polysemy and significantly reduces text length. However, there is no a real explicit generative probabilistic model behind.

1.3 pLSI model

In the probabilistic Latent Semantic Indexing (pLSI), each document is represented as a list of numbers (the mixing proportions for topics), and there is no generative probabilistic model for these numbers. The pLSI models each word in the document as a sample of a mixture models. For each document we calculate the probability of each word coming from that document. Then we calculate the conditional probability of the words in the document.
The main advantage is that we have a more expressive model. However, it does not provide a probabilistic model at the document level and therefore causes serious overfitting problems. Moreover, this model works only on the training sample and it is not clear how to assign probability to a document outside of the training set. The LDA model has been developed to address these shortcomings.

2 Latent Dirichlet Allocation

2.1 Presentation of LDA

Latent Dirichlet Allocation (LDA) [1] is a generative probabilistic model used for unsupervised exploration of sets of observations including textual data. It is one of the most popular methods used in topic modeling.
The main idea is that documents are represented as random mixtures of latent topics. A topic is represented as probabilities on a set of words. The words with the highest probabilities in each topic generally give a good description of the subject dealt with in the topic.
We define the following notations:

- A *vocabulary* is the set of words contained in the corpus. It is indexed by $\{1, \dots, V\}$.
- A *word* is an item of the vocabulary. The v-th word in the vocabulary is represented by a V-vector w such that $w^v = 1$ and $w^u = 0$ for $u \neq v$.
- A *document* is a sequence of N words denoted by $\mathbf{w} = (w_1, w_2, \dots, w_N)$, where w_n is the n-th word in the sequence.
- A *corpus* is a collection of M documents denoted by $\mathbf{D} = \{\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_M\}$.

Let:

- θ represents a vector of proportions of topics for a given document $\mathbf{w}$.
- z_n be a the topic associated with the word w_n in document $\mathbf{w}$. $z_n = (z_n^1, \dots, z_n^k)$ such that $z_n^i = 1$ if w_n belongs to i-th topic, 0 otherwise.
- $\beta = (\beta_{i,j})$ be $k \times V$ matrix where $\beta_{i,j} = p(w^j = 1 | z^i = 1)$
- α be a k-dimensional vector.

LDA assumes the following generative process for each document $\mathbf{w}$ in a corpus $\mathbf{D}$:

1. Choose $N \sim$ poisson(ξ).
2. Choose $\theta \sim$ Dir(α).
3. For each of the N words w_n:
   (a) Choose a topic $z_n \sim$ Multinomial(θ).
   (b) Choose a word w_n from $p(w_n | z_n, \beta)$, a multinomial probability conditioned on the topic z_n.

The LDA is based on some assumptions. First, the number k of the topics is assumed to be known and fixed. Second, it also assumes that documents are exchangeable meaning that the specific ordering of the documents in a corpus can be neglected. Finally, it also supposes that topics and words are infinitely exchangeable within a document. Statistically, for some finite set of $\{z_1, ..., z_N\}$ random variables, **exchangeability assumption** means that their joint distribution is invariant to permutation. If we denote by π a permutation of the integers from 1 to N, then we have:

$$p(z_1, ..., z_N) = p(z_{\pi(1)}, ..., z_{\pi(N)}). \tag{1}$$

An infinite sequence of random variables is *infinitely exchangeable* if every finite subsequence is exchangeable. In this context, de Finetti's representation theorem states that the joint distribution of an infinitely exchangeable sequence of random variables is as if a random parameter (θ in our case) were drawn from some distribution and then, conditioned on that parameter, the random variables in question were *independent and identically distributed* [1]. By this theorem, we can write the probability of a sequence of words and topics as follows:

$$\begin{aligned} p(\mathbf{w}, \mathbf{z}) &= p(\mathbf{w} = (w_1, w_2, ..., w_N), \mathbf{z} = (z_1, z_2, ..., z_N)) \\ &= \int p(\mathbf{w}, \mathbf{z} | \theta) p(\theta) d\theta \\ &= \int p(\theta) \prod_{n=1}^{N} p(w_n, z_n | \theta) d\theta \\ &= \int p(\theta) \left(\prod_{n=1}^{N} p(z_n | \theta) p(w_n | z_n) \right) d\theta. \end{aligned} \tag{2}$$

The graphical representation of LDA is shown in figure (1).

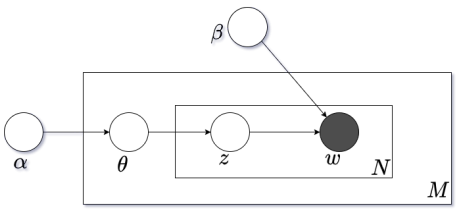

Figure 1: Graphical model representation of LDA. The boxes are "plates" representing replicates. The outer plate represents documents, while the inner plate represents the repeated choice of topics and words within a document.

2.2 Inference and parameter estimation

In the LDA, initially, the latent variables (θ, z) and parameters (α, β) of the model are unknown, and we must try to learn them from the observable data, i.e. the words in the documents.

2.2.1 Inference

The key of the inference is the computation of the posterior distribution of the hidden variables given a document which can be written as:

$$p(\theta, \mathbf{z}|\mathbf{w}, \alpha, \beta) = \frac{p(\theta, \mathbf{z}, \mathbf{w}|\alpha, \beta)}{p(\mathbf{w}|\alpha, \beta)}, \tag{3}$$

where $p(\mathbf{w}|\alpha, \beta)$, by marginalizing out z in equation (2), can be written as:

$$\begin{aligned} p(\mathbf{w}|\alpha, \beta) &= \int p(\theta|\alpha) \left(\prod_{n=1}^{N} \sum_{z_n} p(z_n|\theta) p(w_n|z_n, \beta) \right) d\theta \\ &= \int p(\theta|\alpha) \left(\prod_{n=1}^{N} \sum_{i=1}^{k} \theta_i p(w_n|z_n, \beta) \right) d\theta \\ &= \int p(\theta|\alpha) \left(\prod_{n=1}^{N} \sum_{i=1}^{k} \prod_{j=1}^{V} (\theta_i \beta_{ij})^{w_n^j} \right) d\theta. \end{aligned} \tag{4}$$

The function $p(\mathbf{w}|\alpha, \beta)$ is intractable due to the coupling between θ and β in the summation over latent topics. Therefore, the posterior distribution $p(\theta, \mathbf{z}|\mathbf{w}, \alpha, \beta)$ is intractable to compute in general. However, there are several methods of approximate inference that can be used for LDA [1]. Among these methods the one we are interested in is the variational approximation.

The problematic coupling between θ and β arises due to the edges between θ, $\mathbf{z}$, and $\mathbf{w}$ in figure (1). By dropping these edges and the $\mathbf{w}$ nodes, and endowing the resulting simplified graphical model with free *variational parameters* as shown in figure (2), we obtain a family of distributions on the latent variables. This family is characterized by the following variational distribution:

$$q(\theta, \mathbf{z}|\gamma, \phi) = q(\theta|\gamma) \prod_{n=1}^{N} q(z_n|\phi_n), \tag{5}$$

where the Dirichlet parameter γ and the multinomial parameters $(\phi_1, \ldots, \phi_N)$ are the free variational parameters [1].

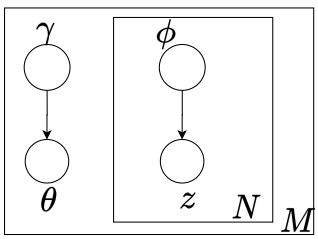

Figure 2: Graphical model representation of the variational distribution used to approximate the posterior in LDA.

The idea of variational inference is to use this variational distribution as a surrogate for the true posterior distribution $p(\theta, \mathbf{z}|\mathbf{w}, \alpha, \beta)$ and to optimize the variational parameters γ and ϕ in order to minimize the Kullback-Leibler (KL) divergence, denoted $D(.||.)$, between these two distributions. Hence, the optimal values of γ and ϕ are given by:

$$(\gamma^\star, \phi^\star) = \arg\min_{(\gamma,\phi)} (D(q(\theta, \mathbf{z}|\gamma, \phi)||p(\theta, \mathbf{z}|\mathbf{w}, \alpha, \beta)) \tag{6}$$

By computing the derivatives of the KL divergence and setting them equal to zero, we obtain the following pair of update equations for γ and ϕ:

$$\phi_{ni} \propto \beta_{iw_n} \exp(E_q[\log(\theta_i)|\gamma]), \tag{7}$$

$$\gamma_i = \alpha_i + \sum_{n=1}^{N} \phi_{ni}, \tag{8}$$

where $E_q[\log(\theta_i)|\gamma] = \Psi(\gamma_i) - \Psi(\sum_{j=1}^{k} \gamma_j)$ and Ψ is the digamma function, the first derivative of the $\log(\Gamma)$ function. Since the two equations (7) and (8) are linked to each other, we can only get $(\gamma^\star, \phi^\star)$ from an iterative method, which is described in algorithm (1):

Algorithm 1: a variational inference algorithm for LDA

```
Initialize φ^0_ni := 1/k for all i and n
Initialize γ_i := α_i + N/k for all i
repeat
    for each n ∈ {1, ..., N} do
        for each i ∈ {1, ..., k} do
            φ^{t+1}_ni := β_{i w_n} exp(ψ(γ^t_i))
        end
        normalize φ^{t+1}_n to sum to 1.
    end
    γ^{t+1} = α + Σ_{n=1}^{N} φ^{t+1}_n
until convergence;
```

It is important to note that the variational parameters $(\gamma^\star, \phi^\star)$ are computed for each document $\mathbf{w}$ because the optimization problem in equation (6) is solved for fixed $\mathbf{w}$.

2.2.2 Parameter estimation

In this section we wish to find the estimations of parameters α and β given a corpus of documents $\mathbf{D}$. For this purpose, we can use an empirical Bayes method which consists in finding parameters α and β that maximize the log likelihood of the data:

$$l(\alpha, \beta) = \sum_{d=1}^{M} \log p(\mathbf{w}_d|\alpha, \beta). \tag{9}$$

As we have explained in section (2.2.1), the quantity $p(\mathbf{w}|\alpha, \beta)$ cannot be computed tractably. However, for a single document, using Jensen's inequality variational inference provides us with a tractable lower bound on the log likelihood verifying the following relationship [1]:

$$\log p(\mathbf{w}|\alpha, \beta) = L(\gamma, \phi; \alpha, \beta) + D(q(\theta, \mathbf{z}|\gamma, \phi)||p(\theta, \mathbf{z}|\mathbf{w}, \alpha, \beta) \tag{10}$$

The overall variational lower bound for the corpus is the sum of the individual variational bounds:

$$L = \sum_{d=1}^{M} L(\gamma_d, \phi_d; \alpha, \beta) \tag{11}$$

Finally, we can thus find approximate empirical Bayes estimates for the LDA via an alternating *variational EM* procedure which we describe in the following algorithm:

1. (E-step): for each document, find the optimizing values of the variational parameters $\{(\gamma_d^\star, \phi_d^\star)\ d \in \mathbf{D}\}$. This is done with algorithm (1).
2. (M-step): maximize the resulting lower bound on the log likelihood $-L-$ with respect to the model parameters α and β. This corresponds to finding maximum likelihood estimates with expected sufficient statistics for each document under the approximate posterior.

These two steps are repeated until the lower bound on the log likelihood converges. To complete the *variational EM* procedure, the M-step update for the conditional β parameter can be written out analytically:

$$\beta_{ij} \propto \sum_{d=1}^{M} \sum_{n=1}^{N_d} \phi_{dni}^\star w_{dn}^j. \tag{12}$$

For the parameter α, taking the derivative of L with respect to α_i gives:

$$\frac{\partial L}{\partial \alpha_i} = M\left(\Psi(\sum_{j=1}^{k}\alpha_j) - \Psi(\alpha_i)\right) + \sum_{d=1}^{M}\left(\Psi(\gamma_{di}) - \Psi(\sum_{j=1}^{k}\gamma_{dj})\right) \tag{13}$$

By setting the equation (13) equal to zero, we see that the value of α_i depends on α_j, where $i \neq j$. Therefore, we must use an iterative method to find the maximal α like Newton-Raphson method:

$$\alpha_{new} = \alpha_{old} - H(\alpha_{old})^{-1} g(\alpha_{old}) \tag{14}$$

where $H(\alpha)$ and $g(\alpha)$ are the Hessian matrix and gradient respectively at the point α. The complete *variational EM* procedure is given by the algorithm (2).

Algorithm 2: a *variational EM* algorithm for LDA

```
Initialize α
Initialize β_ij for all i and j
repeat
    E-step:
    for each document d ∈ {1,...,M} do
        compute (γ*_d, φ*_d) via algorithm (1)
    end
    M-step:
    for each i ∈ {1,...,k} do
        for each j ∈ {1,...,V} do
            for each document d ∈ {1,...,M} do
                for each n ∈ {1,...,N_d} do
                    β_ij += φ*_dni w^j_dn.
                end
            end
        end
    end
    estimate α via (13)
until L converges;
```

We have seen that we have to choose the number k of topics to be discovered in documents in the LDA. This hypothesis is binding because we do not actually know this number k for a given corpus a priori. To overcome this problem, we present in the following section the Hierarchical Dirichlet Process (HDP).

3 Sharing Clusters Among Related Groups: Hierarchical Dirichlet Processes

Hierarchical Dirichlet Process (HDP) is a nonparametric Bayesian model for clustering problems involving multiple groups of data. The goal is discovering topics that are common across multiple documents in the same corpus, as well as across multiple corpora. It allows the number of components (clusters) to be open-ended and inferred automatically by the model. It also allows components to be shared across groups, allowing dependencies across groups to be modeled effectively as well as conferring generalization to new groups. Compared to LDA, here:

1. we do not need to know *a priori* the number of clusters.
2. we do not know how clusters should be shared among groups.

The first point (1) necessitates the replacement of the Dirichlet distribution by the Dirichlet Process (DP). In other words, we associate a DP to each group. But this causes another problem: with this configuration, clusters could not be shared among groups (because for different DPs we have different atoms). So, we have to link DPs together. A first solution is to use common base measure G_0 for groups, but this does not solve the problem because it is often assumed that G_0 is a smooth distribution. A second approach is the **hierarchical Bayesian approach** that allows G_0 be a DP itself [2]. It is this solution that we present in this paper, and we begin in the next section by providing a Dirichlet process notion.

3.1 Dirichlet process

Let $(\Theta, \mathcal{B})$ be a measurable space and G_0 (**the base measure**) be a probability measure on that space. Let $A = (A_1, A_2.., A_r)$ be a finite partition of Θ and α_0 be a positive real number (**concentration parameter**).
We say that $G \sim DP(\alpha_0, G_0)$ if for any A:

$$(G(A_1), ..., G(A_r)) \sim Dir(\alpha_0 G_0(A_1), ..., \alpha_0 G_0(A_r)) \tag{15}$$

DP can be used to classify data in clusters. Considering a set of data $\mathbf{x} = (x_1, \ldots, x_n)$ assumed exchangeable, the clustering process is:

- $G \sim DP(\alpha_0, G_0)$,
- $\phi_i | G \overset{i.i.d.}{\sim} G$,
- $x_i | \phi_i \sim F(\phi_i) = F(.|\phi_i)$ for each i,

where ϕ_i is a latent factor which determines the cluster to which the observation x_i is assigned. This setup is referred to as a *DP mixture model*, which is illustrated in figure (3). The main advantage of using this model is the **clustering property of DP**. Indeed, the draws ϕ_i are not generally distinct, meaning that k (the random number of mixture components) is less than n and its expectation $-\mathbb{E}(k)-$ grows as $O(\log n)$.

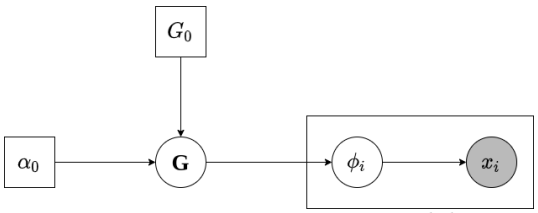

Figure 3: DP mixture model

There are several ways to draw a DP. The Chinese restaurant process (CRP) and the stick-breaking process are the two most commonly used representations of the Dirichlet process.

3.2 Stick-breaking construction of Dirichlet process

The stick-breaking representation is one of the fundamental properties of the Dirichlet process. It represents the random probability measure as a discrete random sum whose weights and atoms are formed by independent and identically distributed sequences of beta variates and draws from the normalized base measure of the Dirichlet process parameter. It is widely used in posterior simulation for statistical models with Dirichlet processes [3].
Indeed, if $G \sim DP(\alpha_0, G_0)$ then its draw is discrete with probability one [4], and for $A \in \mathcal{B}$, we write:

$$G(A) = \sum_{k=1}^{\infty} \pi_k \delta_{\theta_k}(A), \tag{16}$$

where

$$\pi_k = \beta_k \prod_{l=1}^{k-1} (1 - \beta_l), \tag{17}$$

and $\beta_k \sim beta(1, \alpha_0)$, $\theta_k \sim G_0$ and $\sum_{k=1}^{\infty} \pi_k = 1$.
The stick-breaking construction is equivalent to dividing a stick of length 1 into successive pieces. At each step, we break off segments of length π_k. This process is shown by figure (4).

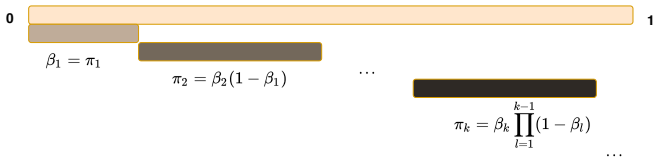

Figure 4: Stick-breaking construction of Dirichlet process

This stick-breaking construction shows that DP mixture models can be viewed as mixture models with a countably infinite number of components where θ_k is the parameter of the k^{th} mixture component and β_k, $k = 1, 2, \ldots$ are the mixing proportions.

3.3 Chinese Restaurant Process

The Chinese Restaurant Process (CRP) helps to exhibit **clustering property** that is: *it is most probably that a new customer chooses to sit at the table with more customers than the others.*
Formally, let:

- $G \sim DP(\alpha_0, G_0)$.
- $\phi_1, \ldots \phi_i, \ldots \overset{i.i.d.}{\sim} G$.
- $\theta_1, \ldots, \theta_K$ be the distinct values taken on by $\phi_1, \ldots \phi_{i-1}$.
- n_k be the number of $\phi_{i'} = \theta_k$. for $1 \leq i' < i$; $k \leq K$.

The conditional distribution of ϕ_i given $\phi_1, \ldots \phi_{i-1}$, α_0 and G_0 have the following form:

$$\phi_i | \phi_1, \ldots, \phi_{i-1}, \alpha_0, G_0 \sim \sum_{k=1}^{K} \frac{n_k}{i - 1 + \alpha_0} \delta_{\theta_k} + \frac{\alpha_0}{i - 1 + \alpha_0} G_0 \tag{18}$$

The figure (5) illustrates the CRP.

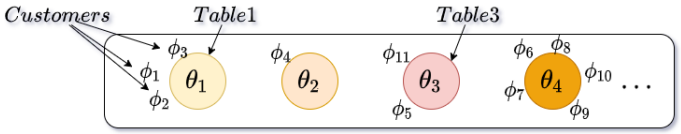

Figure 5: Chinese restaurant process illustration

The idea of CRP is that: let's consider a Chinese restaurant with an unbounded number of tables. Each ϕ_i corresponds to a customer who enters the restaurant, while the distinct values θ_k correspond to the tables at which the customers sit. The i-th customer sits at the table indexed by θ_k, with probability proportional to the number of customers n_k already seated there (in that case we set $\phi_i = \theta_k$), or sits at a new table with probability proportional to α_0 (in that case, we increment K by one, draw $\theta_k \sim G_0$ and set $\phi_{i'} = \theta_k$).
Let us now focus on HDP which is based on the DPs.

3.4 HDP definition and assumptions

The Hierarchical Dirichlet Process (HDP) is a nonparametric Bayesian approach to modeling grouped data. It is a distribution over a set of random probability measures over $(\Theta, \mathcal{B})$: one probability measure G_j for each group j, and a global probability measure G_0.
To ensure that mixture components are shared across different groups, G_0 should be **discrete** (atomic) and be the base distribution for all G_j. Statistical strength is shared across groups when they share the same set of mixture components. This property allows generalization to new groups.
The HDP model is written as follows:

- $G_0 | \gamma, H \sim DP(\gamma, H)$,
- $G_j | \alpha_0, G_0 \overset{i.i.d.}{\sim} DP(\alpha_0, G_0)$ for each j,

where H is the base measure of G_0, γ and α_0 are the concentration parameters.
The HDP supposes that we have J groups of data, each consisting of n_j data points $(x_{j1}, \ldots, x_{jn_j})$. The data points in each group j are exchangeable, and are to be modeled with a mixture model.

3.5 HDP mixture model

In this section we define a mixture version model for HDP.
In addition to the formulas provided in section (3.4), if we associate each observation $x_{j,i}$ with a factor $\phi_{j,i}$; let $F(\phi_{j,i})$ and G_j denote the distributions of $x_{j,i}$ (given the factor $\phi_{j,i}$) and $\phi_{j,i}$ respectively, the *HDP mixture model* is given by:

- $G_0|\gamma, H \sim DP(\gamma, H)$,
- $G_j|\alpha_0, G_0 \overset{i.i.d.}{\sim} DP(\alpha_0, G_0)$ for each j,
- $\phi_{j,i}|G_j \overset{i.i.d.}{\sim} G_j$ for each j and i,
- $x_{j,i}|\phi_{j,i} \overset{i.i.d.}{\sim} F(\phi_{j,i}) = F(.|\phi_{j,i})$ for each j and i.

In this setup, each group j is modeled as DP mixture model. Moreover, according to HDP stick-breaking representation the groups share the same set of mixture components $(\theta_k)_{k=1}^{\infty}$, but with different mixing weights $\pi_j = (\pi_{jk})_{k=1}^{\infty}$. Figure (6) gives an example of graphical representation of HDP mixture model.

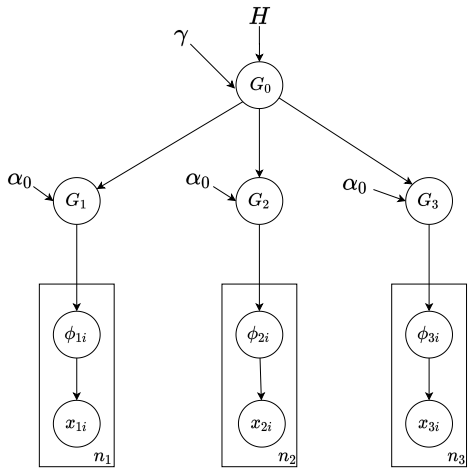

Figure 6: An example of HDP mixture model with 3 groups

As we have previously seen in the case of DP, it is possible to provide a stick-breaking construction and an CRP analogue for HDPs.

3.6 HDP stick-breaking representation

Since G_0 is distributed as a Dirichlet process ($G_0|\gamma, H \sim DP(\gamma, H)$), it can be expressed using a stick-breaking representation:

$$G_0(A) = \sum_{k=1}^{\infty} \beta_k \delta_{\theta_k}(A), \tag{19}$$

where $\theta = (\theta_k)_{k=1}^{\infty}$ are the atoms of G_0 and $\beta = (\beta_k)_{k=1}^{\infty}$ their global weights.

Since G_0 is the base distribution for all G_j, the atoms of the individual G_j are sampled from G_0 (the atoms of G_j must also come from θ), and then we can write:

$$G_j(A) = \sum_{k=1}^{\infty} \pi_{jk} \delta_{\theta_k}(A) \quad \forall j = 1, \ldots, J, \tag{20}$$

where $\pi_j = (\pi_{j,k})_{k=1}^{\infty}$ are the weights of θ specific to the group j.
We see then that the different groups share the same set of mixture components θ, but with mixing proportions π_j specific to each group.
Moreover, we can derive an explicit relationship between β and π_j. For this, let $(A_1, ..., A_r)$ be a measurable partition of Θ and $K_l = \{\, k : \theta_k \in A_l\}$ with $l = 1, \ldots, r$. For each j we have:

$$\begin{aligned}(G_j(A_1), \ldots, G_j(A_r)) \quad &\sim \quad Dir(\alpha_0 G_0(A_1), \ldots, \alpha_0 G_0(A_r)), \\ \Rightarrow \left(\sum_{k\in K_1} \pi_{jk}, \ldots, \sum_{k\in K_r} \pi_{jk}\right) \quad &\sim \quad Dir\left(\alpha_0 \sum_{k\in K_1} \beta_k, \right. \\ &\left. \ldots, \alpha_0 \sum_{k\in K_r} \beta_k\right), \end{aligned} \tag{21}$$

for any finite partition of the positive integers $(K_1, \ldots, K_r)$. Hence π_j is distributed according to $DP(\alpha_0, \beta)$. On the other hand, since the G_j are independent given G_0, the weights π_j are independent given β. By combining the two points, we write: $\pi_j = (\pi_{jk})_{k=1}^{\infty} \overset{i.i.d.}{\sim} DP(\alpha_0, \beta)$.
For a partition $(\{1, \ldots, k-1\}, \{k\}, \{k+1, k+2, \ldots\})$ of positive integers, (21) gives:

$$\begin{aligned}\left(\sum_{l=1}^{k-1} \pi_{jl}, \pi_{jk}, \sum_{l=k+1}^{\infty} \pi_{jl}\right) \quad &\sim \quad Dir\left(\alpha_0 \sum_{l=1}^{k-1} \beta_l, \right. \\ &\left. \alpha_0 \beta_k, \alpha_0 \sum_{l=k+1}^{\infty} \beta_l\right). \end{aligned} \tag{22}$$

By removing the first element, and using standard properties of the Dirichlet distribution, we have:

$$\frac{1}{1 - \sum_{l=1}^{k-1} \pi_{jl}} \left(\pi_{jk}, \sum_{l=k+1}^{\infty} \pi_{jl}\right) \sim Dir\left(\alpha_0 \beta_k, \alpha_0 \sum_{l=k+1}^{\infty} \beta_l\right). \tag{23}$$

Finally, if we define:

$$\pi'_{jk} = \frac{\pi_{jk}}{1 - \sum_{l=1}^{k-1} \pi_{jl}} \quad \text{and observe that} \quad \sum_{l=k+1}^{\infty} \beta_l = 1 - \sum_{l=1}^{k} \beta_l,$$

we have:

$$\pi'_{jk} \sim Beta\left(\alpha_0 \beta_k, \alpha_0 \left(1 - \sum_{l=1}^{k} \beta_l\right)\right) \text{ and}$$

$$\pi_{jk} = \pi'_{jk} \prod_{l=1}^{k-1} (1 - \pi'_{jl}). \tag{24}$$

We recall that, according to equation (19), β_k verifies the following relation:

$$\beta_k = \beta_k^{'} \prod_{l=1}^{k-1}(1-\beta_l^{'}) \quad and \quad \beta_k^{'} \quad \sim \quad beta(1,\gamma). \qquad (25)$$

The above demonstration in adapted from [5].

3.7 Chinese Restaurant Franchise

The Chinese Restaurant Franchise (CRF) is an analog of the Chinese restaurant process for hierarchical Dirichlet processes. It is a generalization of Chinese restaurant process for J restaurants which share the same set of dishes θ_k. In the CRF, we assume that each restaurant j can accommodate an infinite number of customers and set up an unbounded number of tables. Also, multiple tables at multiple restaurants can serve the same dish.
The CRF is essentially a two-level Chinese Restaurant Process. First, within a restaurant customers choose tables, and then dishes are assigned to these specific tables among all restaurants.
Formally, let:

- $G_0 \sim DP(\gamma, H)$.
- $G_j \overset{i.i.d.}{\sim} DP(\alpha_0, G_0)$.
- $\phi_{j,1}, \ldots, \phi_{j,i}, \ldots \overset{i.i.d.}{\sim} G_j$, where $\phi_{j,i}$ denote the i-th customer in restaurant j.
- $\psi_{j,1}, \ldots, \psi_{j,T_j} \overset{i.i.d.}{\sim} G_0$ be the distinct values taken by each $\phi_{j,i}$. $\psi_{j,t}$ denote the table t in restaurant j where the i-th customer sit down.
- $n_{j,t}$ be the number of $\phi_{j,i'} = \psi_{j,t}$ for $1 \leq t \leq T_j$ and $1 \leq i' < i$. It is the number of customers seated at the table t among the $i-1$ customers in the restaurant j.
- $\theta_1, \ldots, \theta_K \overset{i.i.d.}{\sim} H$ be distinct values taken by each $\psi_{j,t}$.
- m_k be the number of $\psi_{j,t} = \theta_k$, for all j, i.e. the number of tables serving dish θ_k among the whole the franchise.
- $t_{j,i}$ be the index of the $\psi_{j,t}$ associated with $\phi_{j,i}$.
- $k_{j,t}$ be the index of θ_k associated with $\psi_{j,t}$.

According to equation (18), the process of table choosing by the i-th customer in restaurant j given its current seating plan is:

$$\phi_{j,i}|\phi_{j,1}, \ldots, \phi_{j,i-1}, \alpha_0, G_0 \sim \sum_{t=1}^{T_j} \frac{n_{j,t}}{i-1+\alpha_0}\delta_{\psi_{j,t}} + \frac{\alpha_0}{i-1+\alpha_0}G_0, \qquad (26)$$

where T_j is the number of occupied tables before the i-th customer comes into restaurant j.
The equation (26) means that when the i-th customer comes into restaurant j, he can choose to sit at an occupied table t with probability proportional to $n_{j,t}$ or a new table T_j+1 with probability proportional to α_0. In the first case, we set $\phi_{j,i} = \psi_{j,t}$ and let $t_{j,i} = t$ for the chosen t. In the second case, we increment T_j by one, draw a new sample $\psi_{j,T_j} \sim G_0$ and set $\phi_{j,i} = \psi_{j,T_j}$ and $t_{j,i} = T_j$. The figure (7) illustrates the tables choosing process in 3 restaurants.

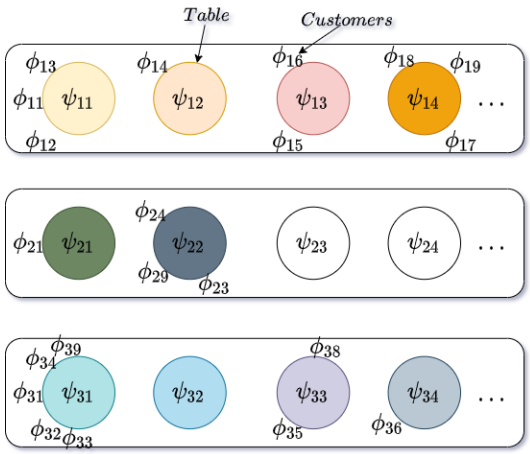

Figure 7: Tables choosing process in CRF with 3 restaurants.

Once the table has been chosen, a dish must be assigned to the customer. In the CRF, we suppose that customers are sociable. If the customer sits at an occupied table, he shares the dish θ_k that has been ordered at that table. If he sits at a new table, he orders a dish for that table according to its popularity among the whole franchise (with probability proportional to m_k), while a new dish can also be tried (with probability proportional to γ). In the first case, we set $\psi_{j,t} = \theta_k$ and let $k_{j,t} = k$ for the chosen k. In the second case, we increment K by one, draw a new sample $\theta_K \sim H$ and set $\psi_{j,t} = \theta_K$, $k_{j,t} = K$. This process of dishes choosing is given, according to equation (18), by equation (27) and illustrated in figure (8).

$$\psi_{j,t}|\psi_{1,1}, \psi_{1,2}, \ldots, \psi_{2,1}, \ldots, \psi_{j,t-1}, \gamma, H \sim \sum_{k=1}^{K} \frac{m_k}{\sum_k m_k + \gamma}\delta_{\theta_k} + \frac{\gamma}{\sum_k m_k + \gamma}H. \qquad (27)$$

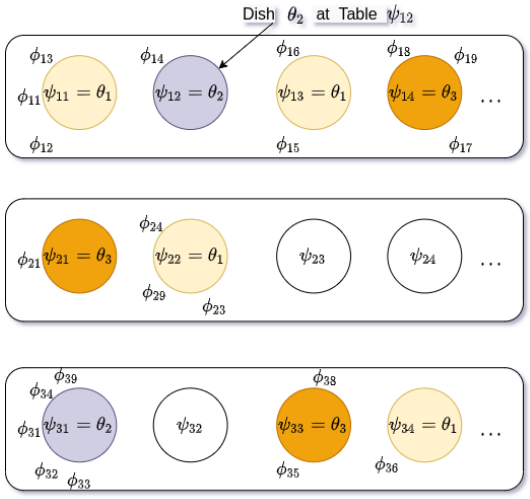

Figure 8: Dishes choosing process in CRF with 3 restaurants.

Now that how the Chinese restaurant franchise works is ex-

plained, we will use it to infer the parameters $\phi_{j,i}$ in the HDP mixture model. This is done in the following section.

3.8 HDP Inference

In the HDP mixture model, we assume that each observation $x_{j,i}$ comes from a distribution $F(.|\phi_{j,i})$ where $\phi_{j,i}$ is the parameter of the distribution. The goal of the inference is to produce **samples** of $\phi_{j,i}$, as well as intermediary samples of $\psi_{j,t}$ and θ_k from their posterior distributions given the observations. For this purpose, we will use a **Gibbs sampling** based on the CRF. The latter, as described in section (3.7), gives the process of assigning each $\phi_{j,i}$ to one of the global parameters θ_k.
To make the Gibbs sampling more efficient, rather than dealing with the $\phi_{j,i}$'s and $\psi_{j,t}$'s directly, we shall sample their index[1] variables t_{ji} and k_{jt} as well as the distinct values θ_k [5]. Furthermore, the exchangeability properties of $\phi_{j,i}$, $\psi_{j,t}$ extends to t_{ji} and k_{jt} and we can rewrite equations (26) and (27) as follows:

$$t_{j,i}|t_{j,1},\dots,t_{j,i-1},\alpha_0 \sim \sum_{t=1}^{T_j} \frac{n_{j,t}}{i-1+\alpha_0}\delta_t + \frac{\alpha_0}{i-1+\alpha_0}\delta_{t^{new}}, \tag{28}$$

$$k_{j,t}|k_{1,1},k_{1,2},\dots,k_{2,1},\dots,k_{j,t-1},\gamma \sim \sum_{k=1}^{K} \frac{m_k}{\sum_k m_k+\gamma}\delta_k + \frac{\gamma}{\sum_k m_k+\gamma}\delta_{k^{new}}, \tag{29}$$

with $\psi_{j,t^{new}} \sim G_0$ and $\theta_{k^{new}} \sim H$.
Before exposing the Gibbs sampling process, let us recall the various variables and quantities of interest. Let:

- $t=(t_{j,i})$, $k=(k_{j,t})$, $\theta=(\theta_k)$ and data items $x=(x_1,\dots,x_J)$ where $x_j=(x_1,\dots,x_{n_j})$.
- $f(\cdot|\phi_{j,i})$ and $h(.)$ be the density functions for $F(\cdot|\phi_{j,i})$ and H respectively.
- $n_{j,t}^{-i}$ be the number of t_{ji}'s equal to t except $t_{j,i}$, and $m_k^{-j,t}$ the number of $k_{j',t'}$'s equal to k except $k_{j,t}$.

According to Bayes theorem, the posterior probability for $t_{j,i}$ given the other variables is proportional to the product of its prior distribution and the likelihood term. This reasoning is the same for $k_{j,t}$ and for θ_k.
Sampling of t:
The prior distribution of $t_{j,i}$ is given by (28). The likelihood is given by $f(x_{j,i}|\theta_{k_{j,t}})$ where for $t=t^{new}$ we may sample $k_{j,t^{new}}$ using (29). With this information, the posterior conditional distribution of $t_{j,i}$ is then:

$$p(t_{j,i}=t|t\backslash t_{j,i},k,\theta,x) \propto \begin{cases} \alpha_0 f(x_{j,i}|\theta_{k_{j,t}}) & \text{if } t=t^{new} \\ n_{j,t}^{-i} f(x_{j,i}|\theta_{k_{j,t}}) & \text{if current } t \end{cases} \tag{30}$$

Sampling of k:
The prior distribution of $k_{j,i}$ is given by (29). The likelihood term is given by $\prod_{i:t_{j,i}=t} f(x_{j,i}|\theta_k)$. Then, the posterior conditional distribution of $k_{j,i}$ is:

$$p(k_{j,i}=k|t,k\backslash k_{j,i},\theta,x) \propto \begin{cases} \gamma \prod_{i:t_{j,i}=t} f(x_{j,i}|\theta_k) \text{ if } k=t^{new} \\ m_k^{-t} \prod_{i:t_{j,i}=t} f(x_{j,i}|\theta_k) \text{ if cur. } k \end{cases} \tag{31}$$

Sampling of θ_k:
The prior distribution of θ_k is given by $h(\theta_k)$. The likelihood is given by $\prod_{j,i:k_{j,t_{j,i}}=k} f(x_{j,i}|\theta_k)$. Hence, the posterior conditional distribution of θ_k is:

$$p(\theta_k|\mathbf{t},\mathbf{k},\theta\backslash\theta_k,x) \propto h(\theta_k) \prod_{j,i:k_{j,t_{j,i}}=k} f(x_{j,i}|\theta_k) \tag{32}$$

The algorithm (3) describes the Gibbs sampling associated with previous samplings.

Algorithm 3: Gibbs sampling in the CRF

```
Initialization of (t, k, θ)
repeat
    for each j do
        for each i do
            t_{j,i} ~ p(t_{j,i} = t | t\t_{j,i}, k, θ, x)
            k_{j,i} ~ p(k_{j,i} = k | t, k\k_{j,i}, θ, x)
            θ_k ~ p(θ_k | t, k, θ\θ_k, x)
        end
    end
until convergence;
```

4 Uncertainty estimation of the decisions of the LDA and HDP

To illustrate the methods we defined earlier we present our experiments in this section. The goal is to study the ability of the LDA and the HDP to infer parameters and topics that we have fixed. We also compare the results of these two models.
We have first simulated 700 documents according to the generative process assumed by LDA. Indeed, we constructed 3 topics which constitute our vocabulary of 46 words. We fixed $\alpha=[1,1,1]$ and $\xi=60$ and we computed the β matrix. The table (1) shows the words contained in each topic.

[1] The $\phi_{j,i}$'s and $\psi_{j,t}$'s can be reconstructed from these index variables and the θ_k

Topics	Words
Topic 1	in; probability; theory; dirichlet; processes; after; peter; gustav; lejeune; dirichlet; are; a; family; of; stochastic; processes; whose; realizations; are; probability; distributions
Topic 2	leads; in; individual; states; may; change; from; one; party; to; another; as; all; the; votes; are; counted; select; a; state
Topic 3	bankpolicies; creates; bank; policy; job; description; and; form; templates; designed; to; make; your; job; easier

Table 1: Topics from which corpus is generated

We fitted both LDA and HDP on the simulated documents to infer the different topics in the corpus. In the LDA estimation, we explicitly fixed the number k of topics to discover ($k = 3$). After 970 000 iterations of algorithm (2), we got the results of the LDA estimation shown in table (2).

Topics	$\hat{\alpha}$	Words
Topic 1	0.92	probability; are; dirichlet; processes; lejeune; family; after; of; in; realizations; distributions; whose stochastic; theory; gustav
Topic 2	0.97	change; one; states; all; counted; are; state; in; individual; leads; may; from; another; the; as
Topic 3	1	job; to; bankpolicies; designed; easier; policy; make; bank; and; creates; your; form; description; templates; dirichlet

Table 2: Topics and α parameter inferred by the LDA. Here we show the top 15 most frequent words for each topic

The estimated value for the parameter α, $\hat{\alpha} = [0.92, 0.97, 1]-$ is close to its the true value ($\alpha = [1, 1, 1]$). In addition, the 15 most frequent words of each topic inferred by the LDA match with the real words contained in the fixed topics except the word "dirichlet" of topic 3 which is not contained in the initial topic 3. Hence, we note that LDA found successfully the initial fixed topics.

The results obtained with the HDP are shown in table (3). This model well infers automatically the number of topics to discover ($k = 3$) but has some difficulties to infer the initial fixed topics.

Topics	Words
Topic 1	are; job; to; in; dirichlet; probability; processes; and; all; after
Topic 2	job; to; processes; probability; in; dirichlet; policy; make; are; templates
Topic 3	in; to; job; templates; state; probability; states; the; make; individual

Table 3: Topics inferred by the HDP. Here we show the top 10 most frequent words for each topic.

In our experiments, we fit the models with TensorFlow Probability, a probabilistic programming language. Our code is available at [6]. This code is inspired by work done by [7], [8] and [9].

Discussion

In this paper, we explored the LDA and the HDP models. We have seen that the LDA allows us to build probabilistic classifications with a fairly good accuracy . However, the choice to infer with variational inference can leads to some problems. Indeed, due to approximations, there may remains a gap between the variational posterior and the true posterior distribution, inherent to algorithm design. This gap will not vanish when the number of samples and the number of iterations increase. Moreover the variational inference uses another optimization method that is Newton Raphson which has its own limitations. Indeed, we are not always sure that the value of α returned by algorithm is the global solution because we do not have more information about the concavity of the objective function in equation (11) used for the optimization. In this case, the initial value of α can impact the solution and lead us to a local maximum. A solution can be the Gibbs sampling approach. Furthermore, the variational inference requires a lot of optimization iterations.
Regarding the HDP, the assumption (3.3) (clustering property) may be unrealistic. A new customer will tend to choose a new table if he has no relationship with customers already seated in the restaurant.
Finally, in our experiments, we have seen that the HDP is less accuracy than LDA. This result may depend on the parameters we set during the estimation step. It can be interesting to fit again the HDP with other parameters.

Glossary

Algorithm: is a finite sequence of instructions used to solve a problem in machine learning.
Assumption: statement that is accepted as true without proof.
Atom: a set $G \in \Theta$ will be called an atom for a measure μ if $\mu(E)$ is strictly positive and $\forall F \in \Theta$, $\mu(E \cap F)$ is 0.
Clustering: is a statistical method that consists of grouping data points into homogeneous groups called clusters by similarity or distance.
Distribution: a function that shows the possible values for a variable and how often they occur.
Draw: random selection of components.
Exchangeability: is the characteristic of a sequence of random variables whose joint probability distribution does not change when the positions in the sequence, in which finitely many of them appear, are altered.
Generalization: involves inferring the results from a sample and applying it to a population.
Grouped data: data that has been bundled together in groups.
Inference: is the process of using data analysis to deduce properties of an underlying distribution of probability.
Likelihood: function that measures the goodness of fit of a statistical model to a sample of data for given values of the unknown parameters.
Measurable space: is a pair (Ω, S) consisting of a set Ω and a $\sigma - algebra$ S of subsets of Ω.
Mixture model: a probabilistic model for representing the presence of subpopulations within an overall population, without requiring that an observed data set should identify the sub-population to which an individual observation belongs.
Model: a relationship between some variables and a dependent variable that we try to explain.
Non-parametric model: models in that the model structure is not specified a priori but is instead determined from data. They are statistical models that do not often conform to a normal distribution.
Parameter: is a quantity entering into the probability distribution of a statistic or a random variable.
Posterior distribution: distribution of an unknown quantity, treated as a random variable, conditional on the evidence obtained from an experiment or survey.
Prior distribution: it is the belief about the true value of a parameter. It is the "best guess".
Probabilistic model: is a model based on the theory of probability or the fact that randomness plays a role in predicting future events.
Probability: is the measure of the likelihood that an event will occur in a Random Experiment.
Sampling: a process or method of drawing a representative group of individuals or cases from a particular population.
Singular Value Decomposition (SVD): is a factorization of a real or complex matrix that generalizes the eigen decomposition of a square normal matrix to any.
Statistical strength: the strength of the relation between the two groups.
Text corpora: a large and structured set of texts.
Topic: a matter dealt with in a text.

References

[1] D. M. Blei, A. Y. Ng, and M. I. Jordan, " Latent Dirichlet Allocation," Journal of Machine Learning Research 3 (2003) 993-1022.

[2] Y. W. Teh, M. I. Jordan, M. J. Beal, and D. M. Blei, "Sharing Clusters Among Related Groups: Hierarchical Dirichlet Processes,"

[3] J. Lee and S. N. MacEachern, "A new proof of the stick-breaking representation of Dirichlet processes,"

[4] T. S. FERGUSON, " A Bayesian analysis of some nonparametric problems.The Annals of Statistics," Mar. 1973, pp. 209–230.

[5] Y. W. Teh, M. I. Jordan, M. J. Beal, and D. M. Blei, "Hierarchical Dirichlet Processes," Oct. 2004.

[6] [Online]. Available: `https://github.com/Carloss1998-lab/Projet_Methodo_LDA_HDP.git`.

[7] D. Piponi, D. Moore, and J. V. Dillon, "Joint Distributions for TensorFlow Probability," Jan. 2020, pp. 1–10.

[8] [Online]. Available: `https://github.com/tensorflow/probability/blob/4aa1ee652853a19c4e80d39216c3fa535ed3e589/tensorflow_probability/examples/latent_dirichlet_allocation_distributions.py`.

[9] [Online]. Available: `https://github.com/linkstrife/HDP.git`.