Topic Models with Power-law using Pitman-Yor process

Issei Sato
Graduate School of Information Science
and Technology
The University of Tokyo, Japan
sato@r.dl.itc.u-tokyo.ac.jp

Hiroshi Nakagawa
Information Technology Center
The University of Tokyo, Japan
nakagawa@dl.itc.u-tokyo.ac.jp

ABSTRACT

One of the important approaches for Knowledge discovery and Data mining is to estimate unobserved variables because latent variables can indicate hidden and specific properties of observed data. The latent factor model assumes that each item in a record has a latent factor; the co-occurrence of items can then be modeled by latent factors. In document modeling, a record indicates a document represented as a "bag of words", meaning that the order of words is ignored and an item indicates a word. Latent Dirichlet allocation (LDA) has stimulated the use of the Dirichlet distribution over the latent topic distribution of a document. LDA assumes that latent topics, i.e., discrete latent variables, are distributed according to a multinomial distribution whose parameters are generated from the Dirichlet distribution. In an experiment using real data, this model outperformed LDA in document modeling.

Keywords

Topic Model, Latent Dirichlet Allocation, Nonparametric Bayes,Pitman-Yor Process, Power-law

1. INTRODUCTION

Probabilistic models with latent variables have attracted attention in Knowledge discovery and Data mining because of their power and flexibility to model real world phenomena. In this approach, it is necessary is to estimate unobserved, namely latent variables. The latent factor model assumes that each item in a record has a latent factor and that the co-occurrence of items can then be modeled by latent factors. The goals of a probabilistic modeling are to find shot descriptions that preserve the statistical relationships of data and predict new occurrences.

In document modeling, a record indicates a document represented as a "bag of words", meaning that the order of words is ignored and an item indicates a word. The latent factor in fact stands for the topic. Probabilistic latent semantic indexing (PLSI)[1] was one of the first topic models. PLSI has a problem in that it cannot treat new document data that does not coincide with any of the training data. Latent Dirichlet allocation (LDA)[2] generalizes PLSI by applying a Bayesian framework that can avoid over-fitting and can treat new documents on the basis of a prior distribution. LDA assumes that latent topics, i.e., discrete latent variables, are distributed according to a multinomial distribution whose parameters are generated from the Dirichlet distribution. PLSI and LDA model a document's property that a document has multiple topics. LDA has stimulated the use of the Dirichlet distribution over the latent topic distribution of a document and inspired many other topic models such as LDA-HMM [3], author-topic model [4, 5], entity-topic models [6], correlated topic model [7], hidden topic Markov models [8], dynamic topic model [9, 10], topic models for text and citations [11], topic model for visualization [12], topic models for Hypertext [13], topic models conditioned on arbitrary features [14], syntactic topic models [15], and so on.

LDA also modes a word distribution by using the multinomial distribution and parameters of the multinomial distributions follows the Dirichlet distribution. These Dirichlet-multinomial settings cannot capture the power-law phenomenon of a word distribution, which is known as "Zipf's law" in linguistics. The Power-law distributions are produced by a stochastic process in which frequent outcomes attract probability mass such as "rich-get-richer" process. A major example of a power-law distribution is a distribution of links pointing to web pages. New web pages are more likely to link to already-popular pages that have already a lot of links. A widely used process is a preferential attachment process[]. One of the statistical properties of natural language is that word frequencies follow a power-law distribution given by

$$p(n^w = x) \propto x^{-l}, \tag{1}$$

where n^w is the number of frequencies of words and l is some constant parameter. This observation is often called Zipf's law. Fig.1 (a) shows the empirical probability of words in Reuters corpus. The plots appear approximately linear on a log-log plot. This behavior is characteristic of a power-law distribution. Fig.1 (c) indicates that the power-law property of a word distribution is also observed in a document. The Pitman-Yor (PY) process[**?**] is one of the most adaptive processes for a document modeling due to its exchangeability property.

In this paper, we develop a topic model based on the hierarchical Pitman-Yor (HPY) process for modeling a word distribution. The PY process is a stochastic process generalized from the Dirichlet process[16]. The PY process has a concentration parameter γ and a discount parameter d that control the power-law property. If discount parameter is set to zero, the PY process has the same property as the Dirichlet process. The a discount parameter place emphasis on a new-word generation that induces a long-tail phenomena of a distribution, which is useful for modeling word-frequency distributions that tend to have many frequency-1 words. The PY process that has a stochastic process called the Chinese restaurant process that is a process of customers' seating arrangement in a restaurant where the number of customers seated at tables follows a power-law distribution. In this case, the power-law parameter l in Eq.(1) is equal to 1+d. Because of the power-law property, the Pitman-Yor process applies well natural language processing, in cases morphological structure analysis of words[17], N-gram language modeling [18, 19], a dependency parsing[20], and so forth.

We assume a power-law word distribution not only in a document but also in a topic. A topic in LDA is represented by a word distribution. A word distribution in a specific topic, e.g. search engines, can give high probability to the word specific to the topic, e.g., “Google ’’, “Yahoo ’’, and “MSN ”. Like many phenomena of linguistics, a power-law property can be observed in a topic-word distribution. For example, Reuters corpus has labels indicating their document's topic and each document has multiple labels. Fig.1 (c) illustrates the empirical probability of words in documents with label “trade”. Fig.1 (c) also provides a power-law phenomena. We models this property by using the hierarchical Pitman-Yor (HPY) process[18, 19].

Contribution and Remainder.

We propose a novel topic model using the PY process, called the PY topic model. The PY topic model captures two properties of a document, a power-low word distribution and multiple topics.

The remainder of this paper is organized as follows. Section 2 overviews LDA. Section 3 describes the PY process. Section 4 proposes the PY topic models. Section 5 presents experimental results. Section 6 analyzes extracted latent topics. Section 7 summarizes the paper.

2. LDA

In this section, we overview LDA where documents are represented as random mixtures over latent topics and each topic is characterized by a distribution over words. First, we define notation. T is the number of topics. M is the number of documents. V is the number of vocabulary size. N_j is the number of words in document j. $w_{j,i}$ denotes the i-th word in document j. $z_{j,i}$ denotes the latent topic of word $w_{j,i}$. $Multi(\cdot)$ is a multinomial distribution. $Dir(\cdot)$ is a Dirichlet distribution. $\boldsymbol{\theta}_j$ denotes a T-dimensional probability vector that is the parameters of the multinomial distribution, and represents the topic distribution of document j. $\boldsymbol{\phi}_t$ is a V dimensional probability vector where $\phi_{t,v}$ specifies the probability of generating word v given topic t. $\boldsymbol{\alpha}$ is the T-dimensional parameter vector of the Dirichlet distribution over θ_j $(j = 1, \cdots, M)$. β is a parameter of th Dirichlet over $\boldsymbol{\phi}_t$ $(t = 1, \cdots, T)$.

LDA assumes the generative process shown in Algorithm **??**. The graphical model of LDA is shown in Fig. 2 (a).

The Gibbs sampler is applied given by

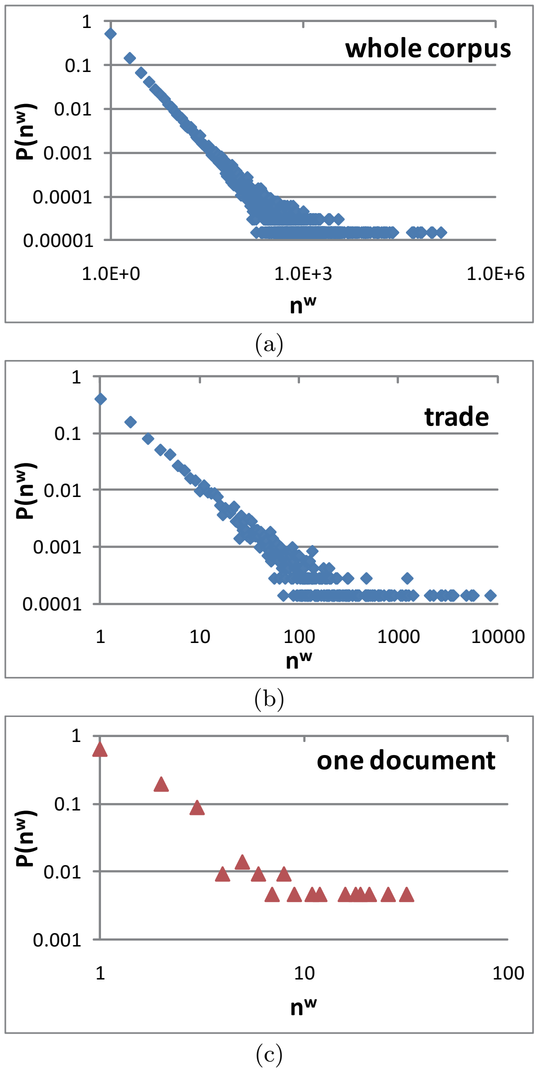
$$p(z_{j,i} = k | w_{j,i} = v, \boldsymbol{Z}^{-j,i}, \boldsymbol{W}^{-j,i}) = \frac{N_{j,k}^{-j,i} + \alpha_k}{N_j^{-j,i} + \alpha_0} \frac{N_{k,v}^{-j,i} + \beta}{N_{k,\cdot}^{-j,i} + V\beta}, \tag{2}$$

where $\alpha_0 = \sum_k \alpha_k$, $\boldsymbol{Z}$ denotes a set of all latent topic variables, $\boldsymbol{Z}^{-j,i} = \boldsymbol{Z} \backslash \{z_{j,i}\}$, $\boldsymbol{W}$ denotes a set of all words, $\boldsymbol{W}^{-j,i} = \boldsymbol{W} \backslash \{w_{j,i}\}$, $\boldsymbol{\alpha}$ is estimated by fixed point iteration[21].

$$\alpha_k^{new} = \frac{\sum_j \{\Psi(\alpha_k^{old} + n_{j,k}) - \Psi(\alpha_k^{old})\}}{\sum_j (\Psi(N_j + \alpha_0^{old}) - \Psi(\alpha_0^{old}))} \alpha_k^{old}. \tag{3}$$

The predictive probability of a new word in document j, given observed data is

$$p(w_j^{new} = v | \boldsymbol{Z}, \boldsymbol{W}) = \sum_{t=1}^{T} \frac{N_{j,t} + \alpha_k}{N_j + \alpha_0} \frac{N_{t,v} + \beta}{N_{t,\cdot} + V\beta}, \tag{4}$$



(a) (b) (c)

Figure 1: Example of power-low property.
(a), (b) and (c) illustrate the power-law distribution of words in all documents, trade-topic documents and one document with 500 words in the Reuters corpus, respectively. n^w is the number of occurrences of words and $p(n^w)$ is a probability of n^w on the log-log axes.

3. PITMAN-YOR PROCESS

In this section, we explain the Pitman-Yor (PY) process [22, 18, 19] by modeling a document. The PY document model captures the power-law property of the word distribution.

Algorithm 1 Generative process of LDA

Draw $\boldsymbol{\phi}_t \sim Dir(\boldsymbol{\phi}|\beta)$ $(t = 1, \cdots, T)$,
for all document $j (= 1, \cdots, M)$ **do**
  Draw $\boldsymbol{\theta}_j \sim Dir(\boldsymbol{\theta}|\boldsymbol{\alpha})$,
  for all word $i (= 1, \cdots, N_j)$ **do**
    Draw topic $z_{j,i} \sim Multi(z|\boldsymbol{\theta}_j)$,
    Draw word $w_{j,i} \sim p(w|z_{j,i}, \boldsymbol{\phi})$,
  end for
end for
where $Dir(\boldsymbol{\theta}|\boldsymbol{\alpha}) \propto \prod_t \theta_k^{\alpha_k - 1}$, $Dir(\phi|\beta) \propto \prod_v \phi_v^{\beta-1}$ and $p(w = v|z = t, \boldsymbol{\phi}) = \phi_{t,v}$.

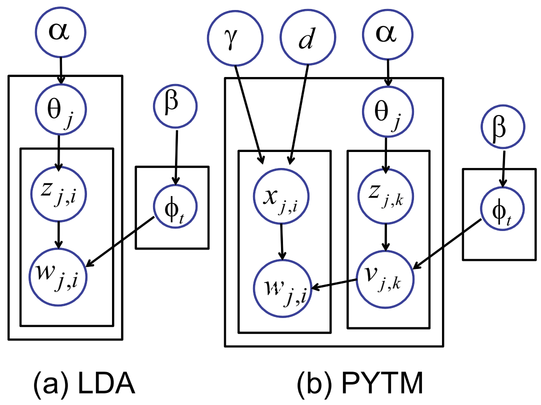


Figure 2: Graphical models of (a) LDA and (b) our proposal

The PY process PY(γ, d, G_0) is a distribution over distributions over a probability space. The PY process has three parameters, a concentration parameter γ, a discount parameter $d (0 \leq d \leq 1)$ that controls the power-law property of distribution and a base distribution G_0 that is understood as a mean of draws from PY(γ, d, G_0). The PY process is a generalization of the Dirichlet process where the discount parameter is regarded as zero in the Dirichlet process. The PY document model has a perspective given by the Chinese restaurant process (CRP) [23]. We consider two kinds of distributions for a document collection: let $G_0(w)$ be a general word distribution, i.e., the base distribution of the whole set of back-off document collections, and $G_j(w)$ be a document-specific word distribution for document j.

The generation process for the PY document model is

$$G_j \sim PY(\gamma, d, G_0), \quad \boldsymbol{w}_j \sim G_j. \tag{5}$$

We now provide details on a CRP representation for the PY document model. A CRP representation is composed of four elements, a customer, a table, a dish, and a restaurant. A customer denotes a word in a document, a table a latent variable, and a dish a word type. A restaurant denotes a document. Let $w_{j,1}$, $w_{j,2}$, $\cdots$ be a sequence of identical, independent draws from G_j, i.e., $\{w_{j,i}\}$ denotes words in document j. The sequence, $\{w_{j,i}\}$, represents customers visiting restaurant j corresponding to G_j with an unbounded number of tables. $\{x_{j,i}\}$ denotes seating arrangements of customers. $x_{j,i} = k$ indicates that the i-th customer sits in the k-th table. $v_{j,k} = v$ denotes that word type v is served at the k-th table in restaurant j. Namely, if $x_{j,i} = k$ and $v_{j,k} = v$, then $w_{j,i} = v$ that means the i-th word in document j is word type v . For example, $w_{i,j}$="the" ($x_{j,i} = k$ and $v_{j,k}$="the") indicates the i-th customer visiting a restaurant j is eating dish (word) "the". Fig. 3 explains an example of the CRP representation. Note that, the HY document model assumes that a document is represented as a "bag of words", meaning that the order of words is ignored and an item indicates a word.

The CRP assigns a distribution over the seating arrangement of the customers. The sequence generated with CRP can be shown to be exchangeable [23]. When the i-th customer $x_{j,i}$ enters restaurant j with K_j occupied tables at which other customers $(x_{j,1}, \cdots, x_{j,i-1})$ have already been seated, the new customer sits at a table under two conditions:

$$\begin{cases} \text{The } k\text{-th occupied table with probability} & \dfrac{N^c_{j,k} - d}{\gamma + N^c_{j,\cdot}}. \\ \text{A new unoccupied table with probability} & \dfrac{\gamma + dK_j}{\gamma + N^c_{j,\cdot}}. \end{cases} \tag{6}$$

Here, $N^c_{j,k}$ denotes the number of customers sitting at the k-th table and $N^c_{j,\cdot} = \sum_t N^c_{j,k}$ indicates the document length N_j. K_j denotes the total number of tables in restaurant j. If a customer sits at a new table, word v^{new} is drawn from the base distribution $G_0(v)$ and served at the new table . This means that $w_{j,i}$ is given value v^{new}, which is a term in the document, i.e., this indicates that the i-th word in document j is term v^{new}. If the customer sits at the k-th table, $x_{j,i}$ is given value v^k, which is the word served at the table. If d is not zero, the number of tables increases as many customers enter the restaurant, and this leads to a power-law phenomenon.

The predictive probability of a new word, given the seating arrangement is

$$p(w_{j,i+1} = v|\{w_{j,i}\}, \{x_{j,i}\}) = \frac{N_{j,v} - dK_{j,v}}{\gamma + N_j} + \frac{\gamma + dK_{j,\cdot}}{\gamma + N_j} G_0(v), \tag{7}$$

where $N_{j,v}$ denotes the number of customers serving word v that indicates the frequency of word v in document j, and $K_{j,v}$ denotes the number of tables serving word v in restaurant j.

The discount parameter, d, and the number of table, $K_{j,v}$, effect smoothing. In a prediction of a word, frequent words such as "the" and "a " often hurt the performance. However, their frequency is discounted by $dK_{j,v}$ in Eq.(7). The discount parameter, which places more emphasis on new-word (new-table in the CRP) generation than $d = 0$, is useful for modeling word frequency distributions that tend to have many frequency-1 words.

4. PROPOSED MODEL

The basic idea of our model is that a word distribution is generated from the PY process. First, we propose the PY topic model and then, the HPY topic model that is more general model.

4.1 Pitman-yor topic model

The difference between LDA and the PY topic model is how to generate a topic in a document. Although LDA generates a topic in each word in a document, We assume that the PY topic model generates a topic in each table in CRP representation for a document. That is, the number of generated topics is equal to that of words in LDA and that of tables in the PY topic model. Therefore, we introduce latent variable $z_{j,k} = t$ which denotes that topic t is assigned in the k-th table in document j. Like the PY document model, a customer sits in a table following Eq.(6) in the PY topic model. Moreover, if a customer sits in a new unoccupied table, then samples topic from a topic distribution corresponding to a document and sample word (dish) from a word distribution corresponding to the sampled topic.

The PY topic model assumes the generative process shown in algorithm 2 from an analogy to the PY document model. The graphical model of the PY topic model is shown in Fig. 2 (b). The inference for seating arrangements is given by Algorithm 3 where AddCustomer and RemoveCustomer are described in the Appendix.

Algorithm 2 Generative process of PYTM

1: Draw $\boldsymbol{\phi}_t \sim Dir(\boldsymbol{\phi}|\beta)$ $(t = 1, \cdots, T)$,
2: **for all** document $j(= 1, \cdots, M)$ **do**
3: Draw $\boldsymbol{\theta}_j \sim Dir(\boldsymbol{\theta}|\boldsymbol{\alpha})$,
4: **for all** word $i(= 1, \cdots, N_j)$ **do**
5: Sit in the k-th occupied table with proportion to $N^c_{j,k} - d$, i.e., $x_{j,i} = k$ and $w_{j,i} = v_{j,k}$.
6: Sit in a new unoccupied table with proportion to $\gamma + dK_j$, draw topic $z_{j,k^{new}} \sim Multi(z|\boldsymbol{\theta}_j)$, and draw word type $v^{new} \sim p(w|z_{j,k^{new}}, \boldsymbol{\phi})$ in the new table, i.e., $x_{j,i} = k^{new}$,$v_{j,k^{new}} = v^{new}$ and $w_{j,i} = v^{j,k^{new}}$,
7: **end for**
8: **end for**

Algorithm 3 Inference for PYTM and HPYTM

for iterasions **do**
  for all document $j(= 1, \cdots, M)$ **do**
    for all word $i(= 1, \cdots, N_j)$ **do**
      RemoveCustomer($w_{j,i}$,document j)
      AddCustomer($w_{j,i}$,document j)
    end for
  end for
  Estimate $\boldsymbol{\alpha}$ by using Eq.(3)
end for

The predictive probability of a new word, given words, topics and the seating arrangements in documents is

$$p(w_j^{new} = v|\boldsymbol{W}, \boldsymbol{Z}, \boldsymbol{X}) = \frac{N_{j,v} - dK_{j,v}}{\gamma + N_j} + \frac{\gamma + dK_{j,\cdot}}{\gamma + N_j} \sum_{t=1}^{T} \frac{N_{j,t} + \alpha_t}{N_j + \alpha_0} \frac{N_{t,v} + \beta}{N_{t,\cdot} + V\beta}, \quad (8)$$

where $N_{j,v}$ denotes the number of customers serving word v that indicates the frequency of word v in document j, and $K_{j,v}$ denotes the number of tables serving word v in document j.

The probability of a topic generating in a new table is given by

$$p(z_{j,k^{new}} = t|w_{j,i} = v, x_{j,i} = k^{new}, \boldsymbol{Z}, \boldsymbol{W}^{-j,i}, \boldsymbol{X}^{-j,i}) = \frac{N_{j,t} + \alpha_k}{K_j + \alpha_0} \frac{N_{t,v} + \beta}{N_{t,\cdot} + V\beta}, \quad (9)$$

where $N_{j,t}$ indicates the number of tables in which a seated word is generated from topic t.

4.2 Hierarchical Pitman-Yor topic model

We propose a more general model, the hierarchical Pitman-Yor (HPY) topic model that assumes a power-law word distribution not only in a document but also in a topic. The HPT topic model models a power-law property of a topic-specific word distribution by using the hierarchical Pitman-Yor (HPY) process[18, 19]. We replace the generation process of $\{\boldsymbol{\phi}_t\}$ in the PY topic model (Algorithm 2 step 2) as follows.

$$\boldsymbol{\phi}_t \sim PY(\gamma_1, d_1, \boldsymbol{\phi}_0) \; (t = 1, \cdots, T), \quad \boldsymbol{\phi}_0 \sim PY(\gamma_0, d_0, U), \quad (10)$$

where $\boldsymbol{\phi}_0$ is a base word distribution in whole corpus, U denotes a uniform distribution in which the probability of all words is assigned according to the size of the vocabulary V, i.e., $U(v) = 1/V$. The relationship of $\boldsymbol{\phi}_t$ and $\boldsymbol{\phi}_0$ just looks like that of (a) and (b) in Fig.1. Like a word distributions of each document, $\boldsymbol{\phi}_t(t = 0, 1, \cdots, T)$ can be also represented as the CRP.

The predictive probability of a new word, given words, topics and the seating arrangements in documents is recursively given by

$$p(w_j^{new} = v|\boldsymbol{W}, \boldsymbol{Z}, \boldsymbol{X}) = \frac{N_{j,v} - dK_{j,v}}{\gamma + N_j} + \frac{\gamma + dK_{j,\cdot}}{\gamma + N_j} \sum_{t=1}^{T} p_t(w_j^{new}) \quad (11)$$

$$p_t(v) = \frac{N_{t,v} - d_1 K_{t,v}}{\gamma + N_t} + \frac{\gamma + d_1 K_{t,\cdot}}{\gamma_1 + N_t} p_0(w_j^{new}) \quad (12)$$

$$p_0(v) = \frac{N_{0,v} - d_0 K_{0,v}}{\gamma + N_t} + \frac{\gamma + d_0 K_{0,\cdot}}{\gamma_0 + N_0} \frac{1}{V} \quad (13)$$

$$(14)$$

where $N_{t,v}(t = 0, 1, \cdots, T)$ denotes the number of customers serving word v in topic t that indicates the frequency of word v appearing in topic t, and $K_{t,v}$ denotes the number of tables serving word t in topic t.

The probability of a topic generating in a new table is given by

$$p(z_{j,k^{new}} = t|x_{j,i} = k^{new}, \boldsymbol{Z}, \boldsymbol{W}, \boldsymbol{X}^{-j,i}) = \frac{\tilde{N}_{j,t} + \alpha_k}{K_j + \alpha_0} p_t(w_{j,i}). \quad (15)$$

Note that $\tilde{N}_{j,t}$ indicates the number of tables in which a seated word is generated from topic t.

5. EXPERIMENTS

6. KNOWLEDGE DISCOVERY

7. CONCLUSION

Acknowledgements

This research was funded in part by a MEXT Grant-in-Aid for Scientific Research on Priority Areas "i-explosion" in Japan.

8. REFERENCES

[1] T. Hofmann. Probabilistic latent semantic indexing. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 50–57. ACM Press, 1999.

[2] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent Dirichlet Allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

[3] T. L. Griffiths, M. Steyvers, D. M. Blei, and J. B. Tenenbaum. Integrating topics and syntax. In *In Advances in Neural Information Processing Systems 17*, pages 537–544. MIT Press, 2005.

[4] M. Rosen-Zvi, T. Griffiths, M. Steyvers, and P. Smyth. The author-topic model for authors and documents. In *Proceedings of the 20th conference on Uncertainty in artificial intelligence*, pages 487–494, Arlington, VA, USA, 2004. AUAI Press.

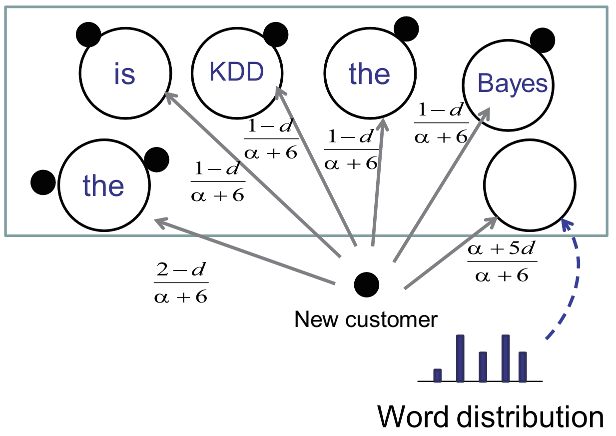


Figure 3: An example of the Chinese restaurant representation for the PY document model.
Black dots indicate customers. A customer sits in a table proportional to the number of customers that have already sat. A customer also can sit in a new table in some probability. For example, a customer sits in a table serving word "the" proportional to $\frac{2-d}{\alpha+6}$ and a new table proportional to $\frac{\alpha+5d}{\alpha+6}$. If a customer sits in a new table, a word is generated from a base distribution G_0 and served in the table. Multiple tables can serve a word type that has already served in other tables, e.g. a new table can serve word "the". Since frequent words tend to have many tables, the total frequency is discounted corresponding to the number of table and parameter d.

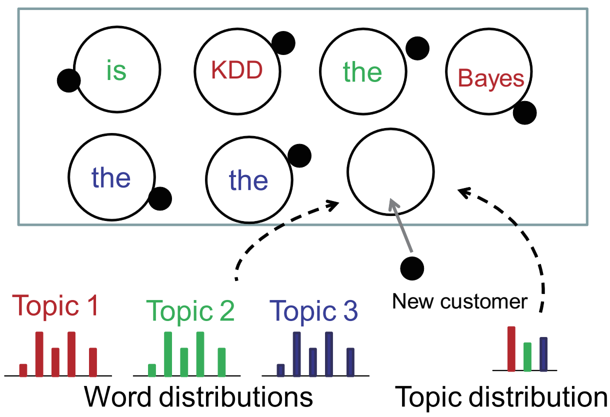


Figure 4: An example of the Chinese restaurant representation for LDA.
A customer always sits in a new table. If a customer sits in a new table, a topic is generated from the topic distribution $\boldsymbol{\theta}_j$ in document j and a word is drawled from a topic-specific word distribution $bphi$. Therefore LDA has a property of multiple topics.

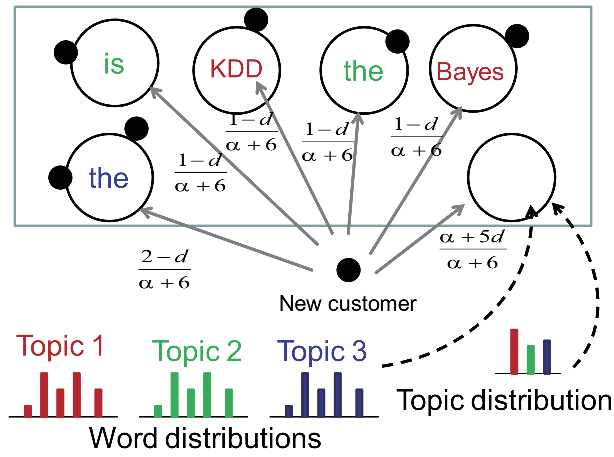


Figure 5: An example of the Chinese restaurant representation for the PY topic model.
A customer sits in a table proportional to the number of customers that have already sat as well as the PY document model. If a customer sits in a new table, a topic is generated from the topic distribution as well as LDA and a word is served in the table from a topic-specific word distribution. The PY topic model has properties of a power-law and multiple topics.

[5] M. Steyvers, P. Smyth, M. Rosen-Zvi, and T. Griffiths. Probabilistic author-topic models for information discovery. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 306–315, New York, NY, USA, 2004. ACM Press.
[6] D. Newman, C. Chemudugunta, and P. Smyth. Statistical entity-topic models. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 680–686, New York, NY, USA, 2006. ACM Press.
[7] D. M. Blei and J. D. Lafferty. Correlated Topic Models. In *NIPS*, 2005.
[8] A. Gruber, M. Rosen-Zvi, and Y. Weiss. Hidden Topic Markov Models. In *Proceedings of In Artificial Intelligence and Statistics*, 2007.
[9] D. M. Blei and J. D. Lafferty. Dynamic topic models. In *Proceedings of the 23rd international conference on Machine learning*, pages 113–120, New York, NY, USA, 2006. ACM.
[10] C. Wang, D. M. Blei, and D. Heckerman. Continuous Time Dynamic Topic Models. In *UAI*, pages 579–586, 2008.
[11] R. M. Nallapati, A. Ahmed, E. P. Xing, and W. W. Cohen. Joint latent topic models for text and citations. In *Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 542–550, New York, NY, USA, 2008. ACM.
[12] T. Iwata, T. Yamada, and N. Ueda. Probabilistic latent semantic visualization: topic model for visualizing documents. In *Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 363–371, New York, NY, USA, 2008. ACM.
[13] A. Gruber, M. Rosen-Zvi, and Y. Weiss. Latent Topic Models for Hypertext. In *UAI*, pages 230–239, 2008.
[14] D. M. Mimno and A. McCallum. Topic Models Conditioned on Arbitrary Features with Dirichlet-multinomial

Algorithm 4 Function WordProbability(word v, topic t)

if PY topic model **then**
  Return $\dfrac{N_{tv} + \beta}{N_{t\cdot} + V\beta}$.
else if HPY topic model **then**
  if $t > 0$ **then**
    Return $\dfrac{N_{t,v} - d_1 K_{t,v}}{\gamma + N_t} + \dfrac{\gamma + d_1 K_{t,\cdot}}{\gamma_1 + N_t}$ WordProbability(word v, topic 0)
  else if $t = 0$ **then**
    Return $\dfrac{N_{0,v} - d_0 K_{0,v}}{\gamma + N_t} + \dfrac{\gamma + d_0 K_{0,\cdot}}{\gamma_0 + N_0} \dfrac{1}{V}$,
  end if
end if

Algorithm 5 Function AddCustomer(word v, document j)

Draw topic t from for a new table using Eq.(9) in PYTM and Eq.(15) in HPYTM, and set $z_{j,k^{new}} = t$,
With probabilities proportional to $\max(0, N^c_{j,v,k} - d)$, sit customer at the k-th table serving word v in document j.
With probabilities proportional to
  $(\alpha + dK_{j,v})$ WordProbability(word v, topic t),
sit a customer at a new table, increment $N_{j,t}$ and call AddCustomer(word v, topic t).

Regression. In *UAI*, pages 411–418, 2008.
[15] J. L. Boyd-Graber and D. Blei. Syntactic Topic Models. In *NIPS*, 2008.
[16] T. Ferguson. A Bayesian analysis of some nonparametric problems. *The Annals of Statistics*, 1:209–230, 1973.
[17] S. Goldwater, T. L. Griffiths, and M. Johnson. Interpolating Between Types and Tokens by Estimating Power-Law Generators. In *NIPS 18*, 2006.
[18] Y. W. Teh. A Hierarchical Bayesian Language Model Based On Pitman-Yor Processes. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 985–992, 2006.
[19] Y. W. Teh. A Bayesian Interpretation of Interpolated Kneser-Ney. Technical Report TRA2/06, School of Computing, National University of Singapore, 2006.
[20] H. Wallach, C. Sutton, and A. McCallum. Bayesian Modeling of Dependency Trees Using Hierarchical Pitman-Yor Priors. In *Proceedings of the Workshop on Prior Knowledge for Text and Langugae (in conjunction with ICML/UAI/COLT)*, pages 15–20, 2008.
[21] T. P. Minka. Estimating a Dirichlet distribution. Technical report, Microsoft, 2000.
[22] J. Pitman and M. Yor. The two-parameter Poisson-Dirichlet distribution derived from a stable subordinator. *Annals of Probability*, 25, 1997.
[23] D. Aldous. Exchangeability and related topics. In *École d' été de Probabilité de Saint-Flour XIII*, 1983.
[24] Escobar and West. Bayesian Density Estimation and Inference using Mixtures. *Journal of the American Statistical Association*, 90, 1995.

APPENDIX

A. GIBBS SAMPLER FOR PROPOSED MODELS

Algorithm 6 Function AddCustomer(word v, topic t)

if PY topic model **then**
  Increment n_{tv}.
else if HPY topic model **then**
  With probabilities proportional to $\max(0, N^c_{t,v,k} - d_1)$, sit customer at the k-th table serving word v in topic j.
  if $t > 0$ **then**
    With probabilities proportional to
      $(\alpha_1 + d_1 K_{t,v})$ WordProbability(word v, topic 0),
    sit a customer at a new table and call AddCustomer(word v, topic 0).
  else if $t = 0$ **then**
    With probabilities proportional to $(\alpha_0 + d_0 K_{t,v}) \frac{1}{V}$, sit a customer at a new table.
  end if
end if

Algorithm 7 Function RemoveCustomer(word v, document j)

With probabilities proportional to $N^c_{j,v,k}$, remove a customer from the k-th table serving word v in document j.
If the k-th table serving word v becomes unoccupied, remove the table from document j and call RemoveCustomer(word v, topic t) if word v at the k-th table is generated from topic t, i.e., $z_{j,k} = t$.

Algorithm 8 Function RemoveCustomer(word v, topic t)

if PY topic model **then**
  Decrement N_{tv}.
else if HPY topic model **then**
  With probabilities proportional to $N^c_{t,v,k}$, remove a customer from the k-th table serving word v in document j.
  If the k-th table serving word v becomes unoccupied, remove the table from document j, and call RemoveCustomer(word v, topic 0) if $t > 0$.
end if

A word distribution of each document, topic, whole corpus can be regarded as a restaurant in the CRP representation of our models. The seating arrangements of customers in a restaurant are sampled by running the two function alternately, **AddCustomer** and **RemoveCustomer**. **AddCustomer** adds the i-th customer into restaurant j shown in Algorithm 5 and 6. **RemoveCustome** removes a customer using menu t from restaurant v shown in Algorithm 7 and 8. $N^c_{j,v,k}$ and $N^c_{j,v,k}$ indicate the number of customers at the k-th table serving word type v in document j and topic t, respectively. **WordProbability(word v, topic t)** indicates the probability that word v is observed in topic t. Note that $N_{j,t}$ in our models indicates the number of tables in which a seated word is generated from topic t in document j, not the number of words generated from topic t in document j.

The parameters α and d of the PY topic model can be estimated by auxiliary variable sampling [18, 19, 24]. Those of the HPY topic model are estimated in a similar way.

First, auxiliary variables x_j, y_{jk}, and z_{jvki} are sampled for each document restaurant $j(=1,\cdots,M)$.

$$\begin{aligned} x_j &\sim Beta(\hat{\alpha}+1, N_j-1)\ (j=0,1,2,\cdots,M), &(16)\\ y_{jk} &\sim Bern(\frac{\hat{\alpha}}{\hat{\alpha}+\hat{d}k})\ (k=1,2,\cdots,K_j-1), &(17)\\ z_{jvki} &\sim Bern(\frac{i-1}{i-\hat{d}})\ (i=1,2,\cdots,n_{jvk}-1), &(18)\end{aligned}$$

Next, given the auxiliary variables, the parameters are sampled.

$$\begin{aligned} d &\sim Beta(\tilde{a}_d, \tilde{b}_d), &(19)\\ \alpha &\sim Gamma(a_\alpha + \sum_{j=1}^{M}\sum_{k=1}^{t_j-1} y_{jk}, b_\alpha - \sum_{j=1}^{M}\log x_j), &(20)\\ \tilde{a}_d &= a_d + \sum_{j=1}^{M}\sum_{k=1}^{t_j-1}(1-y_{jk}), &(21)\\ \tilde{b}_d &= b_d + \sum_{j=1}^{M}\sum_{v,k:n_{jvk}\geq 2}\sum_{i=1}^{n_{jvk}-1}(1-z_{jvki}), &(22)\end{aligned}$$

where a_d, b_d, a_α, and b_α are hyper parameters. We set all hyper parameters to 1.