# Document Modeling using the Hierarchical Dirichlet Process

Jingjing Fan, Qiaohui Lin

May 2019

Note: This project is a replication of the "Hierarchical Dirichlet Processes" paper by Yee Whye Teh, Nichael I Jordan, Matthew J. Beal, and David M. Blei. See end of document for full reference.

## 1 Introduction

The separation of data into distinct groups is a common problem in statistics. In certain circumstances, it is desirable for these clusters to remain statistically linked in some way. Hierarchical models are a natural approach for capturing relationships between groups. In this project, we will apply the hierarchical Dirichlet process to the problem of topic modeling.

The goal of topic modeling is to cluster the words in a document according to their respective topics. The problem can be further extended to include multiple documents, called a corpus, or even multiple corpuses of documents. We treat each document as a 'bag of words', disregarding the ordering of the words. Each word is assumed to be a draw from a multinomial distribution whose measure is the topic. In this framework, the document can be treated as a mixture of topics. Mathematically, a 'topic' is a discrete measure which assigns some probability to each possible word in the vocabulary. Once the words are clustered, each cluster will be described using a probability measure that ideally assigns high weight to the words within each cluster. For our experiment, we choose the word with the highest probability in each topic as the topic label for that cluster.

Intuitively, we expect that different documents in the corpus can share topics. For example, a journal on economics may contain an article about the housing market and an article about the viability of Bitcoin in the same issue. For the housing article, we would like to potentially separate the words into topics like "homes", "mortgage", "demand", etc. While the Bitcoin article has nothing to do with the physical sale of property, it could still contain overlapping topics like "demand". Thus, it is in our interest to create a modeling framework in which it is possible for topics within each document to bear similarities, and clusters between documents to be able to share topics as well.

These two requirements motivate the need for a hierarchical Dirichlet process. A Dirichlet process, $\mathrm{DP}(\alpha_0, G_0)$, is a measure on measures. We can make it possible for clusters within a document to share topics by letting the topics be draws from a Dirichlet process with base distribution $G_0 \sim Dirichlet(\alpha_1, \dots, \alpha_k)$ and some scaling parameter $\alpha_0 > 0$. To allow different documents to share topics, we let $G_0$ itself be a draw from a Dirichlet process.

## 2 Dirichlet Process

Let $(\Theta, \mathcal{B})$ be a measurable space with probability measure $G_0$. Let $\alpha_0$ be a positive real number. The Dirichlet Process, $DP(\alpha_0, G_0)$, is defined as the distribution of a probability measure $G_j$ over $(\Theta, \mathcal{B})$ such that for any finite measurable partition $(A_1, A_2, \dots, A_r)$ of $\Theta$, the random vector $(G_j(A_1), G_j(A_2), \dots, G_j(A_r))$ is distributed as a finite-dimensional Dirichlet Distribution with parameters $(\alpha_0 G_0(A_1), \alpha_0 G_0(A_2), \dots, \alpha_0 G_0(A_r))$

$$(G_j(A_1), G_j(A_2), \dots, G_j(A_r)) \sim Dir(\alpha_0 G_0(A_1), \alpha_0 G_0(A_2), \dots, \alpha_0 G_0(A_r)) \tag{1}$$

We write $G_j \sim DP(\alpha_0, G_0)$ if $G_j$ is a random probability measure with distribution given by the Dirichlet process with base distribution $G_0$ and scaling parameter $\alpha_0$. We do not use the 'stick-breaking' method for our sampling scheme but it provides the most intuitive understanding for the construction of a DP. In stick-breaking, a proper discrete probability measure $\boldsymbol{\pi}$ is generated using the $GEM(\alpha_0)$ process. Each component of $\boldsymbol{\pi}$ corresponds to a value, $\psi_k$, drawn from the base distribution $G_0$. The result is a discrete distribution in which the probability of drawing $\psi_k$ is $\pi_k$.

In the topic modeling setting, $G_0$ is a Dirichlet distribution and each draw from $G_0$ is a potential topic probability vector. The value of each draw is $\psi_k$ where $\psi_k \sim G_0$. Let $\theta_i$ be a draw from $G_j$. Then $\theta_i = \psi_k$ with probability $\pi_k$. In our topic modeling mixture model setting, we have on the document level,

$$\theta_i | G_j \sim G_j \tag{2}$$

$$x_i | \theta_i \sim Multinomial(\theta_i). \tag{3}$$

# 3 The Chinese Restaurant Process

The Chinese Restaurant Process is an equivalent representation of the DP in which we marginalize out $G_j$ and focus only the probability of $\theta_i = \psi_i | \theta_1, \ldots \theta_{i-1}, \alpha_0, G_0$. We will only present the formula for the conditional probability here. See the Appendix for the derivation of this formula.

Let $\psi_1, \ldots, \psi_K$ be the distinct values taken on by some observed $\theta_1, \ldots \theta_{i-1}$, and let $n_k$ be the count of $\theta$'s that are equal to a particular $\psi_k$. The conditional distribution of $\theta_i$ is

$$\theta_i | \theta_1, \ldots \theta_{i-1}, \alpha_0, G_0 \sim \left( \sum_{k=1}^{K} \frac{n_k}{i - 1 + \alpha_0} \delta_{\psi_k} \right) + \frac{\alpha_0}{i - 1 + \alpha_0} G_0. \tag{4}$$

The term $\delta_{\psi_k}$ is essentially an indicator function $I(\theta_i = \psi_k)$. This means that $\theta_i$ takes on the value of $\psi_k$ with probability proportional to the number of $\theta$'s already equal to $\psi_k$. In addition, $\theta_i$ has probability proportional to $\alpha_0$ of equalling a new draw from $G_0$. Notice that this sampling scheme has a positive reinforcement effect in which the more often a $\psi_k$ is drawn, the more likely it is to be drawn again. As the name suggests, the Chinese Restaurant process can be likened to a restaurant in which each table is a cluster and each customer is a point. In this clustering process, each new customer arriving at the restaurant will join an existing table with probability proportionate to how many customers are already seated at that table. The more popular the table, the more likely it is to draw in more customers.

# 4 Hierarchical Dirichlet Process

In the Hierarchical Dirichlet Process, each group, or document, has its own probability measure $G_j$ such that $G_j \sim DP(\alpha_0, G_0)$ where $G_0$ is also a DP with scaling parameter $\gamma$ and base probability measure $H$. In topic modeling, $H \sim Dir(\tilde{\alpha})$ where $\tilde{\alpha}$ is a vector with as many terms as unique words in the corpus vocabulary. Thus, we have

$$G_0 | \gamma, H \sim DP(\gamma, H) \tag{5}$$

$$G_j | \alpha_0, G_0 \sim DP(\alpha_0, G_0) \tag{6}$$

The Hierarchical Dirichlet Process provides a prior for the multi-document mixture model setting. The intuition behind using the HDP is that the discrete nature of the DP will allow clusters within the same document to share topics with non-zero probability. In addition, since the seeds of each documnt-specific DP were drawn from a common discrete distribution, different documents can share topics with non-zero probability. For each group j, let $\theta_{j1}, \theta_{j2}, \ldots$ be iid random variables distributed according to $G_j$. Each $\theta_{ji}$ is a factor corresponding to a single observation $x_{ji}$. The likelihood is given by

$$\theta_{ji} | G_j \sim G_j \tag{7}$$

$$x_{ji} | \theta_{ji} \sim F(\theta_{ji}) \tag{8}$$

In this notation, $G_0$ is the distribution of topics at the corpus level and $G_j$ is the distribution of topics at the document level. Using this construction, the continuity of $H$ no longer affects the ability of the model to

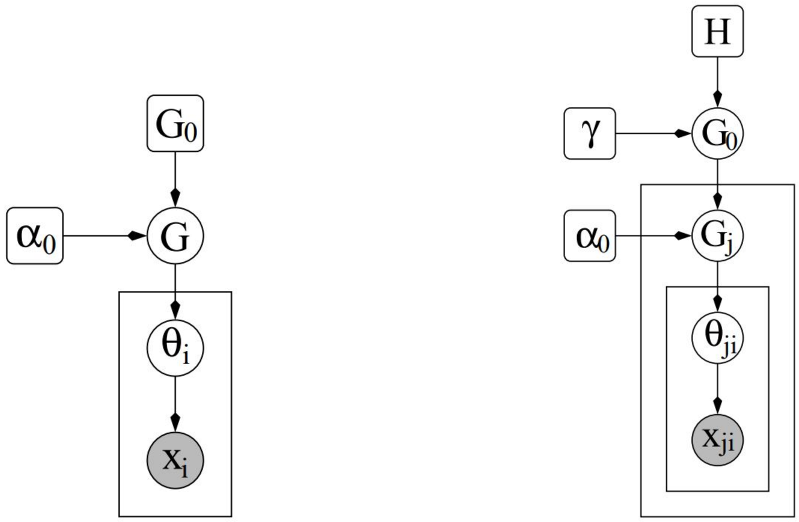


Figure 1: Graphical Model Representation of a DP Mixture Model (left), and a Hierarchical DP Mixture Model (right). Each unfilled node in the graph is a random variable, with the filled nodes denoting observed data. Rectangles denote replication of the model within the rectangle.

share topics. Furthermore, we do not need to know how many groups we will have per document since the distribution of topics within each document has an infinite number of seeds. Figure 1 shows the graphical interpretation of the DP and the hierarchical DP mixture models.

# 5 The Chinese Restaurant Franchise

The analog for a hierarchical DP is the Chinese restaurant franchise. In the franchise, the Chinese restaurant process is extended to include multiple restaurants that share a set of dishes from a global menu. In our document modeling setup, each restaurant corresponds to a document, the customers are the $\theta_{ji}$. A new variable $\psi_{jt}$ represents the table-specific choice of dish. In particular, $\psi_{jt}$ is the dish served at table $t$ in restaurant $j$, and the value of $\psi_{jt}$ some $\phi_k$ where $\phi_k \sim H$. Figure CFR is a pictoral depiction of the Chinese restaurant process.

Since each $\theta_{ji}$ is associated with one $\psi_{jt}$, and each $\psi_{jt}$ is associated with only one $\phi_k$, we can introduce indicators to denote these associations. Let $t_{ji} = t$ when $\theta_{ji}$ is seated at table with dish $\psi_{jt}$ and let $k_{jt} = k$ when $\psi_{jt}$ is seated at a table with $\phi_k$ as the dish. The clustering algorithm we employ uses only the $t_{ji}$ and $k_{jt}$ during the clustering process.

In the CFR, we marginalize out both the $G_j$ and the $G_0$ to find the marginal probabilities of $\theta_{ji}|\theta_{-ji}$ and $\psi_{jt}|\psi_{-jt}$. Because both $G_j$ and $G_0$ are Dirichlet Process distributions, the derivation for the marginals of $\theta_{ji}$ and $\psi_{jt}$ are the same. To facilitate the following equations, we need to introduce some notation for counting customers and tables. Let $n_{jtk}$ denote the number of customers in restaurant $j$ at table $t$ eating dish $k$. Marginal counts are represented with dots. For example, $n_{jt\cdot}$ represents the nubmer of customers in restaurant j at table t, and $n_{j\cdot k}$ represents the number of customers in restaurant $j$ eating dish $k$. The notation $m_{jk}$ denotes the number of tables in restaurant $j$ serving dish $k$. Thus, $m_{j\cdot}$ represents the number of tables in restaurant $j$, $m_{\cdot k}$ represents the number of tables serving dish $k$, and $m_{\cdot\cdot}$ represents the total number of tables occupied.

We first marginalize out $G_j$. This is the same as in the Chinese restaurant process section. Using our

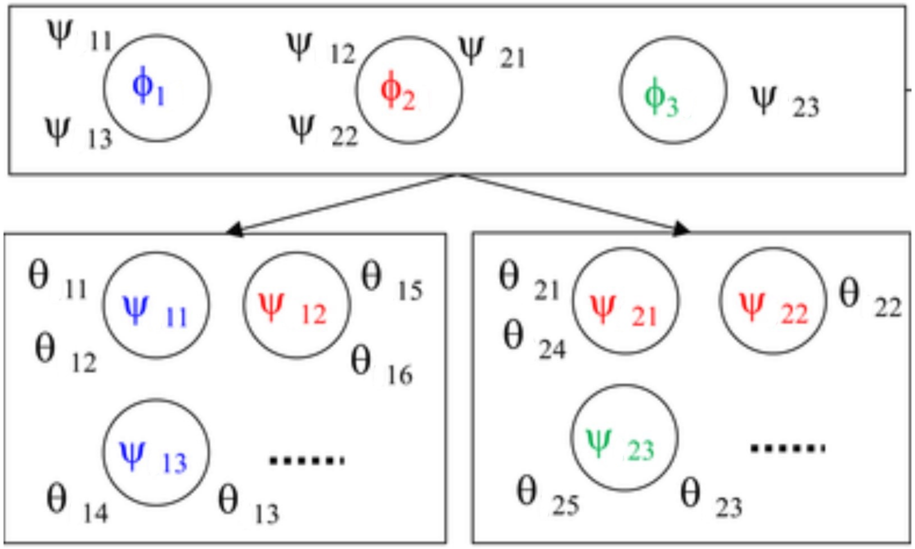


Figure 2: The HDP represented as Chinese restaurant franchise. On the topmost level, we have the clustering of $\psi_j t$ and in the lower levels, we have the clustering of $\theta_{ji}$. In the topic modeling application, $\psi_{jt}$ and $\theta_{ji}$ take on the value of the dish at their table. Not pictured are the words $x_{ji} \sim Mult(\theta_{ji}.)$

new counting notation, we have the conditional distribution for $\theta_{ji}$ given $\theta_{-j1}, \ldots, \theta_{j,i-1}$ is

$$\theta_{ji}|\theta_{j1}, \ldots \theta_{j,i-1}, \alpha_0, G_0 \sim \left( \sum_{j=1}^{m_j.} \frac{n_{jt}.}{i-1+\alpha_0} \delta_{\psi_{jt}} \right) + \frac{\alpha_0}{i-1+\alpha_0} G_0. \tag{9}$$

If $\theta_{ji}$ chooses a term from the summation, then we set $\theta_{ji} = \psi_{jt}$ and let $t_{ji} = t$. If the second term is chosen, we increment $m_j.$ by one and set $\theta_{ji} = \psi_{jm_j}$ where $\psi_{jm_j} \sim G_0.$ and $t_{ji} = m_j..$, which is the table index of the latest table in restaurant m.

Now we integrate out $G_0$ to get

$$\psi_{jt}|\psi_{j1}, \ldots \psi_{j,t-1}, \gamma, H \sim \left( \sum_{k=1}^{K} \frac{m.k}{m.. + \gamma} \delta_{\phi_k} \right) + \frac{\gamma}{m.. + \gamma} H. \tag{10}$$

Similar to $\theta_{Ji}$, if $\psi_{jt}$ chooses a term from the summation, then we set $\psi_{jt} = \phi_k$ and let $k_{jt} = k$. If the second term is chosen, we increment $K$ by one and set $\psi_{jt} = \phi_K$ where $\phi_K \sim H.$ and $k_{jt} = K$, which is the index of the latest drawn seed from $H$.

# 6 Experiment and Sampling Strategy

For our experiment, our data was Taylor Swift song lyrics. Each song is a document containing between 100-400 words. There are 81 songs in total.

## 6.1 Variable Tables

We first list all the variables we are going to use in sampling and explain what they represent in the model.

Table 1: Table of Variables used in Sampling

| | |
|---|---|
| $x_{ji}$ | ith word in jth document |
| $t_{ji}$ | the table $x_{ji}$ sits at |
| $k_{ji}$ | the topic(dish) of the table $x_{ji}$ sits at $(t_{ji})$ takes |
| $k_{jt}$ | the topic(dish) of the table $t$ in document $j$ |
| $n_{jt}$ | the number of words sit at table $t$ in document $j$ |
| $n_{jk}$ | the number of words at topic $k$ in document $j$ |
| $n_k$ | the number of words at topic $k$ |
| $n_{kv}$ | the number times of vocubulary $v$ is catogrized at topic $k$ |
| $n_{jv}$ | the number times of vocubulary $v$ is catogrized at document $j$ |
| $m_{jk}$ | the number of tables at topic $k$ in document $j$ |
| $m_{\cdot k}$ | the number of tables at topic $k$ |
| $m_{\cdot\cdot}$ | the number of tables |
| $V$ | the size of vocabulary |
| $f_k^{-x_{ji}}(x_{ji})$ | conditional probability of $x_ji$ belonging to topic $k$ given all other words except $x_{ji}$ |
| $f_k^{-x_{jt}}(x_{jt})$ | conditional probability of all words in table $t$ document $j$ belonging to topic $k$ given all other words |

We note that the variables have the key relationship from the Dirichlet Process theory described above

$$\theta_{ji} = \psi_{jt_{ji}}, \psi_{jt} = \phi_{k_{jt}}, z_{ji} = k_{jt_{ji}}$$

For the ease of sampling, instead of sampling the tables and dishes $\theta$ and $\psi$ directly, we sample their index variables $t_{ji}$ and $k_jt$ iteratively. The $\theta_{ji}$ and $\psi_{jt}$ can be reconstructed from their index $t_{ji}$ and $k_jt$, and the $\phi_k$s.

## 6.2 Sampling Method

To update $t_{ji}$ and $k_{jt}$ using Gibbs sampling, we first need to compute $f_k^{-x_{ji}}(x_{ji})$ and $f_k^{-x_{jt}}(x_{jt})$.

We know the topics $\phi_k$s are the probability vector of the length of vocabulary size, each element meaning the probability of this vocabulary word. A actual word belonging to topic $k$, means this word is generated from this probability vector $\phi_k$. $\phi_k$ has a dirichlet prior of $Dir(0.5,,,,0.5)$ and the words belonging to topic $k$ is multinomial $\phi_k$.

Using the Dirichlet and Multinomial Conjugacy we are able to calculate $f_k^{-x_{ji}}(x_{ji})$ and $f_k^{-x_{jt}}(x_{jt})$.

$$\begin{aligned}
f_k^{-x_{ji}}(x_{ji}) &= \frac{\int f(x_{ji}|\phi_k)\prod_{j'i'\neq ji, z_{j'i'}=k} f(x_{j'i'}|\phi_k)h(\phi_k)d\phi_k}{\int \prod_{j'i'\neq ji, z_{j'i'}=k} f(x_{j'i'}|\phi_k)h(\phi_k)d\phi_k} \\
&= \frac{\int \prod_{ji, z_{ji}=k} P_{x_{ji}} P_1^{0.5} P_2^{0.5} \ldots P_v^0.5dP}{\int \prod_{j'i'\neq ji, z_{ji}=k} P_{x_{j'i'}} P_1^{0.5} P_2^{0.5} .. P_v^{0.5} dP} \\
&\text{denote } n_j = \#\{x_{j'i'} = v_j\}, \text{WLOG suppose } x_{ji} = v_1 \\
&= \frac{\int P_1^{0.5+n_1+1} P_2^{0.5+n_2} .. P_v^{0.5} dP}{\int P_1^{0.5+n_1} P_2^{0.5+n_2} .. P_v^{0.5} dP} \\
&= \frac{\Gamma(0.5+n_1+1)\prod_{j=2}^V \Gamma(0.5+n_j)/\Gamma((0.5+n_1+1)+\sum_i^V(0.5+n_j))}{\Gamma(0.5+n_1)\prod_{j=2}^V \Gamma(0.5+n_j)/\Gamma((0.5+n_1)+\sum_i^V(0.5+n_j))} \\
&= \frac{0.5+n_1}{\sum_i^v (0.5+n_i)}
\end{aligned}$$

Similarly, we can derive at the table leve, the conditional probability of belonging to topic $k$ except now we are considering all the $x_j i$ at document $j$ table $t$, where $P_{x_{jt}} = P_{x_{jt_1}} P_{x_{jt_2}} .. P_{x_{jt_{n_{jt}}}}$.

$$
\begin{aligned}
f_k^{-x_{jt}}(x_{jt}) &= \frac{\int f(x_{jt}|\phi_k) \prod_{j't' \neq jt, z_{j't'}=k} f(x_{j't'}|\phi_k) h(\phi_k) d\phi_k}{\int \prod_{j't' \neq jt, z_{j't'}=k} f(x_{j't'}|\phi_k) h(\phi_k) d\phi_k} \\
&= \frac{\int P_{x_{jt_1}} P_{x_{jt_2}} .. P_{x_{jt_{n_{jt}}}} \prod_{j't' \neq jt, z_{j't'}=k} P(x_{j't'}) P_1^{0.5} .. P_v^{0.5} d\phi_k}{\int \prod_{j't' \neq jt, z_{j't'}=k} P(x_{j't'}) P_1^{0.5} .. P_v^{0.5} d\phi_k} \\
&\text{WLOG suppose } x_{jt_1},,, x_{jt_{n_{jt}}} \text{ takes value in } v_1,, v_{n_{jt}} \\
&= \frac{\int P^{0.5+n_{kv_1}+n_{jtv_1}} P^{0.5+n_{kv_2}+n_{jtv_2}} ... P^{0.5+n_{kv'_n}+n_{jtv'_n}} P^{0.5+n_{kv_{n'+1}}} ... P^{0.5+n_{kv_v}} dP}{\int P_1^{0.5+n_{kv_1}} P_2^{0.5+n_{kv_2}} .. P_v^{0.5+n_{kv_v}} dP} \\
&= \frac{\frac{\Gamma(0.5+n_{jtv_1})\Gamma(0.5+n_{jtv_2})..\Gamma(0.5+n_{jtv_v}) \prod_{i=n_{jtv+1}}^{V} \Gamma(0.5+n_{kv_i})}{\Gamma(\sum_i^{n_{jt}}(0.5+n_{kv_i}+1)+\sum_{n_{jt}+1}^{V}(0.5+n_{kv_i}))}}{\prod_i^V (0.5+n_{kv_i}) / \Gamma(\sum_i^V (0.5+n_{kv_i}))} \\
&= \frac{(0.5+n_{jt_{v1}})(0.5+n_{jt_{v2}})..(0.5+n_{jt_{v_v}})}{\Gamma(0.5V + \sum_{i=1}^V n_{kv_i} + n_{jt}) - \Gamma(0.5V + \sum_{i=1}^V n_{kv_i})}
\end{aligned}
$$

With $f_k^{-x_{ji}}(x_{ji})$ and $f_k^{-x_{jt}}(x_{jt})$ at hand, we use Gibbs to iteratively sample $t$ and $k$ following the Chinese Restaurant Franchise method we described in the pevious section.

#### 6.2.1 Sampling t

Following the Chinese Restaurant Franchise method, we have the likelihood for $t_{ji} = t_{new}$ is

$$
p(x_{ji}|t^{-ji}, t_{ji} = t_{new}, k) = \sum_{k=1}^{K} \frac{m \cdot k}{m.. + \gamma} f_k^{-x_{ji}}(x_{ji}) + \frac{\gamma}{m.. + \gamma} f_{k^{new}}^{-x_{ji}}(x_j i)
$$

The conditional distribution of $t_{ji}$ given $k$ and all other tables $t^{-ji}$ is

$$
p(t_{ji} = t|k, t^{-ji}) \propto \begin{cases} n_{jt}^{-ji} f_{k_{jt}}^{-x_{ji}}(x_{ji}) & \text{if } t \text{ is an existing table} \\ \alpha_0 p(x_{ji}|t^{-ji}, t_{ji} = t_{new}, k) & \text{if } k = k_{new} \end{cases}
$$

When the word $x_{ji}$ is seated at a new table, i.e., $t_{ji} = t_{new}$, we need to allocate a topic $k_{jt^{new}}$ to this new table by

$$
p(k_{jt^{new}} = k|t, k^{-jt^{new}}) \propto \begin{cases} m_{.k}^{-jt} f_k^{-x_{ji}}(x_{ji}) & \text{if } k \text{ is an existing topic} \\ \gamma f_{k_{new}}^{-x_{ji}}(x_{ji}) & \text{if } k = k_{new} \end{cases}
$$

#### 6.2.2 Sampling k

The conditional probability of $k_{jt} = k$ given the table $t$ and all other topics $k^{-jt}$ is

$$
p(k_{jt} = k|t, k^{-jt}) \propto \begin{cases} m_{.k}^{-jt} f_k^{-x_{jt}}(x_{jt}) & \text{if } k \text{ is an existing topic} \\ \gamma f_{k_{new}}^{-x_{jt}}(x_{jt}) & \text{if } k = k_{new} \end{cases}
$$

Note that if a table becomes unoccupied, i.e., $n_{jt} = 0$ we delete this table and its corresponding topic $k_{jt}$. If the topic $k$ becomes unallocated, i.e, $m_k = 0$, we delete this topic too. But the label (id) of the table $t$ and topic $k$ is reusable.

# 7 Results and Discussion

We benchmarked the effectiveness of the clustering by measuring the log likelihood of the data after every iteration. After clustering, we were able to reconstruct the posterior expected values of topic probability vectors by

$$\begin{aligned} E[topic] &= \frac{\alpha_k}{\sum_k \alpha_k} \\ &= \frac{n_{kv}[k] + 0.5}{n_k + 0.5 * V} \end{aligned}$$

where $n_{kv}, nk, V$ are as defined in the sampling methods section. Using these reconstructions and our clustering results, we were able to compute the log likelihood of each word. We used the sum of the log likelihood to monitor the performance of the algorithm. Figure 3 shows that the log likelihood steadily increases in the first two hundred iterations before stabilizing around -102500.

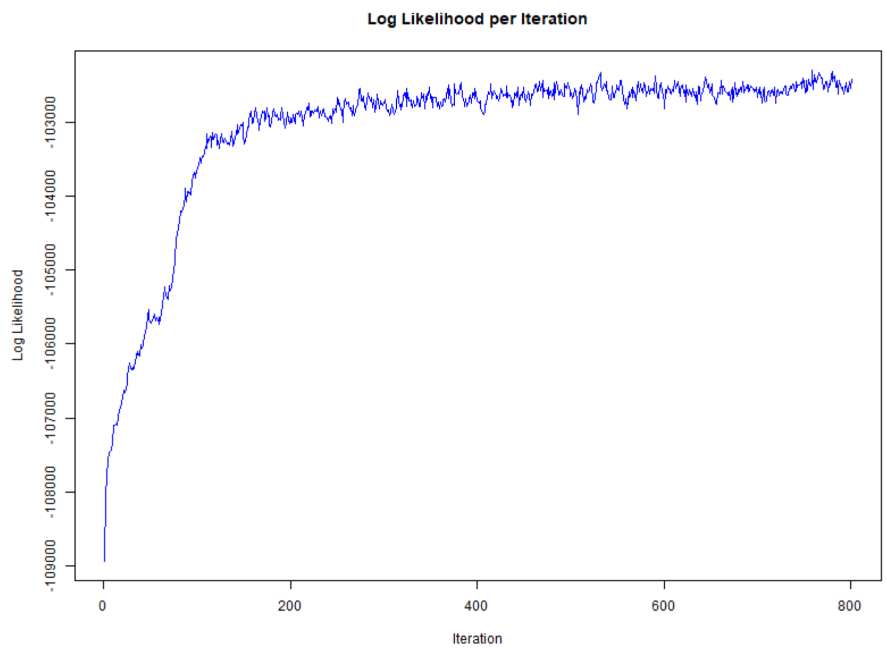


Figure 3: The log likelihood of the data over 800 iterations. The shape of this curve suggests that the algorithm reaches convergence.

As the log likelihood stabilizes, the clustering has also stabilized. Figure 4 shows that for the song "Shake It Off", the clustering proportion of the last twenty iterations reveals that the clustering changes very little. A manual check shows that the clustering proportions for the last hundred iterations are also fairly stagnant. The apparent convergence of the clustering suggests we can use the last iteration of the results to proceed with our analysis.

In total, there were 98 unique topics for the 81 songs. After using a softmax to convert the topic indexes back into words, and counting the number of words clustered under each topic, we find that by far the most popular topic is "I". Other notable topics include "love", "we", "shake", "dance", etc. While these words are heuristically in line with what we expect from Taylor Swift's lyrics, the softmax method of recovering topic labels is an oversimplification of the clustering results. For example, out of the 81 songs, 69 of them included a cluster under topic 9. Applying a softmax to topic 9 reveals that the word of greatest density is 'I'. 'I' is one of the most common words in the English language, and Taylor Swift does like to sing about herself, but the word 'I' not an informative label for explaining why so many words are clustered under this topic. Relaxing the softmax to include some other high density words shows that topic 9 actually includes

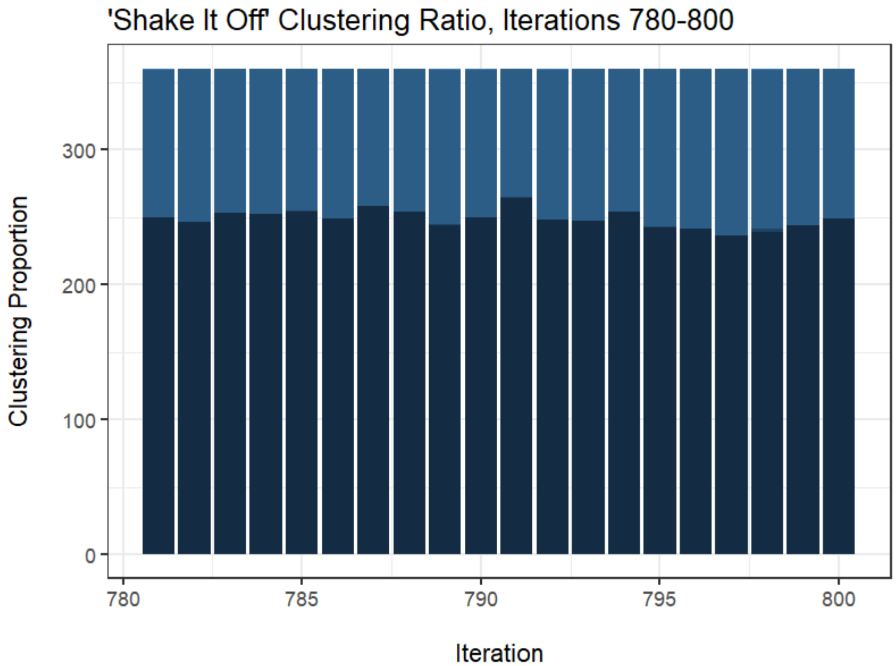


Figure 4: The darker portion of the bar represents the words clustered to topic 38. The lighter portion of the bar represents words clustered to topic 9. While we did not track which specific words were placed in each cluster, the number of words being assigned to each cluster does not change significantly between each iteration. This suggests that the clustering has reached an equilibruim.

Table 2: The words of high density together with their probability under topic 9. Most of the high density words in this topic are frequently used by Swift so it makes sense that this topic would be shared by 69 songs.

| Word | I | and | know | you | like | back | never | but | time | just | cause | love |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Probability | 0.097 | 0.036 | 0.020 | 0.019 | 0.017 | 0.015 | 0.015 | 0.013 | 0.012 | 0.011 | 0.009 | 0.007 |

many common words from Swift's lyrics (see Table 2). Since Swift often recycles her lyric material, her most idiosyncratic lines ended up clustered together into one 'topic', and that topic was then shared by nearly every song. This result vindicates our decision to use the hierarchical DP.

The topics generated by the clustering did not separate words into groups by meaning, but rather by the frequency of their occurrence. Using "Shake It Off" as an example, we can see that the algorithm separated the lyrics into words that occur frequently across songs, and words that are more unique to the current song. Figure 4 shows that the algorithm generally produces two clusters for 'Shake It Off'. One of the topics is 9, the topic containing all the common words. The other is topic 38. The words of highest density in topic 38 are "play", "gonna", "hate", "shake", "break", and "fake". These are words that are very pertinent to this song in particular and show up in much lower frequency in her other songs. Thus, topic 38 does not show up in any song.

The song 'The Best Day' provides an even better example of this phenomenon. The lyrics for 'The Best Day' feature heavy fairy tale imagery and words uncommon words like "pirate", "tractor", and "pumpkin patch." To compensate for the more varied vocabulary, the algorithm was able to produce 6 topics for this song. The results highlight the strengths of the hierarchical Dirichlet model in that the algorithm was able to create as many new groups as was necessary to accommodate unique material, while still being able to recognize that much of the input between documents was repetitive and therefore could be placed into the same cluster.

# 8 Appendix

In this section, we will show the derivation of the Chinese Restaurant process from the Dirichlet process. Recall the definition of the Dirichlet process: for any finite measurable partition $(A_1, A_2, \ldots, A_r)$ of $\Theta$, the random vector $(G_j(A_1), G_j(A_2), \ldots, G_j(A_r))$ is distributed as a finite-dimensional Dirichlet Distribution with parameters $(\alpha_0 G_0(A_1), \alpha_0 G_0(A_2), \ldots, \alpha_0 G_0(A_r))$

$$(G_j(A_1), G_j(A_2), \ldots, G_j(A_r)) \sim Dir(\alpha_0 G_0(A_1), \alpha_0 G_0(A_2), \ldots, \alpha_0 G_0(A_r)) \tag{11}$$

To obtain the Chinese Restaurant process, we integrate out $G_j$. Suppose we are drawing

$$\theta_1, \ldots, \theta_k \overset{iid}{\sim} Multinomial(G_1, \ldots G_k)$$

where

$$(G_1, \ldots, G_k) \sim DP(\alpha_0, G_0).$$

We are interested in the probability of $\theta_i = \psi_i$ given the values of $\theta_j$ excluding $\theta_i$, e.g.

$$P(\theta_i = \psi_i | \theta_{-i}) = P(\theta_i = \psi_i | \theta_1, \ldots, \theta_{i-1}, \theta_{i+1}, \ldots, \theta_n).$$

Let $\tilde{G} = (G_1, \ldots, G_k)$. We now integrate out $\tilde{G}$

$$\begin{aligned}
P(\theta_i = \psi_i | \theta_1, \ldots, \theta_{i-1}, \theta_{i+1}, \ldots, \theta_n) &= \int_{G_{1:k}} P(\theta_i = \psi_i | \theta_{-i}, \tilde{G}) P(\tilde{G} | \theta_{-i}) d\tilde{G} \\
&= \int_{G_{1:k}} P(\theta_i = \psi_i | \tilde{G}) P(\tilde{G} | \theta_{-i}) d\tilde{G} \quad \text{by the conditional independence of } \theta_i \text{ given } \tilde{G} \\
&= \int_{G_{1:k}} G_i P(\tilde{G} | \theta_{-i}) d\tilde{G} \quad \text{where } P(\theta_i = \psi_i | \tilde{G}) = G_i \\
&= \frac{1}{B(\alpha_1, \ldots, \alpha_k)} \int_{G_{1:k}} G_i (G_1^{\alpha_1 + n_1^{-i} - 1} \cdots G_k^{\alpha_k + n_k^{-i} - 1}) d\tilde{G} \quad \text{where } \alpha_k = \alpha_0 G_0(A_k)
\end{aligned}$$

The notation in the last line merits some explanation. Because the Dirichlet and Multinomial distributions are conjugate, the posterior of $G_k$ has in the exponent $n_k^{-i}$, which denotes the number of $\theta_j$ that equal $\psi_k$ excluding the value of the $\theta_i$. Continuing with the derivation by folding $G_i$ into the rest of the Dirichlet kernel, we have:

$$\begin{aligned}
&\frac{1}{B(\alpha_1, \ldots, \alpha_k)} \int_{G_{1:k}} G_1^{\alpha_1 + n_1^{-i} - 1} \cdots G_i^{\alpha_i + n_i^{-i} + 1 - 1} \cdots G_k^{\alpha_k + n_k^{-i} - 1}) d\tilde{G} \\
&= \frac{\Gamma(\sum_{j=1}^k \alpha_j + n_j^{-i})}{\prod_{j=1}^n \Gamma(\alpha_j + n + j^{-i})} \frac{[\prod_{j \neq i} \Gamma(\alpha_j n_j^{-i})] \Gamma(\alpha_i + n_i^{-i} + 1)}{\Gamma(\sum_{j \neq i} (\alpha_j + n_j^{-i}) + \alpha_i + n_i^{-i} + 1)} \\
&= \frac{\Gamma(\alpha_i + n_i^{-i} + 1)}{\Gamma(\alpha_i + n_i^{-i})((\sum_{j=1}^k \alpha_j) + n^{-i})} \quad \text{where } n^{-i} \text{ is the total number of } \theta \text{ - 1.} \\
&= \frac{\alpha_i + n_i^{-i}}{(\sum_{j=1}^n \alpha_j) + n^{-i}}
\end{aligned}$$

As $k \to \infty$, $G_0(A_k) \to 0$ and so $\alpha_j \to 0$. Recall $\alpha_j = \alpha_0 * G_0(A_j)$ where $G_0(A_j)$ is a probability assigned to the partition $A_j$. Thus, $\sum_{j=1}^k = \alpha_0$ and we have

$$P(\theta_i = \psi_i) = \frac{n_i^{-i}}{\alpha_0 + n^{-i}}$$

Summing over all possible values of $i$, we have

$$\theta_i | \theta_{-i} = \left( \sum_{i=1}^k \frac{n_i^{-i}}{\alpha_0 + n^{-i}} \delta_i \right) + \frac{\alpha_0}{\alpha_0 + n^{-i}} G_0$$

where the second term denotes the probability that $\theta_i$ will take on some new value not yet drawn. This second term is included to allow the probabilities to sum up to 1.

# References

[1] Y. W. Teh, M.I. Jordan, M.J. Beal, D.M. Blei. Hierarchical Dirichlet Processes. *Journal of the American Statistical Association*, pg 1566-1581, 2006.