Learning Topic Structure in Text Documents using Generative Topic Models

Nitish Srivastava
CS 397 Report
Advisor: Dr Hrish Karnick

***Abstract*—We present a method for estimating the topic structure for a document corpus by combining the use of efficient clustering algorithms with generative topic models. Our method builds the topic structure as a DAG in a layer-wise manner by estimating the number of topics in each layer and constructing topic models for them. It discovers subsumption relations between topic nodes in consecutive layers by mapping the problem to finding maximum edge-weighted cliques on small and sparse graphs. We analyze the sparsity of the induced graphs and give bounds on the running time of our algorithm. Most methods for solving this problem use variants of hierarchical dirichlet procesess to provide a nonparametric prior on the number of topics and estimate the number as the topic model is built. While these models have been shown to perform well under certain metrics, the estimated number of topics is often quite large, limiting the utility of the topic structures built. Our approach is to build structures where the number of topics would be smaller and comparable to that in structures built by human experts. We evaluate our method using real world text datasets.**

***Keywords*-Data management; Data models; Hierarchical systems; Pattern clustering methods**

I. Introduction

Topic models have been shown to be effective tools for analysis of document corpora. Learning the topic structure is an important problem in this regard. Our method builds arbitrary DAG topic structures over a collection of documents where nodes represent topics and edges point from a topic to a more specific sub-topic. Key contributions are the estimation of number of topics in each layer of the structure and discovery of subsumption relations between topic nodes in adjacent layers as a max-clique problem.

Topic models have been proposed which automatically learn the number of topics using various forms of Hierarchical Dirichlet Processes (HDP, Teh et al. [1]) such as nonparametric bayes PAM [2]. The key idea here is to *Description of HDP, PAM, NPB-PAM*. *Criticism* We explore a different strategy for topic structure extraction. We separate the generative topic model from the topic ontology. We propose a method which separates the processes of learning a topic structure from learning the topic model.

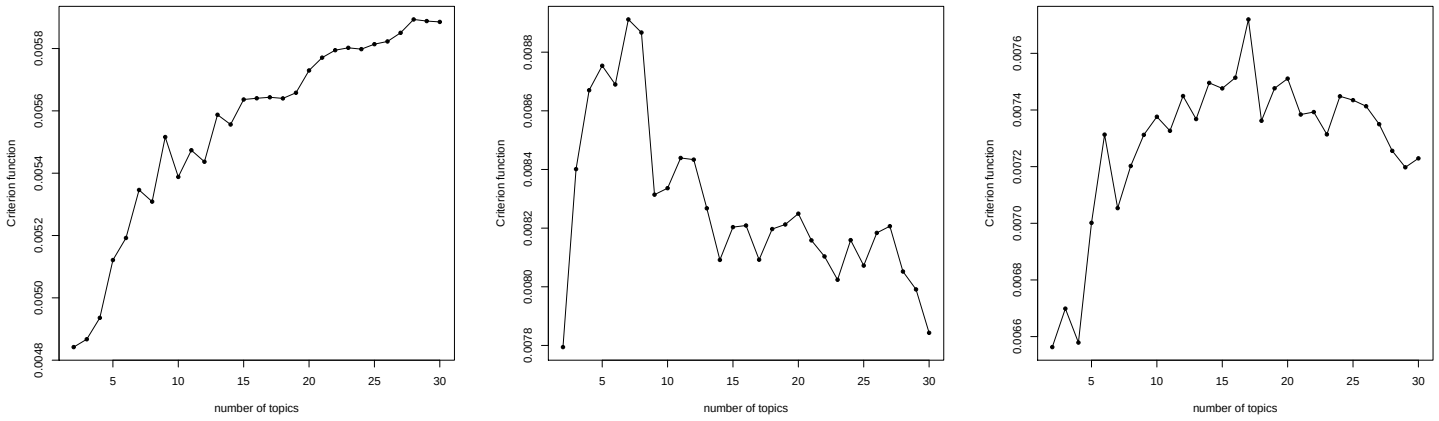
We build the topic structure using simple clustering techniques combined with single layer topic models. These structures can then be used by various sophisticated topic models which require the number of topics and hierarchy structure to be specified. Models such as hierarchical Latent Dirichlet Allocation (hLDA, [3]), Correlated Topics Model (CTM, [4]) and Pachinko Allocation Model (PAM, [5]) and mixture model extensions of these are some such methods.

Document clustering is a widely studied field. Zhao et al. [6] give an excellent comparison of several clustering paradigms. We use a recursive bisection based clustering algorithm [7] to k-cluster a dataset and evaluate the quality of each cluster. We use this to approximate the number of topics at different depths in the DAG. Successive levels of depth in the DAG ("layers") represent more fine-grained topics. We learn single-layer topic models for each depth. The topics in consecutive layers are then linked by subsumption relations. We cast the problem of discovering subsumption relations into finding a maximum edge-weighted clique over a sparse topic graph. Besides finding subsumption relations, our formulation also merges very similar topics in the same layer to further prune the topic structure and make up for errors made in the clustering phase.

In order to validate our results, we use the 20 newsgroups comp5 (ngcomp5) and the NIPS abstracts datasets. These datasets have single level categories. In order to do a more meaningful evaluation, we collected a hierarchical dataset from a crawl of the webpages linked from the Open Directory Project (ODP). This allows us to compare against human-expert determined topic structures.

II. Estimating number of topics

In this section, we describe a simple approach to obtain a *rough* estimate of the number of the topics given a document corpus. Nonparametric variants of topic models which use Hierarchical Dirichlet Processes (HDP) such as Latent Dirichlet Allocation [1] and Pachinko Allocation Model (NPB-PAM [2]) have been used to estimate the number of topics while building the corresponding topic model. These methods place a non-parametric prior on the number of topics and estimate the number as the model is built. These methods have been shown to work well in terms of evaluation metrics which involve likelihood estimates (empirical or otherwise) or classification accuracies on held-out data. Infering the number of topics using HDPs gives more specific topics (as qualitatively shown for the case of NPB-PAM). However, the average number of topics estimated is often very high. For example NPB-PAM discovers 179 sub-topics in the 20newsgroups comp5 dataset which contains 5 topics. Such models are good for application domains where the topic model is meant for tasks where the



(a) ODP (b) NIPS (c) ODP/Computers

Figure 1. Plot of Criterion function vs Number of topics

discovered topics themselves serve only as latent variables. It is not clear if these topics would correspond to a human determined ontology. Our goal is to generate a topic structure with typically fewer topic nodes and more semblance to a topic hierarchy that could be applied to domains which use the actual topic structure for organizing unstructured text information.

We begin by obtaining an approximate number of topics using a clustering algorithm which uses repeated bisections. Zhao and Karypis [6] have shown that this method produces better clustering than agglomerative or graph based methods. A key characteristic of this approach is that it uses a global criterion function whose optimization drives the entire clustering process. The criterion function used is

$$\text{maximize} \sum_{i=1}^{k} \sqrt{\sum_{v,u \in S_i} u.v}$$

where each document u is represented as a vector of normalized tf values.

We k-cluster the set of documents for each $k \in \{2, 3, \ldots, K\}$, where K is the maximum number of topics to be allowed in any layer of the topic DAG. Computing a K-clustering takes $O(NNZ * log(K))$ time where NNZ is the number of non-zero entries in the document similarity matrix. In the process of K clustering the dataset, the algorithm also produces k-clusters for all $k < K$ since the method performs repeated bisections. Hence, computing all clusterings from 2 to K also takes $O(NNZ * \log(K))$ time. Each clustering C_k corresponds to a set of clusters $\{C_k^1, C_k^2, \ldots, C_k^k\}$. The quality of each C_k is evaluated using the function

$$F(\overline{\mu_{int}}, \overline{\mu_{ext}}, \overline{\sigma_{int}}) = \frac{1}{\sum_{i=1}^{k} |C_k^i|} \sum_{i=1}^{k} |C_k^i| \frac{\mu_{ext}^i \sigma_{int}^i}{\mu_{int}^i} \quad (1)$$

Here
$\overline{\mu_i}$ denotes the vector of average intra-cluster similarity.
$\overline{\mu_e}$ denotes the vector of average inter-cluster similarity.
$\overline{\sigma_i}$ denotes the vector standard deviation in intra-cluster similarity.
All similarities refer to cosine similarities.

A lower value of the function indicates a better clustering. The intuition behind the criterion function is to have low similarity with documents outside the cluster and high similarity, with low standard deviation inside. This is to ensure that the clusters are *tight*. The ratio is weighed by the size of the cluster. This function is computed for each k-clustering. This gives a sequence of quality values $q_1, q_2, \ldots, q_K$ corresponding to the sequence of topic numbers. A low value implies a good clustering. Let $\{m_1, m_2, \ldots, m_l\}$ be the sequence of topic numbers corresponding to local minima in the sequence of quality values. These minima represent the regions where the number of topics are such that the clustering produced is locally optimal. These are potentially the number of topics in at increasing depth in the topic structure. The intuition behind this is as follows. Suppose that a document corpus has c topics. If the corpus was clustered into a $c-1$ or $c+1$ clustering, some of the actual c clusters will have to redistribute their documents to fit into a $c-1$ or $c+1$ clustering. This will decrease the *tightness* of the clusters which will be captured by the criterion function in 1. We expect that the value of the function would be higher on either side of a *good* clustering. Figure 1 shows plots of $q_1, q_2, \ldots, q_{30}$ for different datasets.

With this heuristic in place, the goal of the clustering step is to determine $\{m_1, m_2, \ldots, m_L\}$, the sequence of approximate number of topics in each layer. For example, in Figure 1(b) the sequence of minima is $\{6, 9, 14, 18, \ldots\}$. These will be chosen as the approximate number of topics

in the topic structure corresponding to the NIPS abstracts dataset.

The underlying assumption that mutually exclusive document clusters are representative of topics is not entirely accurate since the actual topic structure may be more complex and documents may contain content that could be best described as a mixture of different topics. Topics may exhibit significant correlations. However, the purpose of this clustering is not to discover this rich underlying structure but to get a rough estimate of the number of topics in each layer of the topic structure. This estimate is important because it gives a starting point for the structure building algorithm. A more accurate topic number along with subsumption relations between them is discovered later. Table **??** shows the top few words in the clusters corresponding to the local minima of the criterion function. While the clusters may not be extremely accurate, they are good enough to justify their use as approximatations. Section xxx compares the final topic numbers and structure with these approximations.

III. Building the DAG

A. Building Single Layer topic models

The next step in building the topic structure is to find topic models for each value of the number of topics $\{m_1, m_2, \ldots, m_l\}$ found in the previous section. We treat each layer independently and build single layer topic models for them. Our method allows the use of any statistical topic model as long as it is possible to infer topic proportions present in each document under that model. In this paper, we demonstrate our method using simple Latent Dirichlet Allocation [8] though it is possible to use more sophisticated models. The simplicity of the model allows us to demonstrate the key idea behind the process of discovering subsumption relations more effectively. The framework of building independent models and then subsuming consecutive layers has been previously used by Zavitsanos et al. [9] who demonstrated its use in building document ontologies using LDA.

Latent Dirichlet Allocation is a generative probabilistic model in which documents are represented as random mixtures over latent topics. Topics are themselves modeled as an infinite mixture over a set of topic probabilities. The generative process can be summarized as:

We use variational methods described in Blei et al. [8] to infer topic proportions in each document. We have the sequence $(m)_{i=1}^{L}$ of number of topics estimated from the clustering step. For each element m_i of this sequence, we build an LDA topic model T_i. For each document d of the corpus, the variational inference method gives a posterior dirichlet parameter γ_d, where the proportion of topic u is represented by the component $\gamma_{d,u}$. One way to interpret this situation is to define a probability function P over the set of topics $\{u_1, u_2, \ldots, u_{m_i}\}$, where $P(u_i)$ denotes the probability of occurence of topic u_i in a random document. Given the document corpus D, this can be estimated as,

$$\begin{aligned} P(u_i) &= \frac{1}{|D|} \sum_{d \in D} \gamma'_{d,u_i} \\ \gamma'_{d,u_i} &= \frac{\gamma_{d,u_i}}{\sum_{l=1}^{m_i} \gamma_{d,u_l}} \end{aligned}$$

The joint probability distribution $P(u_i, u_j)$ can be estimated as,

$$P(u_i, u_j) = \frac{1}{|D|} \sum_{d \in D} \gamma'_{d,u_i} \gamma'_{d,u_j} \tag{2}$$

The joint probabilities are a measure of co-occurence of different topics. This idea is exploited next for discovering subsumption relations.

B. Subsumption as a maximum weight clique problem

Now that we have the topic models for each layer, the next step is to find subsumption relations between consecutive layers. The problem of discovering subsumption relations in ontologies has been widely studied. One of the most relevant works in the context of building topic structures is by Zavitsanos et al. [9] who use the idea of conditional independence between sub-topics given a candidate parent topic to decide subsumption relations. In their method, if $\{u_1, u_2, \ldots, u_m\}$ is a topic layer and $\{v_1, v_2, \ldots, v_n\}$ is the layer of topics just below it, then each pair (v_i, v_j) is assigned a parent u_k if the occurence of u_k in a document makes the occurence of topics v_i and v_j conditionally independent. The intuition is that if the co-occurence of topics v_i and v_j is nullified once we know the parent u_k occurs, then the parent captures what is common between the topics and is therefore a good choice to subsume v_i and v_j. Such a criterion for subsumption is reasonable, but it suffers from a constraint that it decides the subsumption relations based on pair-wise independence alone. In this paper, we use a different criterion and a more general graph framework that captures pair-wise relations and uses a clique based method to build higher order sub-topic sets and then determines subsumption relations.

Let there be m parent topics and n child topics in some pair of consecutive layers. For each child topic v_i we find u_k such that $P(v_i|u_k)$ is maximum, where the probability P is defined as in Eq. 2 and 2. The topic v_i is then directly subsumed under u_k. This process associates each child topic with exactly one parent, building a tree structure. Next, we discover pair-wise subsumption relations, i.e., for each pair of distinct child topics (v_i, v_j) we find the parent topic u_k which best subsumes them. We later use pair-wise relations to determine global subsumption relations. We would like u_k to subsume v_i and v_j if they could be considered sub-topics of u_k. For them to be sub-topics, two conditions must hold. v_i and v_j should be *associated* with u_k but at the same time they must be *sufficiently separate* so that they can be

considered separate sub-topics. These two conditions can be interpreted as demanding that $P(v_i|u_k)$ and $P(v_j|u_k)$ should both be high but $P(v_i, v_j|u_k)$ must be low, i.e. the joint probability that v_i and v_j occur together given u_k must be low. In other words, each sub-topic individually must have a high conditional probability of occurence given that the parent topic occurs (indicating that the subtopics capture a part of the parent topic's vocabulary) but at the same time both the sub-topics must not occur together very often given the parent (if they do occur, they are not separate enough to be considered individual sub-topics). In this sense, the occurence of v_i and v_j must be negatively correlated given u_k. Hence for every (v_i, v_j), we can find a u_k^*

$$u_k^* = \text{argmax}_{u_k}(P(v_i|u_k)P(v_j|u_k) - P(v_i, v_j|u_k)) \quad (3)$$

subject to

$$u_k^* \geq th$$

where th is a positive threshold on the objective function to supress very small values. If no such u_k^* exists, then (v_i, v_j) cannot be subsumed under any parent topic. A positive threshold ensures that the pair of sub-topics respects the two conditions above. A slightly different way to look at this objective function is to note that we want

$$\begin{aligned} & P(v_j|v_i, u_k) \leq P(v_j|u_k) \\ \Rightarrow \quad & P(v_i|u_k)P(v_j|v_i, u_k) \leq P(v_i|u_k)P(v_j|u_k) \\ \Rightarrow \quad & P(v_i, v_j|u_k) \leq P(v_i|u_k)P(v_j|u_k) \end{aligned}$$

And the more it is less the better.

Solving the optimization problem in Eq.(3) gives the best parent topic under which a pair of child topics *may* be subsumed. We emphasize that none of these child topics might actually be subsumed under this parent in the final hierarchy. This is just the best parent that could subsume this pair.

Next, we construct graphs G_k for each parent topic u_k. G_k consists of those sub-topics which occur as a part of a pair which is best subsumed under u_k. i.e.,

$$\begin{aligned} V(G_k) &= \{i|\exists j \text{st} u_k \text{ subsumes } (v_i, v_j)\} & (4) \\ E(G_k) &= \{(i,j)| \text{ st } u_k \text{ subsumes } (v_i, v_j)\} & (5) \\ wt(i,j) &= P(v_i|u_k)P(v_j|u_k) - P(v_i, v_j|u_k) & (6) \end{aligned}$$

Figure 2 shows some examples of such graphs for the Reuters dataset. The three graphs correspond to the three parent topics. The child layer consists of 8 topics.

Note that any clique C in this graph represents a set of topics which are mutually negatively correlated in the sense of Eq.(3). We would want that only the largest such set be subsumed under the corresponding parent. For example in the first graph in Figure 2, topics 8 and 5 are connected by an edge meaning that they are negatively correlated. Also topics 8 and 4 are connected. However, topics 4 and 5 are not connected. This would mean that they are not sufficiently negatively correlated to be considered separate subtopics of the parent topic. Hence we would not want both 4 and 5 to be subsumed. The cliques consisting of topics (8,1,4) or (8,2,4) are better choices since each topic is then sufficiently different from the others. Topic 5 would, in essence, be captured by topic 4. The weights associated with the edges denote the strength of the corresponding negative correlation. All vertices in the maximum edge-weighted clique should be subsumed under the parent topic.

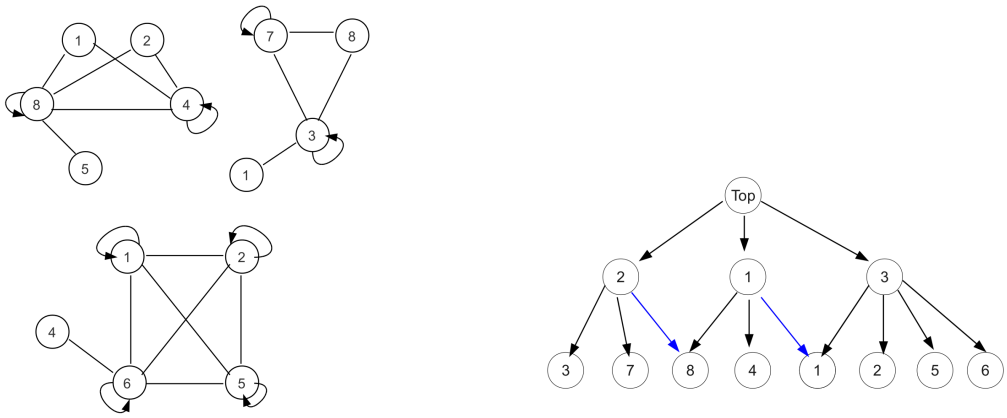
Hence the problem of determining the best subsumption set for any parent topic k reduces to finding the maximum weight clique in graph G_k. Though this problem is hard to solve in general, we observed that the graphs G_k that are induced by real datasets that we experimented on are typically very sparse (not more than 1 connected component and not more than 6-7 vertices in any graph). Hence, even an exponential time algorithm would not be too bad keeping in mind that this ontology building exercise is to be done offline. The pair-wise independence criterion used by Zavitsanos et al. [9] can be seen as a special case of this method, where only the maximum 2-clique is used for building subsumption relations. The next section explores this graph and the weight function in more detail and gives bounds on the sparsity of the induced graphs.

C. Sparsity bounds

Let there be m parent topics and n child topics in some pair of consecutive layers. An edge (i, j) can be in G_k for a unique value of k since only the best candidate parent u_k subsumes (v_i, v_j). There are $O(n^2)$ such edges distributed across m graphs. Hence, the expected number of edges in each graph is $O(n^2/m)$. Since m and n are the number of topics in adjacent layers, we can assume that $m = O(n)$. Therefore, the expected number of edges is $O(n)$. This means that the maximum-clique can be of size $O(\sqrt{n})$. Consider the function

$$f_k(i, j) = P(i|k)P(j|k) - P(i, j|k)$$

Where i and j refer to child topics and k refers to the parent. If the occurence of topics i and j is independent given k, $P(i, j|k) = P(i|k)P(j|k)$ and hence $f_k(i, j) = 0$. If the occurence of the topics is negatively correlated $f_k(i, j) > 0$.



(a) Graphs G_k for Reuters-25 topics

(b) Top few layers of the ontology

Figure 2. Determining subsumption relations

Also,

$$\begin{aligned}
\sum_{i,j} f_k(i,j) &= \sum_{i,j} (P(i|k)P(j|k) - P(i,j|k)) \\
&= \sum_{i,j} P(i|k)P(j|k) - \sum_{i,j} P(i,j|k) \\
&= \sum_{i} P(i|k) \left(\sum_{j} P(j|k) \right) - 1 \\
&= \sum_{i} P(i|k)1 - 1 \\
&= 1 - 1 = 0
\end{aligned}$$

D. Model Trimming

So far we have built the topic structure assuming that the number of topics chosen in the clustering step were correct. Clustering methods are however susceptible to errors and use a very primitive notion of similarity. The objective function 3 can be put to further use for trimming the topic structure. This time, we can look at it as follows

$$w_{i,j}(k) = P(v_i|u_k)P(v_j|u_k) - P(v_i, v_j|u_k)$$

Recall that a high value of $w_{i,j}(k)$ would mean that topics v_i and v_j are separated enough from each other and yet associated sufficiently to u_k to be considered its sub-topics. A low value, on the other hand, would mean that the topics are quite similar and as far as u_k is considered, they might be merged. This can be seen as a scheme where each parent topic u_k votes if (v_i, v_j) should be merged and the value of the vote is $w_{i,j}(k)$. A high vote value means that the corresponding parent topic wants the pair to be kept separate and a low indicates that they should be merged. These votes are then polled together to get a final value.

$$W(i,j) = \sum_{k'=1}^{k} w_{i,j}(k')$$

If $W(i,j)$ is positive, the pair is left alone, else it is merged. A possible improvement is to weigh the votes of each parent k by $P(v_i|u_k)P(v_j|u_k)$. This factor ensures that the vote of a parent topic whose occurence is more correlated with the occurence of the child topics is taken more seriously, the idea being that such a parent topic carries statistically more weight than a parent topic which has nothing to do with this pair of child topics.

$$W'(i,j) = \sum_{k'=1}^{k} P(v_i|u'_k)P(v_j|u'_k)w_{i,j}(k')$$

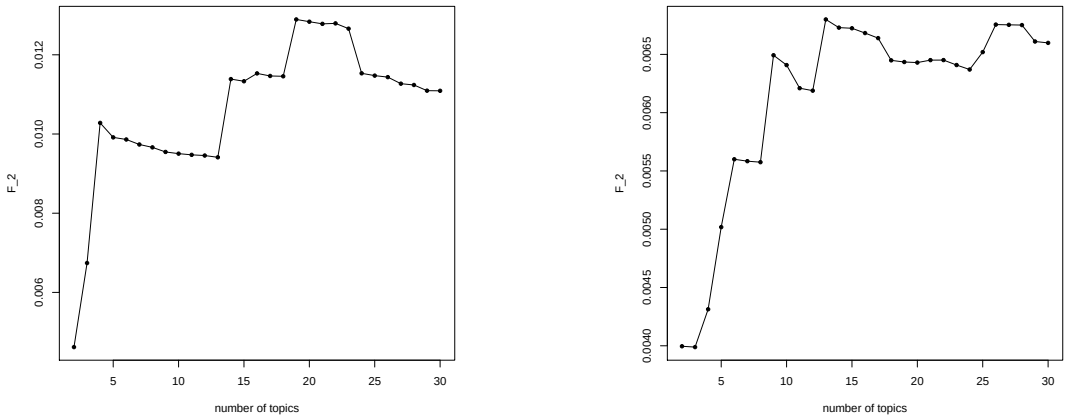
Merge (v_i, v_j) if $W'(i,j) < 0$
While the intuition behind the method seems sound, more experiments need to done to evaluate it.

IV. Evaluation

The aim of this study is to provide a method to build semantically meaningful topic structures which are similar to those built by human experts. We compare the topic structures determined using our method with human built ones.

A. Datasets

We chose a subset of the Open Directory Project (ODP) category structure and crawled a random number of webpages linked under those categories. This data was collected using the rdf dump available from the ODP homepage `http://www.dmoz.org/` accessed on 10th April 2010. Besides, we use standard datasets such as the NIPS abstracts dataset, Reuters-21578 and the 20 newsgroups dataset.



(a) Reuters-13 (b) Reuters-25

Figure 3. Plot of Criterion function vs Number of topics for different subsets of Reuters-21578 dataset

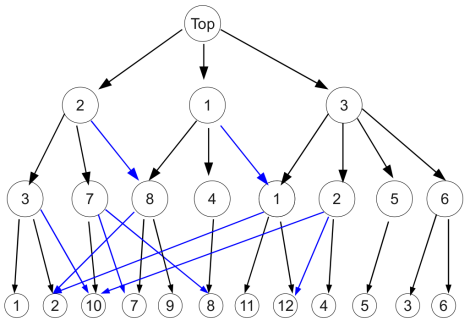


Figure 4. Top 3 layers of onotology for Reuters dataset. Black lines indicate major parent, blue lines indicate lesser degree of association.

B. *Quality of clustering*

Reuters-21578 25 topic dataset

Minima for criterion function at : 3, 8, 12, 20 and 24 topics. See Figure 3.

Top few words for a 3-topic clustering using LDA

Topic 1	ship, offici, union, strike, gulf
Topic 2	tonne, mln, wheat sugar, export, grain
Topic 3	oil, price, mln, dlr, pct, produc

Top few words for a 8-topic clustering using LDA

Topic 1	price, market, dlr, futur, exchange, trade
Topic 2	oil, mln, pct, dlr, gold
Topic 3	tonne, export, sugar, wheat, mln
Topic 4	ship, strike, compani, port, union
Topic 5	oil, opec, price, mln, bpd, saudi
Topic 6	coffee, produc, export, quota, stock, cocoa
Topic 7	mln, crop, tonne, pct, product, grain, plant
Topic 8	propos, offici, farm, wheat, agricultur, grain

Figure 2 shows the graphs generated for deciding subsumption between 8-topic layer and 3-topic layer. Figure 4 shows the first 3 layers for the generated ontology tree. The subsumption seems to be quite appropriate considering the topic keywords given above.

ODP/Comp

This data was collected by us using the rdf dump available from the Open Directory Project page accessed on 10th April 2010. A subset of the science directory urls were crawled and converted to bag of words.

Topic Structure:

- Algorithms
- Artificial Intelligence : Academic Departments, People, Conferences and Events, Machine Leanring , Natural Langugae, Neural Networks, Vision.
- Hardware : Buses, Cables, Peripherals (Audio, Keyboards, Displays, Printers)
- OpenSource
- Systems
- Internet :Organizations, Searching, Web design and development

- Security : Firewalls, Intrusion Detection Systems, Malicious software

Figure 1(c) shows the plot of the cluster quality criterion function. A significant amount of data cleaning needs to be done before this set is usable for further experiments.

This presents a very qualitative analysis of the method. A more quantitative comparison needs to be done to fully validate it. We are currently working on implementing other methods as such Non-Parametric LDA and Non-Parametric PAM which also try to estimate the size and structure of a topic hierarchy. We plan to compare our method against these on metrics such as perplexity and deviation from human defined structure.

V. Conclusions and Future Work

Our method estimates the size and structure of the topic space underlying a set of documents. However, an important step in using this method for a large number of practical applications is to develop a method for category naming that will be used to label each of the discovered categories. Our method currently lacks a formal justification and a quantitative analysis. We plan to work further in these directions.

References

[1] Y. W. Teh, M. I. Jordan, M. J. Beal, and D. M. Blei, "Hierarchical Dirichlet processes," *Journal of the American Statistical Association*, vol. 101, no. 476, pp. 1566–1581, 2006.

[2] W. Li, D. Blei, and A. Mccallum, "Nonparametric bayes pachinko allocation," in *UAI 07*, 2007. [Online]. Available: http://www.cs.umass.edu/˜mccallum/papers/npbpam-uai2007s.pdf

[3] D. Blei, T. L. Griffiths, M. I. Jordan, and J. B. Tenenbaum, "Hierarchical topic models and the nested chinese restaurant process," *Advances in Neural Information Processing Systems 16.*, 2004. [Online]. Available: http://cog.brown.edu/ gruffydd/papers/ncrp.pdf

[4] D. M. Blei and J. D. Lafferty, "Correlated topic models," in *NIPS*, 2005.

[5] W. Li and A. McCallum, "Pachinko allocation: Dag-structured mixture models of topic correlations," in *ICML '06: Proceedings of the 23rd international conference on Machine learning*. New York, NY, USA: ACM, 2006, pp. 577–584.

[6] Y. Zhao and G. Karypis, "Empirical and theoretical comparisons of selected criterion functions for document clustering," *Mach. Learn.*, vol. 55, no. 3, pp. 311–331, 2004.

[7] M. Steinbach, G. Karypis, and V. Kumar, "A comparison of document clustering techniques," in *KDD Workshop on Text Mining*, 2000.

[8] D. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *Journal of Machine Learning Research*, vol. 3, p. 2003, 2001.

[9] E. Zavitsanos, S. Petridis, G. Paliouras, and G. A. Vouros, "Determining automatically the size of learned ontologies," in *ECAI*, ser. Frontiers in Artificial Intelligence and Applications, M. Ghallab, C. D. Spyropoulos, N. Fakotakis, and N. M. Avouris, Eds., vol. 178. IOS Press, 2008, pp. 775–776.