Hierarchical Clustering on HDP Topics to build a Semantic Tree from Text

Jianfeng Si
Dept. of Computer Science
City University of Hong Kong
Hong Kong, China
jianfsi2@student.cityu.edu.hk

Qing Li
Dept. of Computer Science
City University of Hong Kong
Hong Kong, China
itqli@cityu.edu.hk

Tieyun Qian
State Key Lab of Software Eng.
Wuhan University
Wuhan, China
qty@whu.edu.cn

Xiaotie Deng
Dept. of Computer Science
City University of Hong Kong
Hong Kong, China
csdeng@cityu.edu.hk

ABSTRACT

An ideal semantic representation of text corpus should exhibit a hierarchical topic tree structure, and topics residing at different node levels of the tree should exhibit different levels of semantic abstraction(i.e., the deeper level a topic resides, the more specific it would be). Instead of learning every node directly which is a quite time consuming task, our approach bases on a nonparametric Bayesian topic model, namely, Hierarchical Dirichlet Processes (HDP). By tuning on the topic's Dirichlet scale parameter settings, two topic sets of different levels of abstraction are learned from the HDP separately and further integrated into a hierarchical clustering process. We term our approach as HDP Clustering(HDP-C). During the hierarchical clustering process, a lower level of specific topics are clustered into a higher level of more general topics in an agglomerative style to get the final topic tree. Evaluation of the tree quality on several real world datasets demonstrates its competitive performance.

1. INTRODUCTION

The ever-increasing explosion of online unstructured information puts forward a strong demand to organize online resources in a more efficient way. Hierarchical structures are widely used in knowledge representation, resource organization or document indexing. For example, the web directories organize web pages into a hierarchical tree, providing a comprehensive navigation tool. The discovery of such rich semantic hierarchies from raw data collections becomes a fundamental research in data analysis.

In this paper, we aim to learn a semantic representation of text corpus in the form of a topic tree structure. This can be regarded as a kind of high level summarization on the content of any document collection, as a topic tree expresses a shared conceptualization of interests in certain domain. Such a topic tree functions as an outline to help readers get the main idea of the document collection, which works similarly to the table of content(TOC) of a printed book.

Instead of learning every node directly which is a quite time consuming task, we treat the the construction process of the topic hierarchy mainly as a two-phase task: 1) the identification or definition of topics; 2) the derivation of hierarchical relationships between or among the topics. Our approach is built on a nonparametric Bayesian topic model, namely, Hierarchical Dirichlet Processes(HDP)[13]. By tuning on the topic's Dirichlet scale parameter settings, two topic sets are learned from the HDP separately with different levels of semantic abstraction. One as the top level which represents a small collection of general topics and another as the down level which corresponds to a relatively larger collection of specific topics. Topics from the two different sets exhibit different topic granularity on semantic representation. Based on these, we can efficiently construct the "middle" level topics directly without modeling them explicitly. As a result, the hierarchical structure comes out straightforwardly and the whole learning process speeds up.

Fig.1 shows a sub-tree of our learned topic tree on the JACM[1] dataset which contains 536 abstracts of the Journal of the ACM from 1987-2004. There are two super topics on this sub-tree, one as system related topic and another as database related topic. When we look into the database topic, we find that it is further divided into 3 specific aspects, which are "Scheme Design", "DB Robust" and "Transaction Control". Also we observe that the super topic mainly contain some widely used function words or stop words, resulting in the most "general" topic as the root.

The organization of the paper is as follows. In Section 2 we briefly introduce the related works. We define in Section 3 our problem formulation and propose the HDP-C model. Our experiment on several real world datasets is presented in Section 4, and we conclude our work in Section 5.

[1]http://www.cs.princeton.edu/∼blei/downloads/jacm.tgz

VLDS'12 August 31, 2012. Istanbul, Turkey.

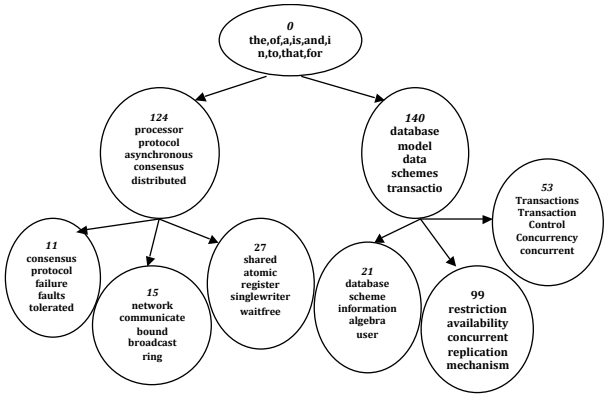

Figure 1: A sub-tree learned from JACM dataset

2. RELATED WORK

Topic modeling as a task of document modeling has attracted much attention in recent years. But much work just focuses on inferring hidden topics as a flat cluster over the term space[1]. One of the basic and also the most widely used ones is the Latent Dirichlet Allocation(LDA)[4]. LDA can learn a predefined number of topics under a bag-of-topics. A question that comes with LDA is how many topics we should take for the model to estimate. To address this problem, another nonparametric Bayesian model, namely, Hierarchical Dirichlet Processes(HDP), was introduced and adopted[13]. In both LDA and HDP, no relationship is defined explicitly between topics during learning, and the estimated topics are "flat".

Comparing to the "flat" models, hierarchical modeling of topics can learn more accurate and predictive models, because hierarchical modeling is more likely to capture the generative nature of text collections([11],[9],[10]). The Correlated Topic Models(CTM)[3] which is an extension of LDA, captures the relations between topics, but it only models the pair-wise correlations. The Pachinko Allocation Model (PAM)[10] uses a directed acyclic graph(DAG) structure to learn and represent the topic correlations. PAM connects the words and topics on a DAG, where topics reside on the interior nodes and words reside on the leaf nodes, but PAM is unable to represent word distributions as parents of other word distributions. Zavitsanos *et al.*[16] learned a topic tree using LDA for each level, by starting with one topic at level 0 and incrementing the number of topics in each further iteration/level, and used the symmetric KL divergence between neighbor hierarchies to indicate the convergence. The basis of their work is quite similar to ours, but they learn the predefined number of topics for each level explicitly. The Hierarchical Latent Dirichlet Allocation(hLDA)[2] is the first model to learn a tree-structure topic distribution. In hLDA, each document spans a single path starting from the root node to a leaf node of the tree with a predefined depth, then words of that document are generated via topics on that path. This model arranges the topics into a tree, with the desideratum that more general topics should appear near the root and more specialized topics should appear near the leaves[8].

Hierarchical HDP(hHDP)[15], on the other hand, learns a topic hierarchy from text by defining an HDP for each level of the topic hierarchy. The topic hierarchy is learned in a bottom-up fashion: starting with a document corpus, the leaf topics are inferred first, then, the word distributions of all leaf topics make up the observations for the estimation of the next up level. The procedure repeats until the root topic is inferred. In hHDP, the parent/child relationships between up/down topics are not clearly identified. Also, this recursive definition of HDP is likely to suffer from the low time efficiency.

Our work is also built on HDP, but only for the most root level and lowest level topics. We construct the interior level topics by a simple clustering algorithm which is quite efficient, and an evaluation on the final tree quality also demonstrates its competitive performance. Different from traditional hierarchical clustering, which gives a hierarchical partition on documents[12], points in our hierarchical clustering refer to the word distributions.

Evaluation on the learned tree is also a relevant and an interesting topic. In [14], an ontology evaluation method is proposed, and we adopt the same evaluation method for our work here due to the close relevance.

3. HDP-C MODEL

In this section, we firstly analyze the impact of the scale parameter of Dirichlet distribution, then introduce the HDP briefly, followed by describing our clustering algorithm(HDP-C) in detail.

3.1 Dirichlet distribution and its scale parameter η

Our HDP-C model is built upon the HDP, the idea of which lies on tuning on topic's Dirichlet scale parameter settings, so as to help control the topic granularity which is used to model the text content. Dirichlet distribution is a multi-parameter generalization of the Beta distribution, and it defines a distribution over distributions, i.e., the samples from a Dirichlet are distributions on some discrete probability space. The Dirichlet is in the exponential family, and is a conjugate prior to the parameters of the multinomial distribution which facilitates the inference and parameter estimation.

Let θ be a k-dimensional Dirichlet random variable with $\theta_i \geq 0, \sum_{i=1}^{k} \theta_i = 1$, it lies in the k-1 dimensional probability simplex with the following probability density:

$$p(\theta|\alpha) = \frac{\Gamma(\sum_{i=1}^{k} \alpha_i)}{\prod_{i=1}^{k} \Gamma(\alpha_i)} \theta_1^{\alpha_1 - 1} \cdots \theta_k^{\alpha_k - 1} \quad (1)$$

where the parameter α is a k-vector with components $\alpha_i > 0$, and α_i can be interpreted as "prior observation counts" for events governed by θ_i. Furthermore, $\alpha_0 = \sum_i \alpha_i$ is called the scale or concentration parameter with the base measure $(\alpha_1/\alpha_0, \cdots, \alpha_i/\alpha_0)$, and $\Gamma(x)$ is the Gamma function.

A frequently used special case is the symmetric Dirichlet distribution, where $\alpha_1 = \cdots = \alpha_k = \eta$, indicating that we have no idea of which components are more favorable in our prior knowledge, and as a result, we use a uniform base measure. The scale parameter η plays an important role in controlling the variance and sparsity of the samples. For example, when $\eta = 1$, the symmetric Dirichlet distribution is equivalent to a uniform distribution over the k-1 probabil-

ity simplex, i.e., it is uniform over all points in its support. Values of the scale parameter above 1 prefer variates that are dense, evenly-distributed distributions, i.e., all probabilities returned are similar to each other. Values of the scale parameter below 1 prefer sparse distributions, i.e., most of the probabilities returned will be close to 0, and the vast majority of the mass will be concentrated on a few of the probabilities.

Fig.2 depicts five samples for each different η setting ($\eta = 0.1, \eta = 1, \eta = 10$) from a 10-dimensional Dirichlet distribution. Obviously, $\eta = 0.1$ leads to getting samples biasing probability mass to a few components of the sampled multinomial distribution; $\eta = 1$ leads to a uniform distribution, and $\eta = 10$ leads to a situation that all samples are closer to each other(in another word, each component gets similar probability mass).

In a word, a smaller η setting encourages fewer words to have high probability mass in each topic; thus, the posterior requires more topics to explain the data. As a result, we get relative more specific topics. Based on this characteristic, we can further obtain two topic sets with different granularity measure, corresponding to the up-bound and low-bound topic sets in the sense of granularity .

3.2 Hierarchical Dirichlet Processes

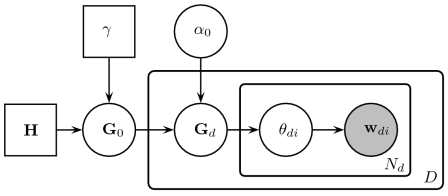

Figure 3: HDP graphical model

HDP is a nonparametric hierarchical Bayesian model which can automatically decide the number of topics. Fig.3 shows the graphical model proposed in [13]. A global random probability measure G_0 is distributed as a Dirichlet process(DP)[5] with a concentration parameter γ and the base probability measure H. For each document d, a probability measure G_d is drawn from another Dirichlet process with concentration parameter α_0 and base probability measure G_0, where:

$$G_0|\gamma, H \sim DP(\gamma, H) \tag{2}$$

$$G_d|\alpha_0, G_0 \sim DP(\alpha_0, G_0). \tag{3}$$

The Chinese restaurant franchise is a good metaphor for HDP. Assume there is a restaurant franchise holding a global shared menu of dishes across all restaurants. At each table of each restaurant, only one dish is served from the global menu selected by the first customer who sits there, and it is shared among all customers who sit in the same table. The same dish can be served for multiple tables in multiple restaurants. For the document modeling scenario, each document corresponds to a restaurant, each word corresponds to a customer, and topics are dishes of the global shared menu. As a result, using HDP, we can finally learn a set of global topics, and each document should cover a subset of the topics.

For our particular application, the base measure H is the Dirichlet distribution over term space, i.e., $H \sim Dirichlet(\eta)$.

```
learn a low-level topic set from HDP with η = 0.125, M = {t_l1, ··· , t_l|M|}
learn a top-level topic set from HDP with η = 1.0, N = {t_u1, ··· , t_u|N|}
for each t_li ∈ M, find its "closest" topic t_uj ∈ N:
for i = 1 to |M| step 1 do
    t_uj = argMin_{t_uj ∈ U}(D(t_li, t_uj))
    t_uj.childList.add(t_li)
    t_uj.nchild++
end for
cluster the top-lever topics's children in an agglomerative hierarchical clustering style:
for  i = 1 to |N| step 1 do
    while t_ui.nchild > 3 do
        find the most closest children pair (t_x, t_y)
        merge (t_x, t_y) into a new inner topic t_m
        t_ui.childList.remove(t_x)
        t_ui.chileList.remove(t_y)
        t_ui.chileList.add(t_m)
        t_uj.nchild − −
    end while
end for
```

Algorithm 1: Hierarchical clustering algorithm(HDP-C) from low-level topics to the top level

So, the scale parameter η is used as the granularity indicator in our experiment.

3.3 Hierarchical clustering on topics

Based on the top-level topic and down-level topic sets, we use an agglomerative clustering algorithm to build up the interior nodes.

The top level topics give a raw partition on the topic distribution and can be directly combined to form the root topic node. Also it can help supervise the agglomerative clustering process for the low level topics. So, the whole algorithm is divided into three phrases:

1. assign all low level topics into their immediate top level topics;
2. for all subsets of low level topics indexed under each top level topic, an agglomerative clustering process is invoked;
3. finally, define the root topic node as a combination of top level topics.

So, the size of the final topic tree is determined by the number of topics on the top level and down level, and we decide the depth of the tree afterward according to user requirement by truncating unwanted lower levels.

During the clustering process, a pair of "closest" topics are merged for each iteration. The whole algorithm is presented as Algorithm 1.

Algorithm 1 use a "bottom up" approach instead of "top down" approach. That is because we have no idea on how to split a topic distribution into two sub topics, while merging of two sub topics into one is much more straightforward.

4. EXPERIMENT

In this section, we set up our golden line[14] topic trees from hierarchical data collections, test the tree quality with

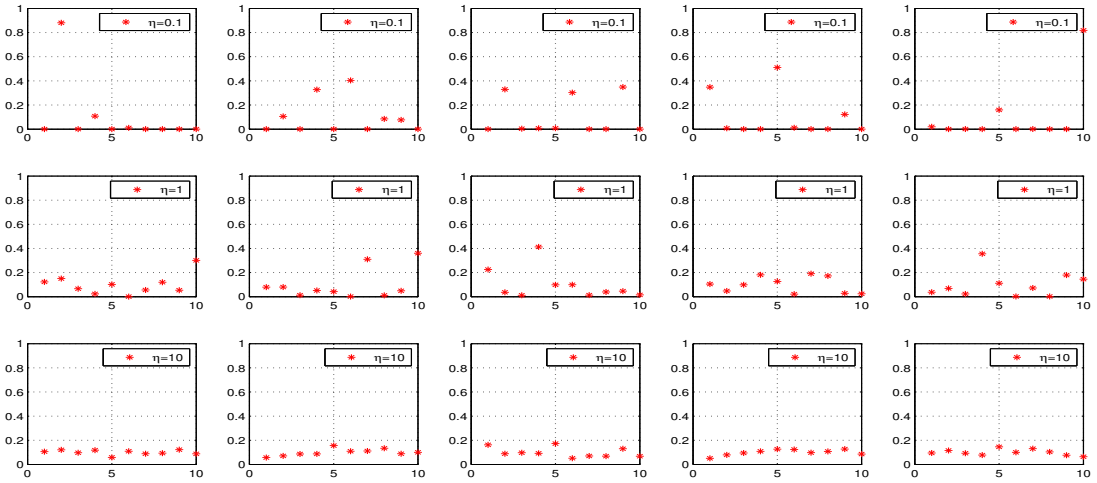

Figure 2: Samples of Dirichlet distributions with different scale parameter settings (x-axis are the components [1-10], y-axis is the probability)

Table 1: General Statistics on Data sets

Dataset	rootId	#Docs	TermSpace	#Golden Topics
JACM	N/A	536	1809	N/A
wiki1	2020309	748	7032	36
wiki2	2337326	420	7047	46
wiki3	2319090	310	5516	29
dmoz1	38825	781	15082	68
dmoz2	39220	289	6290	16
dmoz3	38997	625	13907	87

different probability metrics as distance measure of our clustering, and use hLDA as the baseline to compare the tree quality in the terms of *Precision, Recall, F-Score.*

4.1 Data set

To evaluate the learned tree quality, we use the Wikipedia (WIKI) dataset from the third Pascal Challenge on Large Scale Hierarchical Text Classification(LSHTC3)[2] and the Open Directory Project(DMOZ) dataset from Second Pascal Challenge onLarge Scale Hierarchical Text Classification (LSHTC2)[3]. Totally, we obtain three datasets from each of these two sources. All these datasets contain a hierarchy file defining the organization of each document into a hierarchical tree structure. Each document is assigned to one or more leaf nodes. The general statistics over these datasets and the JACM one are shown in Table 1.

Given the hierarchical relationship, we randomly choose some sub-trees from it and build their corresponding golden line topic trees according to the term frequencies from documents' assignments.

4.2 Scale effect of η settings on HDP

The scale parameter η is used as the granularity indicator in our experiment. Fig.4 shows how the count of topics learned from HDP on our datasets changes under different

[2]http://lshtc.iit.demokritos.gr/LSHTC3_DATASETS
[3]http://lshtc.iit.demokritos.gr/LSHTC2_datasets

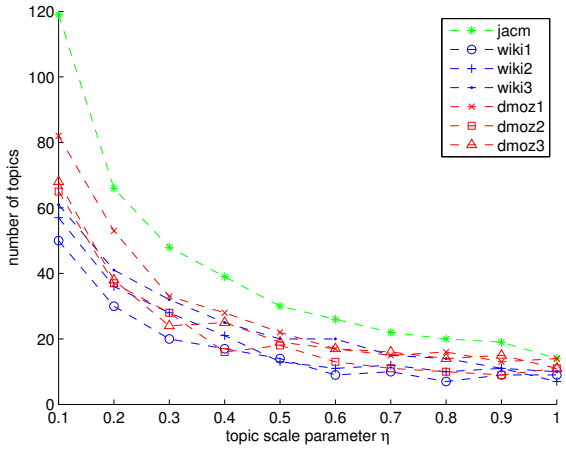

Figure 4: Topic counts estimated by HDP vs η settings ranging in [0.1,1]

η settings. Fig.5 shows the distribution variances of the learned topic collection from HDP on our datasets. For each η setting, the inner variance of that learned topic collection is measured by taking an average on the symmetric KL divergence between every topic and the centroid distribution of that collection. As shown, the variances almost drop consistently while the η ranges from 0.1 to 1.0. This observation is consistent with the η's consideration.

4.3 Evaluation method

Given the learned topic tree and the golden line topic tree, we want to measure how close these two structures are in a quantitative metric. We use the ontology evaluation method proposed in [14], which can capture, quite accurately, the deviations of learned structure from the gold line by means of ontology alignment techniques. In this method, the ontology

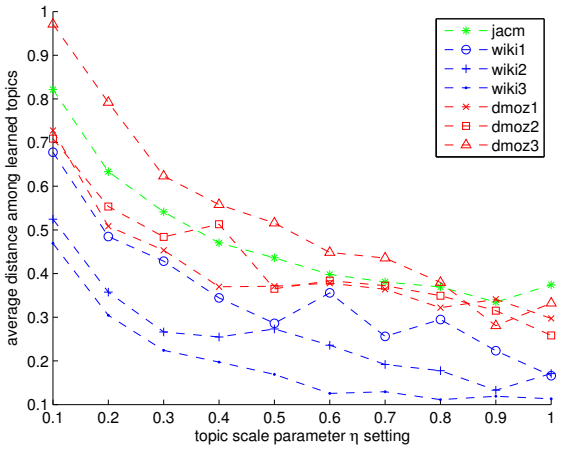

Figure 5: Average Symmetric KL distance between learned topics under different Dirichlet scale parameter setting)

concepts are defined as vector space representations which are the same as ours. We summarize this method as follows:

1. set up a one-to-one matching collection $M = \{1, \cdots, |M|\}$ based on the dissimilarity measure between nodes of the learned tree $L = \{t_{l1}, \cdots, t_{l|L|}\}$ and nodes of the golden tree $G = \{t_{g1}, \cdots, t_{g|G|}\}$, where $|M| = smaller(|G|, |L|)$;
2. for each matching $m = (t_{li}, t_{gj})$, compute the Probabilistic Cotopy Precision(PCP_m) and Probabilistic Cotopy Recall(PCR_m);
3. take a weighted average of PCP and PCR to compute the P, R and the F-score, the weight is the similarity between the nodes of the matching pair.

The corresponding formulas needed for steps 2 and 3 above are shown in the following:

$$PCP_m = \frac{|CS(t_{li}) \bigcap CS(t_{gj})|}{|CS(t_{li})|} \tag{4}$$

$$PCR_m = \frac{|CS(t_{li}) \bigcap CS(t_{gj})|}{|CS(t_{gj})|} \tag{5}$$

$$TVD = \frac{1}{2} \sum_i |p(i) - q(i)|, TVD \in [0, 1]. \tag{6}$$

$$P = \frac{1}{|M|} \sum_{m=1}^{|M|} (1 - TVD_i) PCP_m \tag{7}$$

$$R = \frac{1}{|M|} \sum_{m=1}^{|M|} (1 - TVD_i) PCR_m \tag{8}$$

$$F = \frac{P * R}{P + R} \tag{9}$$

In above equations, the $CS(t)$ is the Cotopy Set of node t, which includes all its direct and indirect super and subtopics

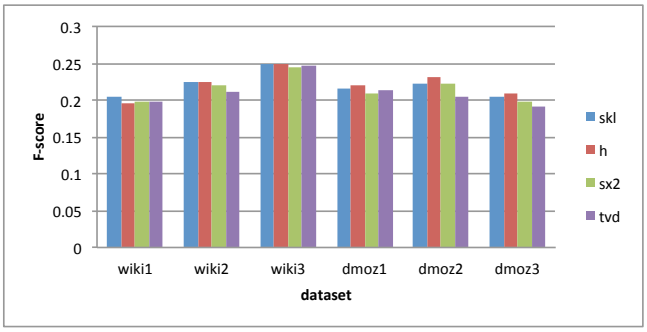

Figure 6: F-Scores of learned tree on different probability metrics for hierarchical clustering distance measure

and itself. TVD is the Total Variational Distance[7], it is used to measure the dissimilarity of two probability distributions.

4.4 Probability metrics for hierarchical clustering's distance measure

Gibbs *et al.*[6] reviewed on ten of the most popular probability metrics/distances used by statisticians and probabilists. We choose four of these and test their influence on our learned tree quality. The selected metrics are symmetric KL divergence(D_{skl}), Hellinger distance(D_h), and symmetric χ^2 distance($D_{s\chi^2}$) whose definitions are given below, as well as the TVD(D_{tvd}) defined in Equation.6.

$$KL(p, q) = \sum_{i=1}^{V} p_i log(\frac{p_i}{q_i})$$
$$D_{skl}(p, q) = 1/2[KL(p, q) + KL(q, p)] \tag{10}$$

$$D_h(p, q) = (\sum_{i=1}^{V} (\sqrt{p_i} - \sqrt{q_i})^2)^{1/2} \tag{11}$$

$$D_{\chi^2}(p, q) = \sum_{i=1}^{V} \frac{(p_i - q_i)^2}{q_i}$$
$$D_{s\chi^2}(p, q) = 1/2[D_{\chi^2}(p, q) + D_{\chi^2}(q, p)] \tag{12}$$

Fig.6 plots the F-Scores of our learned topic trees from all the datasets with different distance measures. As observed from this figure, all choices perform similarly, so we choose the symmetric KL divergence as our distance measure of the clustering in further experiment. The relationship between those measures can be found in [6].

4.5 Performance of HDP-C

We use hLDA as the baseline to learn a 4-depth topic tree with the scale parameter settings $\eta = 1.0, \eta = 0.5, \eta = 0.25$, and $\eta = 0.125$ for each level. This is then compared to the top 4-depth sub-tree of our learned tree through HDP-C. To be consistent with hLDA's η settings, the top-level topics are learned with $\eta = 1.0$ and down-level topics are learned with $\eta = 0.125$. We use the default value for other parameters: $\gamma = 1.0, \alpha_0 = 1.0$, and the max iteration is 1000. For hLDA, we set the max iteration to be 2000 due to that it gets a bigger learning space than HDP.

Table 2: Comparison of tree quality with baseline

	hLDA			*HDP − C*			Enhancement(%)		
	P	R	F	P	R	F	P	R	F
wiki1	0.323	0.459	0.189	0.346	0.464	0.198	7.1	1.1	4.8
wiki2	0.333	0.461	0.193	0.401	0.460	0.214	20.4	-0.2	10.9
wiki3	0.335	0.532	0.205	0.432	0.574	0.247	29.0	7.9	20.5
dmoz1	0.350	0.449	0.197	0.437	0.432	0.217	24.9	-3.8	10.2
dmoz2	0.234	0.472	0.156	0.389	0.497	0.218	66.2	5.3	39.7
dmoz3	0.302	0.364	0.165	0.445	0.392	0.208	47.4	7.7	26.1

The evaluation result is given in Table.2(Note that JACM dataset is not included here due to the lack of golden line topic tree). In terms of the *F-Score*, our approach performs, on average, 12.1% better on wiki datasets and 25.3% better on dmoz datasets. One reason is that, for hLDA each document only spans a single path from the root to a leaf node, which is a quite tough restriction in the mixture of topics for each document. In contrast, our approach does not make any prior restriction on each document's topic choice. Actually, each document can span any arbitrary sub-tree, which can explain its generative nature.

Besides, we observe from Table 2 that the improvement in terms of P is much better that R, which indicates that our approach is more preferable to those tasks which care the precision more.

5. CONCLUSIONS

This paper builds a semantic topic tree representation for a document collection based on a non-parametric Bayesian topic model. Only the up-bound and low-bound topic sets are directly inferred with the tuning on topic's Dirichlet scale parameter for different levels of abstraction. A hierarchical clustering algorithm(HDP-C) is proposed to derive the middle level topics in order to construct the final topic tree. Our experimental study on several real world datasets shows competitive performance of our approach.

6. ACKNOWLEDGMENTS

The work described in this paper has been supported by the NSFC Overseas, HongKong & Macao Scholars Collaborated Researching Fund (61028003) and the Specialized Research Fund for the Doctoral Program of Higher Education, China (20090141120050).

7. REFERENCES

[1] D. M. Blei. Introduction to probabilistic topic models. *Communications of the ACM*, 2011.

[2] D. M. Blei, T. L. Griffiths, and M. I. Jordan. The nested chinese restaurant process and bayesian nonparametric inference of topic hierarchies. *J. ACM*, 57:7:1–7:30, February 2010.

[3] D. M. Blei and J. D. Lafferty. A correlated topic model of science. *AAS*, 1(1):17–35, 2007.

[4] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022, Mar. 2003.

[5] T. S. Ferguson. A Bayesian Analysis of Some Nonparametric Problems. *The Annals of Statistics*, 1(2):209–230, 1973.

[6] A. L. Gibbs, Francis, and E. Su. On choosing and bounding probability metrics. *Internat. Statist. Rev.*, pages 419–435, 2002.

[7] A. L. Gibbs and F. E. Su. On Choosing and Bounding Probability Metrics. *International Statistical Review*, 70:419–435, 2002.

[8] T. Hofmann. The cluster-abstraction model: unsupervised learning of topic hierarchies from text data. In *Proceedings of the 16th international joint conference on Artificial intelligence - Volume 2*, IJCAI'99, pages 682–687, San Francisco, CA, USA, 1999. Morgan Kaufmann Publishers Inc.

[9] W. Li, D. Blei, and A. Mccallum. Nonparametric Bayes Pachinko Allocation. In *UAI 07*, 2007.

[10] W. Li and A. Mccallum. Pachinko allocation: Dag-structured mixture models of topic correlations. In *In Proceedings of the 23rd International Conference on Machine Learning*, pages 577–584, 2006.

[11] D. Mimno, W. Li, and A. Mccallum. Mixtures of hierarchical topics with pachinko allocation. In *In Proceedings of the 24th International Conference on Machine Learning*, pages 633–640, 2007.

[12] F. Murtagh. A Survey of Recent Advances in Hierarchical Clustering Algorithms. *The Computer Journal*, 26(4):354–359, 1983.

[13] Y. W. Teh, M. I. Jordan, M. J. Beal, and D. M. Blei. Hierarchical Dirichlet processes. *Journal of the American Statistical Association*, 101(476):1566–1581, 2006.

[14] E. Zavitsanos, G. Paliouras, and G. Vouros. Gold standard evaluation of ontology learning methods through ontology transformation and alignment. *Knowledge and Data Engineering, IEEE Transactions on*, 23(11):1635 –1648, nov. 2011.

[15] E. Zavitsanos, G. Paliouras, and G. A. Vouros. Non-parametric estimation of topic hierarchies from texts with hierarchical dirichlet processes. *J. Mach. Learn. Res.*, 999888:2749–2775, Nov. 2011.

[16] E. Zavitsanos, S. Petridis, G. Paliouras, and G. A. Vouros. Determining automatically the size of learned ontologies. In *Proceedings of the 2008 conference on ECAI 2008: 18th European Conference on Artificial Intelligence*, pages 775–776, Amsterdam, The Netherlands, 2008. IOS Press.