Learning Size and Structure of Document Ontologies using Generative Topic Models

Nitish Srivastava
Advisor:Dr Harish Karnick

April 26, 2010

Abstract

We look into the problem of learning the size and structure of an interlinked hierarchy of topics which best describes a given document corpus. With increasing amount of text content being generated today, it has become important to develop techniques for learning ontologies which categorize documents into semantically meaningful classes. The ontology is best represented as a DAG to capture correlations and links between topics. We propose a method of estimating the number of topics in each layer of the topic hierarchy using agglomerative clustering. We use generative topic models (such as Latent Dirichlet Allocation [2]) to discover topics and link them to form an ontology. The links are made by finding subsumption relations between topics in consecutive layers. Our method uses a variant of the algorithm proposed in [10]. We validate our method on real-world document corpora and present the results.

1 Introduction

Document clustering has received widespread attention as a challenging problem for machine learning. Both generative and discriminative approaches to the problem have been studied. Document clustering has become an important area of interest due to its extensive use in managing real world documents, building a semantic web, improving search time and quality. The problem that we take up here is to learn an ontology that categorize a set of documents into topics most appropriately. A desirable feature of such an ontology is that it should not only categorize documents into a hierarchy of topics, but also discover associations and relations between topics across the hierarchy. Building such an ontology can be considered a problem composed of two tasks. One task is to estimate the number of topics and subtopics in the hierarchy at each level. The other, is given the number of topics, categorize the documents into these topics and discover associations between them. We focus our attention on estimating the number of topics and then using probabilistic generative topic models to categorize them. We split our task into three parts. First we estimate the number of topics in each layer of the hierarchy. Then we use this information to categorize documents using topic models. Finally, we discover subsumption relations between topics in adjacent layers of the hierarchy which allows us to find associations and correlations between different subtopics.

2 Related Work

Several approaches using generative probabilistic models have been proposed that aim to build ontologies , for example Latent Dirichlet Allocation (LDA) [2], hierarchical LDA (hLDA) [1], correlated topics model (CTM) [4] and pachinko allocation model (PAM) [7]. All these methods assume some parametrized prior distribution on the topics (such as a Dirichlet distribution in LDA, or a logistic normal in CTM) and estimate their parameters. This estimation is based on the documents and the number of topics and layers that are assumed to be known (or estimated otherwise). Efficient inference and estimation methods have been proposed including variational ([3]) and Markov Chain Monte Carlo (MCMC) methods ([8]). Heinrich et al.([5], [6]) proposed a generic view of topic models and gave a generic inference method. Given that such models have been shown to give good performnace on real datasets, we focus our attention on the problem of discovering the right number of topics in each layer. We then propose a method to discover subsumption relations. Our approach uses the posterior variational dirichlet components for each documents infered using LDA and converts the problem to finding a maximum weight clique on a very sparse graph. Zavitsanos et al. [10] approach the same problem using infered dirichlet components but use a local independence criterion to decide subsumption. Our method generalizes their approach.

3 Generative topic models

The key idea here is to establish a probabilistic procedure for sampling random documents, such that the drawn sample can model the properties of real world documents. To set up an analogy to simpler domain, consider the problem of finding a probabilistic procedure for sampling points in $\mathbb{R}^2$. One solution to this problem is to define a mixture of gaussians model for this domain. It has been shown that any distribution in $\mathbb{R}^2$ can be modelled arbitrarily closely using this procedure. Coming back to the domain of documents, we find that such simple processes cannot model the attributes of real-world documents, i.e., it would not only be computationally infeasible but also wrong to assume that documents can be generated by randomly sampling from a mixture of high dimensional gaussians. Thus we need to look for a distribution which is capable of modeling desirable properties such as the tendency of documents to be composed of "topics" and of these topics to be inter-related. At the same time, the number of parameters to be estimated must be managable.

First we need to settle what we mean by a "topic". In the discussion that follows, a topic is a multinomial distribution over words in the dictionary. Suppose we want to build a document about topic "Computers", we would choose words randomly from the multinomial distribution corresponding to this topic. Hence, for each word in the dictionary, a topic defines the probability of that word being chosen next when generating a document of that topic. In this case words "algorithm", "software", "memory" etc would have a high probability of occurence. These words would have a low probability when talking about topic "Animals". So if a document contains such words, then it is probably about the topic "Computers" and not about "Animals". In this sense, this distribution captures the topic element of the document. Note however, that the words are

sampled i.i.d. from this distribution. Thus, the document is essentially a bag of words with no syntactic structure. This is both good and bad for the model. Good as it simplifies matters a lot (makes inference and estimation possible) and bad because it misses out on the information contained in the relative closeness of different words in the document.

3.1 Latent Dirichlet Allocation

In this model each document is a mixture of various topics. A topic is characterized by a distribution over words. The key idea is that the topics themselves have a probability distribution over them, i.e., the model inforces a distribution from which the topics are sampled. Words are then sampled later from the chosen topic. We now decribe the model by building it up step by step.

A topic distribution is multinomial, with k choices, each representing a topic. The probabilities for each of these choices constitute a parameter vector for the multinomial distribution denoted by θ. So the multinomial is denoted as multinomial(θ). Since we want to allow each document to have multiple topics, the document must be composed from a mixture of various topics. If θ was held constant throughout the process, then for each sampled document, the contribution of topic k would be proportional to θ_k, in expectation. Hence each document would be expected to have the same amount of contribution from different topics. This creates a problem, if the generative model is to be general enough. Each document must be allowed to have different amounts of contributions. For example documents containing topics "Computers", "Internet" and "Laptops" together will be common. Hence we want the parameter θ to contain high proportions for these topics together. Also documents containing topic "Animals", "Pets" and "Birds" will be common. So θ should have high proportions for these also. But the number of documents containing both topic "Computers" and topic "Animals" will be rare. But in the parameter θ, these two topics have high proportion. We want that θ should adapt itself according to the topics that are present in it, otherwise the model will not be powerful enough to reject topic "Animals" when topic "Computers" has already been chosen once from the multinomial distribution. This will lead to topics "Computers" and "Animals" to be given a high chance of occuring together. Essentially there will be no way of discriminating between topics "Internet" and "Animals", since both will have high proportions. This makes it difficult to generate meaningful documents.

Probabilistic Latent Semantic Analysis (PLSA) overcomes this problem by learning a new θ for each document. This clearly has its limitations. The number of parameters to be estimated is linear in the size of the learning set. Also the generalization to unseen documents relies to smoothing and other such techniques.

LDA overcomes this problem by defining another distribution, this time on the parameter θ. So, now we have in all a 3-level bayesian system. This distribution is a Dirichlet distribution, which is parametrized by a k-dimensional vector α. The distribution is denoted as Dir(α). It has the property that a k-dimensional vector sampled from it is clustered in some sense. The clustering is the sense of a Chinese Restaurant Processs (Blei et al.([1]), i.e., the number of clusters could be potentially infinite, but the chance of a new document starting a new cluster decreases as the total number of documents increase. This trans-

lates to the property that the model represents topics as an infinte mixture and converges as more training samples are seen. So now, the vector θ will not be constant any more and before generating each document, θ will be sampled from Dir(α). This ensures that θ has certain desirable properties. It will be clustered so that similar topics can have high proportions together. Hence when a θ is sampled which has a high proportion for "Computers", the model when trained properly (i.e., for the right value of α), will allow "Laptops" and "Internet" to have high proprtion but not "Animals" , "Pets" or "Birds". Conversely, it will also suppress the proportion of "Computers" when the proportion for "Animals" is high. Hence the topic "Computers" has been represented as an infinite dimensional vector whose components are closer to the vector for "Internet" than for "Animals". This gives the model power to distinguish between similar topics and avoids the problem of PLSA since the quantity to be estimated is just the vector α which is a constant.

Generating a random N word document:

1. θ =Dir(α)
2. For i in 1 to N
3. (a) z_i =Multinomial(θ)
   (b) Choose w_i according to $p(w_i|z_i, \beta)$.

Estimation for LDA has been widely studied and several methods including variational ([3]) and Markov Chain Monte Carlo (MCMC) methods ([8]) have been used. In our experiments we use the variational inference by Blei et al. [2]. This method also gives a posterior dirichlet component vector for each document, i.e., a vector that contains the proportions of each topic in that document. It also learns the multinomial distribution over words for each topic.

The bottom line is that we now have a method which takes the document set and the the number of topics k as input and gives us the multinomial distribution for each topic and also a vector for each document in the learning set which tells us how much each topic has contributed to it.

4 Estimating Structure of the Ontology

Generative topic models require the number of topics to be specified. In order to estimate them using the corpus we use a bottom-up clustering approach, i.e., cluster the documents in k-sized clusters for $k \in \{2, 3, \ldots, K\}$ and then evaluate the quality of the clustering to find which ones are best. Though this clustering is discriminative and forms mutually exclusive clusters, we expect that the best values of k would not be too far from the optimal numbers. Note that the topic models discussed in the previous section do not yield mutually exclusive topics and that allows us to capture association between topics. This clustering is based on a mutually exclusive assumption but we are not using the clusters themselves any further in the process of building the ontology. We just use the number of clusters. Our experiments show that this number is indeed quite close to the real number of topics. See Figure 1 for examples.

4.1 Agglomerative Clustering

In order to cluster the documents of the corpus, each document is represented as a vector of normalized term frequencies. The similarity metric used in the cosine measure. So if $\overline{d_i}$ and $\overline{d_2}$ are two documents, then their similarity is

$$s(\overline{d_1}, \overline{d_2}) = \overline{d_1}.\overline{d_2}$$

Agglomerative clustering initially assigns each document to its own cluster. Then pairs of clusters are repeatedly merged until a certain stopping criterion is met. For determining the next pair of clusters to be merged UPGMA criterion was used. This algorithm was used to produce k-clusters for different values of k.

4.2 Measuring quality of clusters

We use the following criterion to measure quality of a k-clustering of the corpus

$$F_(\overline{\mu_i}, \overline{\mu_e}, \overline{\sigma_i}) = \frac{1}{\sum_{k=1}^{K} |C_i|} \sum_{k=1}^{K} |C_i| \frac{\mu_e^k \sigma_i^k}{\mu_i^k} \tag{1}$$

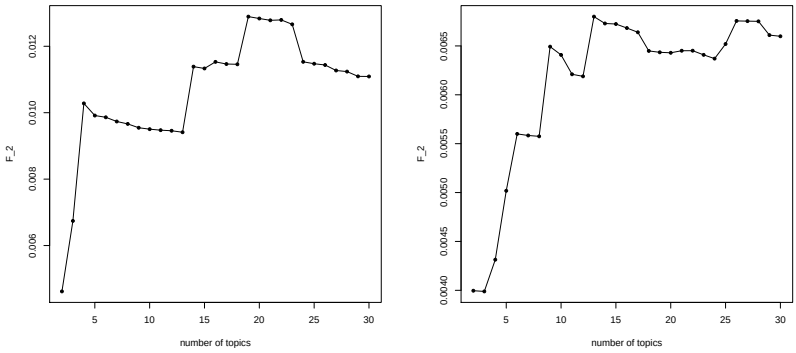
Here
$\overline{\mu_i}$ denotes the vector of intra-cluster similarity.
$\overline{\mu_e}$ denotes the vector of inter-cluster similarity.
$\overline{\sigma_i}$ denotes the vector standard deviation in intra-cluster similarity.
The idea is to have low similarity with documents outside the cluster and high similarity, with low standard deviation inside. The ratio is weighed by the size of the cluster. This function is computed for each k-clustering. This givea a sequence of quality values corresponding to the sequence of topics. A low value implies a good clustering. Hence ,the local minima of the sequence represent the regions where the number of topics are such that the clustering produced is locally optimal. These values are chosen as the number of topics in successive layers.

5 Discovering subsumption relations

The next step in finding the ontology is to build its link structure. We propose a method using the posterior variational dirichlet components for each document.

5.1 Document as a finite mixture over topics

In LDA, each document is considered a finite mixture of topics. The variational inference algorithm used for categorizing the documents gives the posterior dirichlet components for each training document, i.e. for each document d, the topic model gives a vector $p_d = (p_1, p_2, \ldots, p_k)$. Each p_i represents the proportion of topic i in the document. Thus, the component p_i can be interpreted as the probability of topic i occuring in the document. This definition of probability of occurence of a topic ina document allows us to then define probability of ocuurence of a topic in the corpus, or the joint probablity of occurence of topics etc. These probabilities are used in the next step for discovering subsumption relations.



(a) Reuters 13 topic dataset (b) Reuters 25 topic dataset

Figure 1: Measuring cluster quality for differnt number of topics

5.2 Criterion for subsumption

Let A and B be any two topics in the child layer and C be a topic in the parent layer. We would like C to subsume A and B if they could be considered sub-topics of C. For them to be subtopics, two conditions must hold. A and B should be associated with C but at the same time they must be sufficiently separate so that they can be considered separate subtopics. Thus if $P(X)$ denotes the probability of occurence of topic X in the corpus, these conditions can be intrepreted as demanding that $P(A|C)$ and $P(B|C)$ should both be high (among all possible C's) but $P(A, B|C)$ must be low, i.e. the joint probability that A and B occur together given C must be low. Then C is a good candidate for subsuming A and B. In other words, the occurence of A and B must be negatively correlated given C. Hence for every (A, B), we can find a C^*

$$C^* = \text{argmax}_C(P(A|C)P(B|C) - P(A, B|C)) \tag{2}$$

subject to

$$C^* \geq th$$

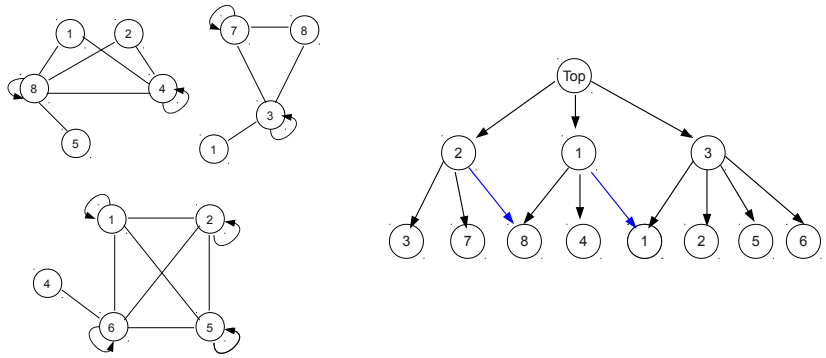
where th is a threshold on the criterion function to supress very small values. If no such C^* exists, then (A, B) cannot be subsumed under any parent topic. Another way of getting at this objective function is to note that we wnat

$$\begin{aligned} & P(B|A, C) \leq P(B|C) \\ \Rightarrow \quad & P(A|C)P(B|A, C) \leq P(A|C)P(B|C) \\ \Rightarrow \quad & P(A, B|C) \leq P(A|C)P(B|C) \end{aligned}$$

And the more its is less the better.

5.3 Conversion to a max-weight clique problem

The above method gives for every (v_i, v_j) in the child layer, a parent node u_k which best subsumes them, if there is any. The value of the objective function



(a) Graphs G_k for Reuters 25 topics (b) Top few layers of the ontology

Figure 2: Determining the subsumption set

at C^* gives a score for this relation. Using this we construct a weighted graph G_k for every parent node k.

$$V(G_k) = \cup\{v_i, v_j\} \text{ such that } u_k \text{ subsumes } (v_i, v_j) \quad (3)$$

$$E(G_k) = \cup\{(v_i, v_j)\} \text{ such that } u_k \text{ subsumes } (v_i, v_j) \quad (4)$$

$$wt(v_i, v_j) = P(u_k|v_i)P(u_k, v_j) - P(v_i, v_j|u_k) \quad (5)$$

Figure 2(a) shows some examples of such graphs for the Reuters-21578 dataset. The three graphs correspond to the three parent topics. The child layer consists for 8 topics.

Note that any clique C in this graph represnts a set of topics which are mutually negatively correlated in the sense of Eq.(2). We would want that only the largest such set be subsumed under the corresponding parent. For example in the first graph in Figure 2(a), 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. The weights associated with the edges denote the strength of the corresponding negative correlation.

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 NP-Complete 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.

So, we find the maximum weight clique C in graph G_k.
If $wt(C) \geq th$, an edge (u_k, v_i) is added to the ontology for every $v_i \in C$.

This process is repeated layer after layer, till there are no more subsumption relations to be found. The threshold th suppreses cliques of very small weight.

Previous work in subsumption by Zavitsanos et al. [10] uses an independence criterion.
For every u_k in the parent layer, they choose the pair (v_i, v_j) in the child layer which makes v_i and v_j independent given u_k. This approach becomes a special case of our method that of finding the maximum weight 2-clique. Our method

uses a different criterion and a general clique framework.

6 Results

6.1 Reuters-21578 25 topic dataset

Minima for criterion function at : 3, 8, 12, 20 and 24 topics. See Figure 1

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 3 shows the first 3 layers for the generated ontology tree. The subsumption seems to be quite appropriate considering the topic keywords given above.

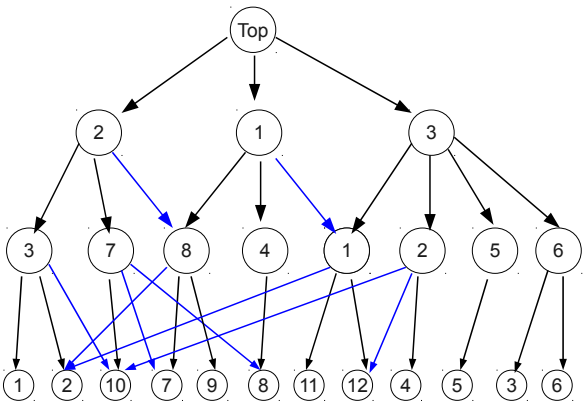


Figure 3: Ontology for Reuters 25 topic dataset

6.2 Science Directory of the Open Directory Project

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:

- Astronomy : Astronomers, Cosmology, Galaxies, Stars
- Biology : Biochemistry, Botany, Genetics
- Physics : Astrophysics, Condensed Matter, Electromagnetism, Relativity, Quantum Mechanics
- Chemistry : Computational, Organic, Physical

Figure 4 shows the plot of the cluster quality criterion function. The local minima can be used to estimate number of topics for this set. The estimated number seems quite close to the actual.

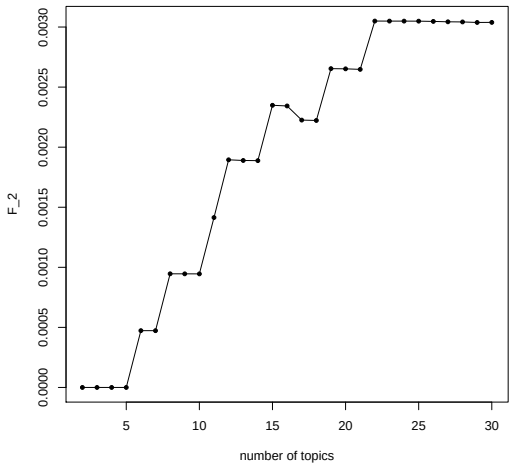


Figure 4: Finding number of topics for ODP Science Data

7 Conclusion

We conclude that the 3 step process for learning an ontology that we describe seems to be a reasonable way of doing the task. It performs well on standard datasets and stands on a good theoretical ground which is intuitively motivated.

8 Acknowledgments

We used the LDA-C code by Blei [2]. Agglomerative clustering was done using CLUTO, a clustring toolkit [9]. We thank them for their help.

References

[1] 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.

[2] David Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:2003, 2001.

[3] David M. Blei and Michael I. Jordan. Variational methods for the dirichlet process. In Carla E. Brodley, editor, *ICML*, volume 69 of *ACM International Conference Proceeding Series*. ACM, 2004.

[4] David M. Blei and John D. Lafferty. Correlated topic models. In *NIPS*, 2005.

[5] Gregor Heinrich. A generic approach to topic models. In *In Proc. European Conf. on Mach. Learn. / Principles and Pract. of Know. Discov. in Databases (ECML/PKDD*, 2009.

[6] Gregor Heinrich and Michael Goesele. Variational bayes for generic topic models. In Bärbel Mertsching, Marcus Hund, and Muhammad Zaheer Aziz, editors, *KI*, volume 5803 of *Lecture Notes in Computer Science*, pages 161–168. Springer, 2009.

[7] Wei Li and Andrew McCallum. Pachinko allocation: Dag-structured mixture models of topic correlations. In *ICML '06: Proceedings of the 23rd international conference on Machine learning*, pages 577–584, New York, NY, USA, 2006. ACM.

[8] Ian Porteous, David Newman, Alexander Ihler, Arthur Asuncion, Padhraic Smyth, and Max Welling. Fast collapsed gibbs sampling for latent dirichlet allocation. In *KDD '08: Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 569–577, New York, NY, USA, 2008. ACM.

[9] Michael Steinbach, George Karypis, and Vipin Kumar. A comparison of document clustering techniques. In *KDD Workshop on Text Mining*, 2000.

[10] Elias Zavitsanos, Sergios Petridis, Georgios Paliouras, and George A. Vouros. Determining automatically the size of learned ontologies. In Malik Ghallab, Constantine D. Spyropoulos, Nikos Fakotakis, and Nikolaos M. Avouris, editors, *ECAI*, volume 178 of *Frontiers in Artificial Intelligence and Applications*, pages 775–776. IOS Press, 2008.