The Cluster-Abstraction Model: Unsupervised Learning of Topic Hierarchies from Text Data

Thomas Hofmann
Computer Science Division, UC Berkeley &;
International CS Institute, Berkeley, CA
hofmann@cs.berkeley.edu

Abstract

This paper presents a novel statistical latent class model for text mining and interactive information access. The described learning architecture, called *Cluster-Abstraction Model* (CAM), is purely data driven and utilizes contact-specific word occurrence statistics. In an intertwined fashion, the CAM extracts hierarchical relations between groups of documents as well as an abstractive organization of keywords. An annealed version of the Expectation-Maximization (EM) algorithm for maximum likelihood estimation of the model parameters is derived. The benefits of the CAM for interactive retrieval and automated cluster summarization are investigated experimentally.

1 Introduction

Intelligent processing of text and documents ultimately has to be considered as a problem of natural language understanding. This paper presents a statistical approach to learning of language models for context-dependent word occurrences and discusses the applicability of this model for interactive information access. The proposed technique is purely data-driven and does not make use of domain-dependent background information, nor does it rely on predefined document categories or a given list of topics.

The *Cluster-Abstraction Model* (CAM), is a statistical latent class or mixture model [McLachlan and Basford, 1988] which organizes groups of documents in a hierarchy. Compared to most state-of-the-art techniques based on agglomerative clustering (e.g., [Jardine and van Rijsbergen, 1971; Croft, 1977; Willett, 1988]) it has several advantages and additional features: As a *probabilistic model* the most important advantages are:

- a sound foundation in statistics and probabilistic inference
- a principled evaluation of generalization performance for model selection,
- efficient model fitting by the EM algorithm,
- an explicit representation of conditional independence relations.

Additional advantages are provided by the hierarchical nature of the model, namely:

- multiple levels of document clustering,
- discriminative topic descriptors for document groups,
- coarse-to-fine approach by annealing.

The following section will first introduce a non-hierarchical probabilistic clustering model for documents, which is then extended to the full hierarchical model.

2 Probabilistic Clustering of Documents

Let us emphasize the clustering aspect by first introducing a simplified, non-hierarchical version of the CAM which performs 'flat' probabilistic clustering and is closely related to the *distributional clustering* model [Pereira *et al.*, 1993] that has been used for word clustering and text categorization [Baker and McCallum, 1998]. Let $d \in \mathcal{D} = \{d^{(1)}, \ldots, d^{(I)}\}$ denote documents and $w \in \mathcal{W} = \{w^{(1)}, \ldots, w^{(J)}\}$ denote words or word stems. Moreover let w_d refer to the vector (sequence) of words W_{dt} constituting d. Word frequencies are summarized using count variables $n(d,w)$ which indicate how often a word w occurred in a document d; $n(d) = \sum_w n(d,w)$ denotes the document length.

Following the standard latent class approach, it is assumed that each document d belongs to exactly one cluster $c_d \in \mathcal{C} = \{c^{(1)}, \ldots, c^{(K)}\}$, where the number of clusters is assumed to be fixed for now. Introducing class conditional word distributions $P(w|c)$ and class prior probabilities $P(c)$ (stacked in a parameter vector θ), the model is defined by $P(c_d = c; \theta) = P(c)$ and

$$P(w_d|c_d = c; \theta) = \prod_{t=1}^{n(d)} P(w_{dt}|c) = \prod_w P(w|c)^{n(d,w)}. \quad (1)$$

The factorial expression reflects conditional independence assumptions about word occurrences in w_d (bag-of-words model).

Starting from (1) the standard EM approach [Dempster *et ai,* 1977] to latent variable models is employed. In EM two re-estimation steps are alternated:

- an Expectation (E)-step for estimating the posterior probabilities of the unobserved clustering variables $P(c_d = c|\mathbf{w}_d; \theta')$ for a given parameter estimate θ',
- a Maximization (M)-step, which involves maximization of the so-called *expected complete data log-likelihood* for given posterior probabilities with respect to the parameters. \

The EM algorithm is known to increase the observed likelihood in each step, and converges to a (local) maximum under mild assumptions.

An application of Bayes' rule to (1) yields the following E-step re-estimation equations for the distributional clustering model

$$P(c_d = c|\mathbf{w}_d; \theta) = \frac{P(c)\prod_w P(w|c)^{n(d,w)}}{\sum_{c'} P(c')\prod_w P(w|c')^{n(d,w)}}. \qquad (2)$$

The M-step stationary equations obtained by differentiating *C* are given by

$$P(c) = \frac{1}{I}\sum_d P(c_d = c|\mathbf{w}_d; \theta) \qquad (3)$$

$$P(w|c) = \frac{\sum_d P(c_d = c|\mathbf{w}_d; \theta) n(d, w)}{\sum_d P(c_d = c|\mathbf{w}_d; \theta) n(d)}. \qquad (4)$$

These equations are very intuitive: The posteriors $P(c_d = c|\mathbf{w}_d; \theta)$ encode a probabilistic clustering of documents, while the conditionals $P(w|c)$ represent *average* word distributions for documents belonging to group c. Of course, the simplified flat clustering model defined by (1) has several deficits. Most severe are the lack of a multi-resolution structure and the inadequacy of the 'prototypical' distributions *P(w\c)* to emphasize discriminative or characteristic words (they are in fact typically dominated by the most frequent word occurrences). To cure these flaws is the main goal of the hierarchical extension.

3 Document Hierarchies and Abstraction

3.1 The Cluster-Abstraction Model

Most hierarchical document clustering techniques utilize agglomerative algorithms which generate a cluster hierarchy or dendogram as a by-product of successive cluster merging (cf. [Willett, 1988]). In the CAM we will use an *explicit abstraction model* instead to represent hierarchical relations between document groups. This is achieved by extending the 'horizontal' mixture model of the previous section with a 'vertical' component that captures the specificity of a particular word *w* in the context of a document *d.* It is assumed that each word occurrence *Wdt* has an associated *abstraction node* a, the latter being identified with inner or terminal nodes of the cluster hierarchy (cf. Figure 1 (a)).

To formalize the sketched ideas, additional latent variable vectors ad with components a_{dt} are introduced which assign words in *d* to exactly one of the nodes in the hierarchy. Based on the topology of the nodes in the hierarchy the following constraints between the cluster variables c_d and the abstraction variables a_{dt} are imposed:

$$a_{dt} \in \{a|a \text{ is above } c_d \text{ in the hierarchy}\} \qquad (5)$$

The notation $a \uparrow c$ will be used as a shortcut to refer to nodes *a* above the terminal node c in the hierarchy. Eq. (5) states that the admissible values of the latent abstraction variables a_{dt} for a particular document with latent class c_d are restricted to those nodes in the hierarchy that are predecessors of c_d. This breaks the permutation-symmetry of the abstraction nodes as well as of the document clusters. An abstraction node a at a particular place in the hierarchy can only be utilized to "explain" words of documents associated with terminal nodes in the subtree of a. A pictorial representation can be found in Figure 1 (b): if *d* is assigned to c the choices for abstraction nodes for word occurrences *Wdt* are restricted to the 'active' (highlighted) vertical path. One may think of the CAM as a mixture model with a *horizontal* mixture of clusters and a *vertical* mixture of abstraction levels. Each horizontal component is a mixture of vertical components on the path to the root, vertical components being shared by different horizontal components according to the tree topology.

Generalizing the non-hierarchical model in (1), a probability distribution $P(w|a)$ over words is attached to each node (inner or terminal) of the hierarchy. After application of the chain rule, the complete data model (i.e., the joint probability of all observed and latent variables) can be specified in three steps $P(c_d = c; \theta) = P(c)$, $P(a_{dt} = a|c_d = c; \theta) = P(a|c, d)$, and

$$P(\mathbf{w}_d|\mathbf{a}_d; \theta) = \prod_{t=1}^{n(d)} P(w_{dt}|a_{dt}). \qquad (6)$$

Note that additional document-specific vertical mixing proportions $P(a|c, d)$ over abstraction nodes above cluster c have been introduced, with the understanding that $P(a|c, d)$ = 0 whenever it is not the case that a | c. If one makes the simplifying assumption that the same mixing proportions are shared by all documents assigned to a particular cluster (i.e., $P(a|c, d) = P(a|c)$), the solution degenerates to the distributional clustering model since one may always choose $P(a|c) = \delta_{ac}$. However, we propose to use this more parsimonious model and fit P(a|c) from held-out data (a fraction of words held out from each document), which is in the spirit of model interpolation techniques [Jelinek and Mercer, 1980].

3-2 EM Algorithm

As for the distributional clustering model before, we will derive an EM algorithm for model fitting. The E-step requires to compute (joint) posterior probabilities of the form $P(c_d = c, a_{dt} = a|\mathbf{w}_d; \theta)$. After applying the chain

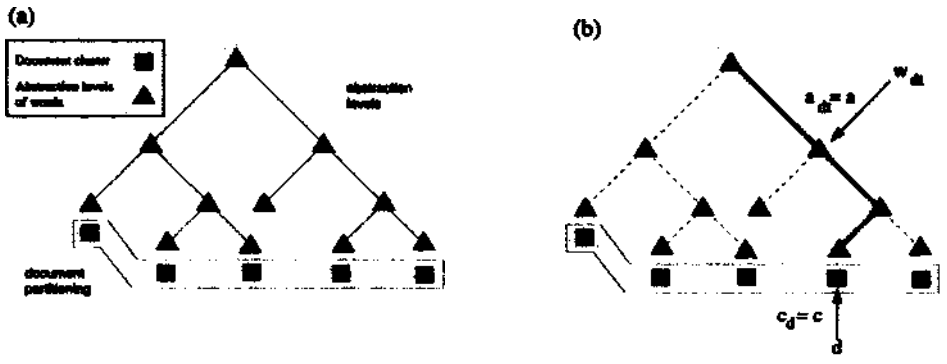

Figure 1: (a) Sketch of the cluster-abstraction structure, (b) the corresponding representation for assigning occur* rences to abstraction levels in terms of latent class variables.

rule one obtains:

$$P(c_d=c|\mathbf{w}_d;\theta) \propto P(c)\prod_w \left[\sum_a P(w|a)P(a|c)\right]^{n(d,w)}, \quad (7)$$

$$P(a_{dt}=a|\mathbf{w}_d,c_d=c;\theta) = \frac{P(w_{dt}|a)P(a|c)}{\sum_{a'} P(w_{dt}|a')P(a'|c)}. \quad (8)$$

The M-step re-estimation equations for the conditional word distributions are given by

$$P(w|a) = \frac{\sum_d \sum_{t:w_{dt}=w} P(a_{dt}=a|\mathbf{w}_d;\theta)}{\sum_d \sum_t P(a_{dt}=a|\mathbf{w}_d;\theta)}, \quad (9)$$

where $P(a_{dt}=a|\mathbf{w}_d;\theta) = \sum_c P(c_d=c|\mathbf{w}_d;\theta)P(a_{dt}=a|\mathbf{w}_d,c_d=c;\theta)$. Moreover, we have the update equation (3) for the class priors $P(c)$ and the formula

$$P(a|c) \propto \sum_d P(c_d=c|\mathbf{w}_d;\theta) \times \sum_t P(a_{dt}=a|\mathbf{w}_d,c_d=c;\theta) \quad (10)$$

which is evaluated on the held-out data. Finally, it may be worth taking a closer look at the predictive word probability distribution *P(w\d)* in the CAM which is given by

$$P(w|d) = \sum_c P(c_d=c|\mathbf{w}_d;\theta)\sum_a P(a|c)P(w|a). \quad (11)$$

If we assume for simplicity that $P(c_d=c|\mathbf{w}_d;\theta)=1$ for some *c* (hard clustering case), then the word probability of *d* is modeled as a mixture of occurrences from different abstraction levels a. This reflects the reasonable assumption that each document contains a certain mixture of words ranging from general terms of ordinary language to highly specific technical terms and specialty words.

3.3 Annealed EM Algorithm

There are three important problems which also need to be addressed in a successful application of the CAM: First and most importantly, one has to avoid the problem of *overfitting.* Second, it is necessary to specify a method to determine a meaningful tree topology including the maximum number of terminal nodes. And third, one may also want to find ways to reduce the sensitivity of the EM procedure to local maxima. An answer to all three questions is provided by a generalization called *annealed EM* [Hofmann and Puzicha, 1998]. Annealed EM is closely related to a technique known as *deterministic annealing* that has been applied to many clustering problems (e.g. [Rose *et al.,*1990; Pereira *et* al., 1993]). Since a thorough discussion of annealed EM is beyond the scope of this paper, the theoretical background is skipped and we focus on a procedural description instead. The key idea in deterministic annealing is the introduction of a temperature parameter $T \in \mathbb{R}^+$. Applying the annealing principle to the clustering variables, the posterior calculation in (7) is generalized by replacing *n(d,w)* in the exponent by $n(d,w)/T$. For $T > 1$ this dampens the likelihood contribution linearly on the log-probability scale and will in general increase the entropy of the (annealed) posterior probabilities. In annealed EM, *T* is utilized as a control parameter which is initialized at a high value and successively lowered until the performance on the held-out data starts to decrease. Annealing is advantageous for model fitting, since it offers a simple and inexpensive regularization which avoids overfitting and improves the average solution quality. Moreover, it also offers a way to generate tree topologies, since annealing leads through a sequence of so-called phase transitions, where clusters split. In our experiments, *T* has been lowered until the perplexity (i.e., the log-averaged inverse word probability) on held-out data starts to increase, which automatically defines the number of terminal nodes in the hierarchy. More details on this subject can be found in [Hofmann and Puzicha, 1998].

4 Results and Conclusion

All documents used in the experiments have been preprocessed by word suffix stripping with a word stemmer. A standard stop word list has been utilized to eliminate the most frequent words, in addition very rarely occurring words have also been eliminated. An exam-

Saul, L.; Jordan, M.I. Learning *in* Boltnnann trees.
Neural Computation, Nov. 1094, voL6, {no.6):1174~M.

(a) *Verbatim*

Introduces a large family of Boitimann machines that can be trained by standard gradient descent. The networks can have one or more layers of hidden units, with tree-like connectivity. We show how to implement a supervised learning algorithm for these Boltzmann machines exactly, without resort to simulated or mean-field annealing. The stochastic averages that yield the gradients in weight space are computed by the technique of decimation. We present results on the problems of N-bit parity and the detection of hidden symmetries.

(b) *Word stems*

introdnc larg famili boltzmann machin train standard gradient descent network layer hidden unit connect implem supervis learn algorithm boltzmann machin exactli simul anneal stochast averag yield gradient weight space techniqu present result problem pariti detec hidden symmetri

(c) *Ghost writer*

level 1	paper model base new method gener process differ effect approach provid set studi develop author
level 2	function propos model error method input optim gener neural paramet paper obtain shown appli output
level 3	gener number set neural propos function perform method inform data given obtain approxim dynamic input
level *4*	neural pattern rule number process recogni rate perform classif propos gener input neuron time properti data
level 5	converg neural optim method rule rate dynamic process pattern paramet studi statist condition adapt limit
level 6	perceptron exampl error gener rale onlin calcul deriv backpropag simpl output asymptot solution separ unsupervis
level 7	error neural architectur perform entropi statist multilay activ backpropag gener number maximum pattern phase
level 8	teacher delta output introdnc sampl replica decai nois projec correl student temperatur gain dynamic predic

Figure 2: (a) Abstract from the generated LEARN document collection, (b) representation in terms of word stems, (c) words with lowest perplexity under the CAM for words not occurring in the abstract (differentiated according to the hierarchy level).

Most frequent words					CAM node top words				
# 33	#35	#42	#50	# 88	# 33	#35	#42	#50	# 88
learn	learn	control	learn	learn	analog	program	feedback	reinforc	interact
network	exampl	learn	algorithm	educ	control	theori	desir	scheme	video
neural	algorithm	robot	network	student	oper	set	dynamic	oper	multimedia
algorithm	gener	model	neural	technologi	implement	space	position	adapt	scienc
weight	problem	propos	propos	develop	backpropag	induc	arm	parallel	remot

Figure 3: Group descriptions for exemplary inner nodes by most frequent words and by the highest probability words from the respective CAM node.

ple abstract and its index term representation is depicted in Figure 2 (a),(b). The experiments reported are some typical examples selected from a much larger number of performance evaluations. They are based on two datasets which form the core of our current prototype system: a collection of 3609 recent papers with 'learning' as a titleword, including all abstracts of papers from *Machine Learning* Vol. 10-28 (LEARN), and a dataset of 1568 recent papers with 'cluster' in the title (CLUSTER).

The first problem we consider is to estimate the probability for a word occurrence in a text based on the statistical model. Figure 2 (c) shows the most probable words from different abstraction levels, which did not occur in the original text of Figure 2 (a). The abstractive organization is very helpful to distinguish layers with trivial and unspecific word suggestions (like 'paper') up to highly specific technical terms (like 'replica').

One of the most important benefits of the CAM is the resolution-specific extraction of characteristic keywords. In Figure 4 and 6 we have visualized the top 6 levels for the dataset LEARN and CLUSTER, respectively. The overall hierarchical organization of the documents is very satisfying, the topological relations between clusters seems to capture important aspects of the inter-document similarities. In contrast to most multiresolution approaches the distributions at inner nodes of the hierarchy are not obtained by a coarsening procedure which typically performs some sort of averaging over the respective subtree of the hierarchy. The abstraction mechanism in fact leads to a specialization of the inner nodes. This specialization effect makes the probabilities *P(w\a)* suitable for *cluster summarization.* Notice, how the low-level nodes capture the specific vocabulary of the documents associated with clusters in the subtree below. The specific terms become automatically the most probable words in the component distribution, because higher level nodes account for more general terms. To stress this point we have compared the abstraction result with probability distributions obtained by averaging over the respective subtree. Figure 3 summarizes some exemplary comparisons showing that averaging mostly results in high probabilities for rather unspecific terms, while the CAM node descriptions are highly discriminative. The node-specific word distribution thus offer a principled and very satisfying solution to the problem of finding resolution-specific index terms for document groups as opposed to many circulating ad hoc heuristics to distinguish between typical and topical terms.

An example run for an interactive coarse-to-fine retrieval with the CLUSTER collection is depicted in Figure 5, where we pretend to be interested in documents on clustering for texture-based image segmentation. In a real interactive scenario, one would of course display more than just the top 5 words to describe document groups and use a more advanced shifting window approach to represent the actual focus in a large hierarchy. In addition to the description of document groups by inner node word distributions, the CAM also offers the possibility to attach prototypical documents to each of the nodes (the ones with maximal probability P(a|d)), to compute most probable documents for a given query, etc. All types of information, the cluster summaries by

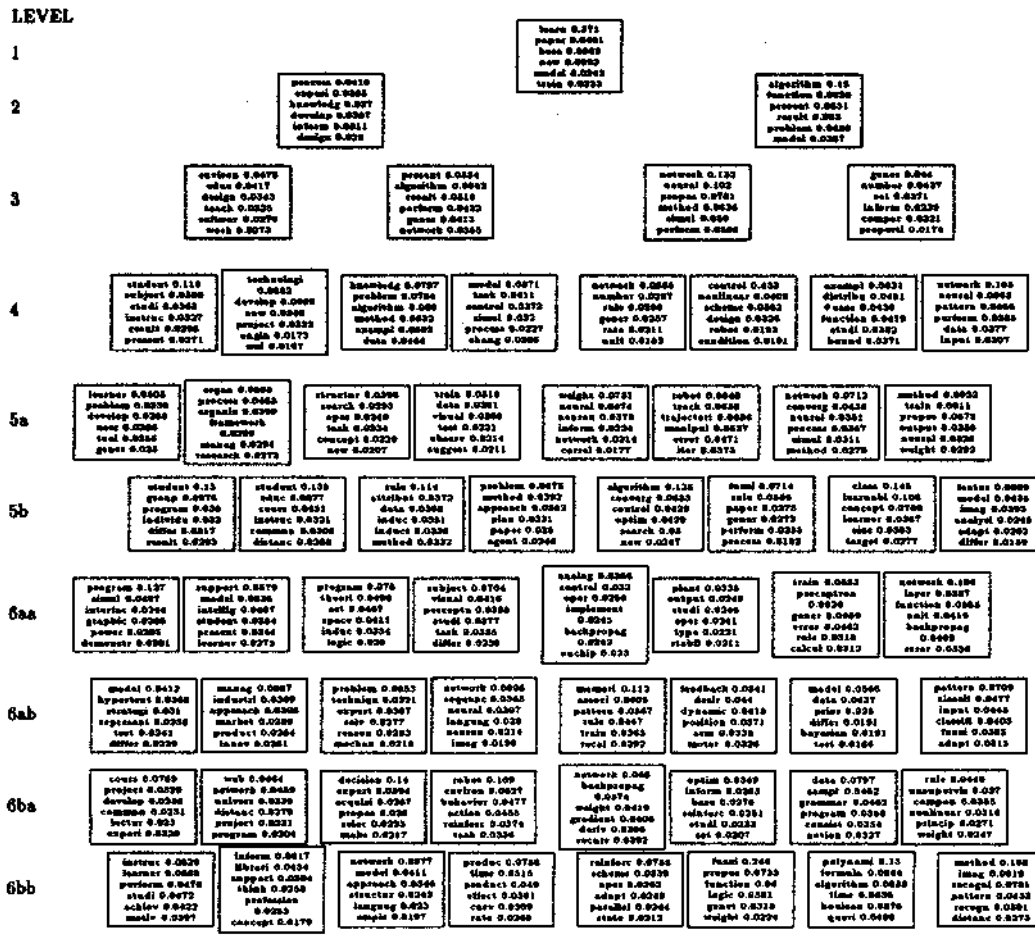

Figure 4: Top 6 levels of the cluster hierarchy for the LEARN dataset. Nodes are represented by their most probable words. Left/right successors of nodes in row 4 are depicted in row 5a and 5b, respectively. Similarly, left successors of nodes in row 5a/5b can be found in rows 6aa/6ba and right successors in rows 6ab/6bb.

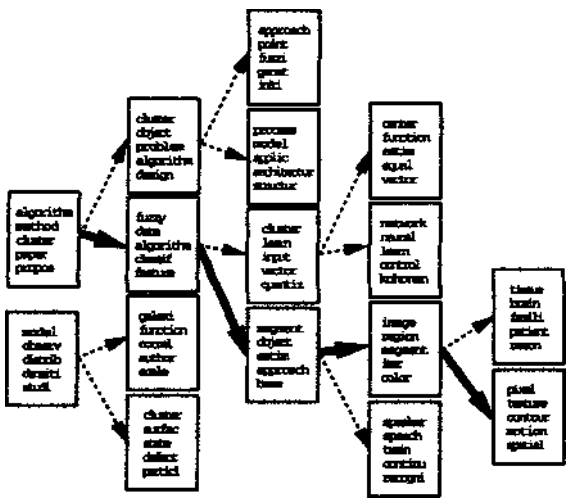

Figure 5: Example run of an interactive image retrieval for documents on 'texture-based image segmentation' with one level look-ahead in the CAM hierarchy.

(locally) discriminant keywords, the keyword distributions over nodes, and the automatic selection of prototypical documents are particularly beneficial to support an interactive retrieval process. Due to the abstraction mechanism the cluster summaries are expected to be more comprehensible than descriptions derived by simple averaging. The hierarchy offers a direct way to refine queries and can even be utilized to actively ask the user for additional specifications.

Conclusion: The cluster-abstraction model is a novel statistical approach to text mining which has a sound foundation on the likelihood principle. The dual organization of document cluster hierarchies and keyword abstractions makes it a particularly interesting model for interactive retrieval. The experiments carried out on small/medium scale document collections have emphasized some of the most important advantages. Since the model extracts hierarchical structures and supports resolution dependent cluster summarizations, the application to large scale databases seems promising.

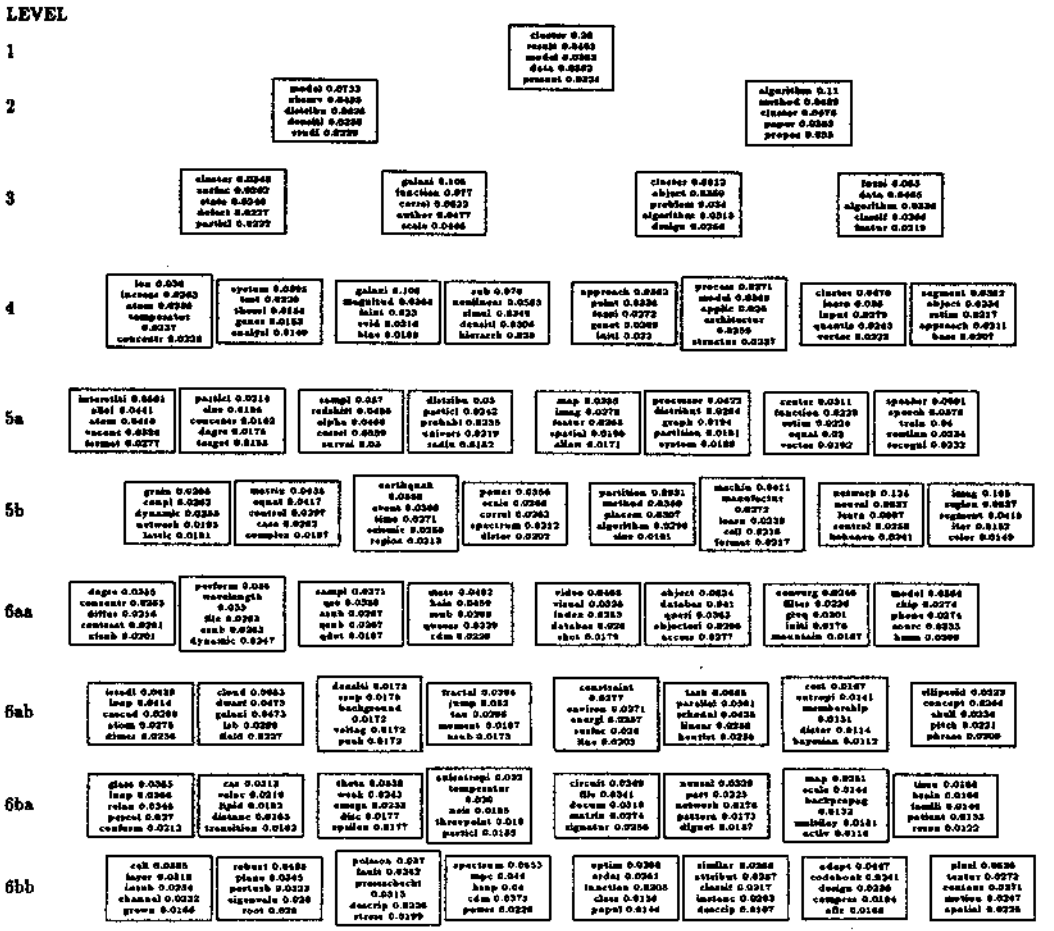

Figure 6: Top 6 levels of the cluster hierarchy for the CLUSTER dataset (cf. comments for Figure 4).

Acknowledgments: Helpful comments of Jan Puzicha, Sebastian Thrun, Andrew McCallum, Tom Mitchell, Hagit Shatkay, and the anonymous reviewers are greatly acknowledged. Thomas Hofmann has been supported by a DAAD postdoctoral fellowship.

References

[Baker and McCallum, 1998] L.D. Baker and A.K. McCallum. Distributional clustering of words for text classification. In *SIGIR,* 1998.

[Croft, 1977] W.B. Croft. Clustering large files of documents using the single-link method. *Journal of the American Society for Information Science,* 28:341-344, 1977.

[Dempster *et al,* 1977] A.P. Dempster, N.M. Laird, and D.B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society B,* 39:1-38, 1977.

[Hofmann and Puzicha, 1998] T. Hofmann and J. Puzicha. Statistical models for co-occurrence data. Technical Report 1625, Memo, AI Lab/CBCL, M.I.T., 1998.

[Jardine and van Rijsbergen, 1971] N. Jardine and C.J. van Rijsbergen. The use of hierarchical clustering in information retrieval. *Information Storage and Retrieval,* 7:217-240, 1971.

[Jelinek and Mercer, 1980] F. Jelinek and R. Mercer. Interpolated estimation of Markov source parameters from sparse data. In *Proceedings of the Workshop of Pattern Recognition in Practice,* 1980.

[McLachlan and Basford, 1988] G.J. McLachlan and K. E. Basford. *Mixture Models.* Marcel Dekker, INC, New York Basel, 1988.

[Pereira *et al,* 1993] F.C.N. Pereira, N.Z. Tishby, and L. Lee. Distributional clustering of english words. In *Proceedings of the Association for Computational Linguistics,* pages 183-190, 1993.

[Rose *et al,* 1990] K. Rose, E. Gurewitz, and G. Fox. Statistical mechanics and phase transitions in clustering. *Physical Review Letters,* 65(8):945-948,1990.

[Willett, 1988] P. Willett. Recent trends in hierarchical document clustering: a critical review. *Information Processing & Management,* 24(5):577-597, 1988.