MASSACHUSETTS INSTITUTE OF TECHNOLOGY
ARTIFICIAL INTELLIGENCE LABORATORY

A.I. Memo No. 1625
C.B.C.L. Memo No. 159

February, 1998

Statistical Models for Co-occurrence Data

Thomas Hofmann **Jan Puzicha**

This publication can be retrieved by anonymous ftp to publications.ai.mit.edu.

Abstract

Modeling and predicting *co-occurrences of events* is a fundamental problem of unsupervised learning. In this contribution we develop a statistical framework for analyzing co-occurrence data in a general setting where elementary observations are joint occurrences of pairs of abstract objects from two finite sets. The main challenge for statistical models in this context is to overcome the inherent data sparseness and to estimate the probabilities for pairs which were rarely observed or even unobserved in a given sample set. Moreover, it is often of considerable interest to extract *grouping structure* or to find a *hierarchical data organization*. A novel family of *mixture models* is proposed which explain the observed data by a finite number of shared *aspects* or *clusters*. This provides a common framework for statistical inference and structure discovery and also includes several recently proposed models as special cases. Adopting the maximum likelihood principle, *EM* algorithms are derived to fit the model parameters. We develop improved versions of EM which largely avoid overfitting problems and overcome the inherent locality of EM–based optimization. Among the broad variety of possible applications, e.g., in information retrieval, natural language processing, data mining, and computer vision, we have chosen document retrieval, the statistical analysis of noun/adjective co-occurrence and the unsupervised segmentation of textured images to test and evaluate the proposed algorithms.

This report describes research accomplished at the Center for Biological and Computational Learning, the Artificial Intelligence Laboratory of the Massachusetts Institute of Technology, the University of Bonn and the beaches of Cape Cod. Thomas Hofmann was supported by a M.I.T. Faculty Sponser's Discretionary Fund. Jan Puzicha was supported by the German Research Foundation (DFG) under grant # BU 914/3–1.

1 Introduction

The ultimate goal of statistical modeling is to explain observed data with a *probabilistic model.* In order to serve as a useful explanation, the model should reduce the complexity of the raw data and has to offer a certain degree of *simplification.* In this sense statistical modeling is related to the information theoretic concept of *minimum description length* [47, 48]. A model is a 'good' explanation for the given data if encoding the model and describing the data conditioned on that model yields a significant reduction in encoding complexity as compared to a 'direct' encoding of the data.

Complexity considerations are in particular relevant for the type of data investigated in this paper which is best described by the term *co-occurrence data* (COD) [30, 9]. The general setting is as follows: Suppose two finite sets $\mathcal{X} = \{x_1, \ldots, x_N\}$ and $\mathcal{Y} = \{y_1, \ldots, y_M\}$ of *abstract objects* with arbitrary labeling are given. As elementary observations we consider pairs $(x_i, y_j) \in \mathcal{X} \times \mathcal{Y}$, i.e., a joint occurrence of object x_i with object y_j. All data is numbered and collected in a sample set $\mathcal{S} = \{(x_{i(r)}, y_{j(r)}, r) : 1 \leq r \leq L\}$ with arbitrary ordering. The information in $\mathcal{S}$ is completely characterized by its sufficient statistics $n_{ij} = |\{(x_i, y_j, r) \in \mathcal{S}\}|$ which measure the *frequency* of co-occurrence of x_i and y_j. An important special case of COD are histogram data, where each object $x_i \in \mathcal{X}$ is characterized by a distribution of measured features $y_j \in \mathcal{Y}$. This becomes obvious by partitioning $\mathcal{S}$ into subsets $\mathcal{S}_i$ according to the $\mathcal{X}$–component, then the sample set $\mathcal{S}_i$ represent an empirical distribution $n_{j|i}$ over $\mathcal{Y}$, where $n_{j|i} \equiv n_{ij}/n_i$ and $n_i \equiv |\mathcal{S}_i|$.

Co-occurrence data is found in many distinctive application. An important example is information retrieval where $\mathcal{X}$ may correspond to a collection of documents and $\mathcal{Y}$ to a set of keywords. Hence n_{ij} denotes the number of occurrences of a word y_j in the (abstract of) document x_i. Or consider an application in computational linguistics, where the two sets correspond to words being part of a binary syntactic structure such as verbs with direct objects or nouns with corresponding adjectives [22, 43, 10]. In computer vision, $\mathcal{X}$ may correspond to image locations and $\mathcal{Y}$ to (discretized or categorical) feature values. The local histograms $n_{j|i}$ in an image neighborhood around x_i can then be utilized for a subsequent image segmentation [24]. Many more examples from data mining, molecular biology, preference analysis, etc. could be enumerated here to stress that analyzing co-occurrences of events is in fact a very general and fundamental problem of unsupervised learning.

In this contribution a general statistical framework for COD is presented. At a first glance, it may seem that statistical models for COD are trivial. As a consequence of the arbitrariness of the object labeling, both sets only have a purely nominal scale without ordering properties, and the frequencies n_{ij} capture all we know about the data. However, the intrinsic problem of COD is that of *data sparseness*, also known as the *zero frequency problem* [19, 18, 30, 64]. When N and M are very large, a majority of pairs (x_i, y_j) only have a small probability of occurring together in $\mathcal{S}$. Most of the counts n_{ij} will thus typically be zero or at least significantly corrupted by sampling noise, an effect which is largely independent of the underlying probability distribution. If the normalized frequencies are used in predicting future events, a large number of co-occurrences is observed which are judged to be impossible based on the data $\mathcal{S}$. The sparseness problem becomes still more urgent in the case of higher order COD where triplets, quadruples, etc are observed.[1] Even in the domain of natural language processing where large text corpora are available, one has rarely enough data to completely avoid this problem.

Typical state-of-the-art techniques in natural language processing apply smoothing techniques to deal with zero frequencies of unobserved events. Prominent techniques are, for example, the *back-off method* [30] which makes use of simpler lower order models and *model interpolation* with held–out data [28, 27]. Another class of methods are similarity–based local smoothing techniques as, e.g., proposed by Essen and Steinbiss [14] and Dagan et al. [9, 10]. An empirical comparison of smoothing techniques can be found in [7].

In information retrieval, there have been essentially two proposals to overcome the sparseness problem. The first class of methods relies on the *cluster hypothesis* [62, 20] which suggests to make use of inter–document similarities in order to improve the retrieval performance. Since it is often prohibitive to compute all pairwise similarities between documents these methods typically rely on random comparisons or random fractionation [8]. The second approach focuses on the index terms to derive an improved feature representation of documents. The by far most popular technique in this category is Salton's Vector Space Model [51, 58, 52] of which different variants have been proposed with different word weighting schemes [53]. A more recent variant known as latent semantics indexing [12] performs a dimension reduction by singular value decomposition. Related methods of feature selection have been proposed for text categorization, e.g., the term strength criterion [66].

In contrast, we propose a model–based statistical approach and present a family of *finite mixture models* [59, 35] as a way to deal with the data sparseness problem. Since mixture or class–based models can also be combined with other models our goal is orthogonal to standard interpolation techniques. Mixture models have the advantage to provide a sound statistical foundation with the calculus of probability theory as a powerful inference mechanism. Compared to the unconstrained table count 'model', mixture models offer a controllable way to reduce the number of free model parameters. As we will show, this significantly improves statistical inference and generalization to new data. The canonical way of complexity control is to vary the number of components in the mixture. Yet, we will introduce a different technique to avoid overfitting problems which relies on an *annealed* generalization of the classical EM algorithm [13]. As we will argue, annealed EM has some additional advantages making it an important tool for fitting mix-

[1] Word n-gram models are examples of such higher order co-occurrence data.

ture models.

Moreover, mixture models are a natural framework for unifying statistical inference and clustering. This is particularly important, since one is often interested in *discovering structure*, typically represented by groups of similar objects as in *pairwise data clustering* [23]. The major advantage of clustering based on COD compared to similarity-based clustering is the fact that it does not require an external similarity measure, but exclusively relies on the objects occurrence statistics. Since the models are directly applicable to co-occurrence and histogram data, the necessity for pairwise comparisons is avoided altogether.

Probabilistic models for COD have recently been investigated under the titles of class–based n-gram models [4], *distributional clustering* [43], and aggregate Markov models [54] in natural language processing. All three approaches are recovered as special cases in our COD framework and we will clarify the relation to our approach in the following sections. In particular we discuss the distributional clustering model which has been a major stimulus for our research in Section 3.

The rest of the paper is organized as follows: Section 2 introduces a mixture model which corresponds to a probabilistic grouping of object *pairs*. Section 3 then focuses on clustering models in the strict sense, i.e., models which are based on partitioning either one set of objects (asymmetric models) or both sets simultaneously (symmetric models). Section 4 presents a hierarchical model which combines clustering and *abstraction*. We discuss some improved variants of the standard EM algorithm in Section 5 and finally apply the derived algorithms to problems in information retrieval, natural language processing, and image segmentation in Section 6.

2 Separable Mixture Models

2.1 The Basic Model

Following the maximum likelihood principle we first specify a parametric model which generates COD over $\mathcal{X} \times \mathcal{Y}$, and then try to identify the parameters which assign the highest probability to the observed data. The first model proposed is the *Separable Mixture Model* (SMM). Introducing K abstract classes $\mathcal{C}_\alpha$ the SMM generates data according to the following scheme:

1. choose an abstract class $\mathcal{C}_\alpha$ according to a distribution π_α
2. select an object $x_i \in \mathcal{X}$ from a class-specific conditional distribution $p_{i|\alpha}$
3. select an object $y_j \in \mathcal{Y}$ from a class-specific conditional distribution $q_{j|\alpha}$

Note that steps 2. and 3. can be carried out independently. Hence, x_i and y_j are conditionally independent given the class $\mathcal{C}_\alpha$ and the joint probability distribution of the SMM is a mixture of separable component distributions which can be parameterized by[2]

$$p_{ij} \equiv P(x_i, y_j) = \sum_{\alpha=1}^{K} \pi_\alpha P(x_i, y_j|\alpha) = \sum_{\alpha=1}^{K} \pi_\alpha p_{i|\alpha} q_{j|\alpha}. \quad (1)$$

The number of independent parameters in the SMM is $(N + M - 1)K - 1$. Whenever $K \ll \min\{N, M\}$ this is significantly less than a complete table with NM entries. The complexity reduction is achieved by restricting the distribution to linear combinations of K separable component distributions.

2.2 Fitting the SMM

To optimally fit the SMM to given data $\mathcal{S}$ we have to maximize the log-likelihood

$$\mathcal{L} = \sum_{i=1}^{N}\sum_{j=1}^{M} n_{ij} \log \left(\sum_{\alpha=1}^{K} \pi_\alpha p_{i|\alpha} q_{j|\alpha} \right) \quad (2)$$

with respect to the model parameters $\theta = (\pi, p, q)$. To overcome the difficulties in maximizing a log of a sum, a set of unobserved variables is introduced and the corresponding *EM algorithm* [13, 36] is derived. EM is a general iterative technique for maximum likelihood estimation, where each iteration is composed of two steps:

- an Expectation (E) step for estimating the unobserved data or, more generally, averaging the complete data log–likelihood,
- and a Maximization (M) step, which involves maximization of the *expected log–likelihood* computed during the E–step in each iteration.

The EM algorithm is known to increase the likelihood in every step and converges to a (local) maximum of $\mathcal{L}$ under mild assumptions, cf. [13, 35, 38, 36].

Denote by $R_{r\alpha}$ an indicator variable to represent the unknown class $\mathcal{C}_\alpha$ from which the observation $(x_{i(r)}, y_{j(r)}, r) \in \mathcal{S}$ was generated. A set of indicator variables is summarized in a Boolean matrix $R \in \mathcal{R}$, where

$$\mathcal{R} = \left\{ R = (R_{r\alpha}) : \sum_{\alpha=1}^{K} R_{r\alpha} = 1 \right\} \quad (3)$$

denotes a space of Boolean assignment matrices. R effectively partitions the sample set $\mathcal{S}$ into K classes. Treating R as additional unobserved data, the *complete data log-likelihood* is given by

$$\mathcal{L}^c = \sum_{r=1}^{L}\sum_{\alpha=1}^{K} R_{r\alpha}(\log \pi_\alpha + \log p_{i(r)|\alpha} + \log q_{j(r)|\alpha}) \quad (4)$$

and the estimation problems for π, p and q decouple for given R.

The posterior probability of $R_{r\alpha}$ for a given parameter estimate $\hat{\theta}^{(t)}$ (E–step) is computed by exploiting Bayes' rule and is in general obtained by

$$\begin{aligned} \langle R_{r\alpha} \rangle &\equiv P(R_{r\alpha} = 1|\theta, \mathcal{S}) \\ &\propto P(\mathcal{S}|\theta, R_{r\alpha} = 1)\, P(R_{r\alpha} = 1|\theta). \end{aligned} \quad (5)$$

[2] The joint probability model in (1) was the starting point for the distributional clustering algorithm in [43], however the authors have in fact restricted their investigations to the (asymmetric) clustering model (cf. Section 3).

Thus for the SMM $\langle R_{r\alpha}\rangle$ is given in each iteration by

$$\langle R_{r\alpha}\rangle^{(t+1)} = \frac{\hat{\pi}_\alpha^{(t)}\ \hat{p}_{i(r)|\alpha}^{(t)}\ \hat{q}_{j(r)|\alpha}^{(t)}}{\sum_{\nu=1}^{K} \hat{\pi}_\nu^{(t)}\ \hat{p}_{i(r)|\nu}^{(t)}\ \hat{q}_{j(r)|\nu}^{(t)}} \ . \tag{6}$$

The M–step is obtained by differentiation of (4) using (6) as an estimate for $R_{r\alpha}$ and imposing the normalization constraints by the method of Lagrange multipliers. This yields a (normalized) summation over the respective posterior probabilities

$$\hat{\pi}_\alpha^{(t)} = \frac{1}{L}\sum_{r=1}^{L}\langle R_{r\alpha}\rangle^{(t)}, \tag{7}$$

$$\hat{p}_{i|\alpha}^{(t)} = \frac{1}{L\hat{\pi}_\alpha^{(t)}}\sum_{r:i(r)=i}\langle R_{r\alpha}\rangle^{(t)} \tag{8}$$

and

$$\hat{q}_{j|\alpha}^{(t)} = \frac{1}{L\hat{\pi}_\alpha^{(t)}}\sum_{r:j(r)=j}\langle R_{r\alpha}\rangle^{(t)} \ . \tag{9}$$

Iterating the E– and M–step, the parameters converge to a *local* maximum of the likelihood. Notice, that it is unnecessary to store all $L \cdot K$ posteriors, as the E- and M-step can be efficiently interleaved.

To distinguish more clearly between the different models proposed in the sequel, a representation in terms of directed graphical models (belief networks) is utilized. In this formalism, random variables as well as parameters are represented as nodes in a directed acyclic graph (cf. [41, 32, 17] for the general semantics of graphical models). Nodes of observed quantities are shaded and a number of i.i.d. observations is represented by a frame with a number in the corner to indicate the number of observations (called a *plate*). A graphical representation of the SMM is given in Fig. 1.

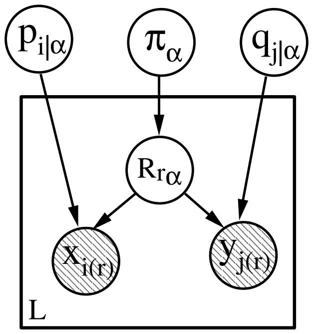

Figure 1: Graphical model representation for the symmetric parameterization of the Separable Mixture Model (SMM).

2.3 Asymmetric Formulation of the SMM

Our specification of the data generation procedure and the joint probability distribution in (1) is symmetric in $\mathcal{X}$ and $\mathcal{Y}$ and does not favor an interpretation of the abstract classes $\mathcal{C}_\alpha$ in terms of clusters of objects in $\mathcal{X}$ or $\mathcal{Y}$. The classes $\mathcal{C}_\alpha$ correspond to groups of *pair occurrences* which we call *aspects*. As we will see in comparison with the cluster–based approaches in Section 3 this is different from a 'hard' assignment of objects to clusters, but differs also from a probabilistic clustering of objects. Different observations involving the same $x_i \in \mathcal{X}$ (or $y_j \in \mathcal{Y}$) can be explained by different aspects and each objects has a particular distribution over aspects for its occurrences. This can be stressed by reparameterizing the SMM with the help of Bayes' rule

$$p_i = \sum_{i=1}^{N} p_{i|\alpha}\pi_\alpha, \quad \text{and} \ \ p_{\alpha|i} = \frac{p_{i|\alpha}\ \pi_\alpha}{p_i} \ . \tag{10}$$

The corresponding generative model, which is in fact equivalent to the SMM, is illustrated as a graphical model in Fig. 2. It generates data according to the following scheme:

1. select an object $x_i \in \mathcal{X}$ with probability p_i
2. choose an abstract class $\mathcal{C}_\alpha$ according to an object-specific conditional distribution $p_{\alpha|i}$
3. select an object $y_j \in \mathcal{Y}$ from a class-specific conditional distribution $q_{j|\alpha}$

The joint probability distribution of the SMM can thus be parameterized by

$$p_{ij} \equiv P(x_i, y_j) = p_i q_{j|i}, \quad q_{j|i} \equiv \sum_\alpha p_{\alpha|i} q_{j|\alpha} \ . \tag{11}$$

Hence a specific conditional distribution $q_{j|i}$ defined on $\mathcal{Y}$ is associated with each object x_i, which can be understood as a linear combination of the prototypical conditional distributions $q_{j|\alpha}$ weighted with probabilities $p_{\alpha|i}$ (cf. [43]). Notice, that although $p_{\alpha|i}$ defines a probabilistic assignment of objects to classes, these probabilities are not induced by the uncertainty of a hidden class membership of object x_i as is typically the case in mixture models. In the special case of $\mathcal{X} = \mathcal{Y}$ the SMM is equivalent to the word clustering model of Saul and Pereira [54] which has been developed parallel to our work. Comparing the graphical models in Fig. 1 and Fig. 2 it is obvious that the reparametrization simply corresponds to an arc reversal.

2.4 Interpreting the SMM in terms of Cross Entropy

To achieve a better understanding of the SMM consider the following cross entropy (Kullback–Leibler divergence) D between the empirical conditional distribution $n_{j|i} = n_{ij}/n_i$ of y_j given x_i and the conditional $q_{j|i}$ implied by the model,

$$\begin{aligned} D\big[n_{j|i}|q_{j|i}\big] &= \sum_{j=1}^{M} n_{j|i}\ \log\frac{n_{j|i}}{q_{j|i}} \qquad (12) \\ &= \sum_{j=1}^{M} n_{j|i}\log n_{j|i} - \frac{1}{n_i}\sum_{j=1}^{M} n_{ij}\log\sum_\alpha p_{\alpha|i}q_{j|\alpha}. \end{aligned}$$

Note, that the first entropy term does not depend on the parameters. Let $a(\mathcal{S}) = \sum_i n_i \sum_j n_{j|i}\log n_{j|i}$ and

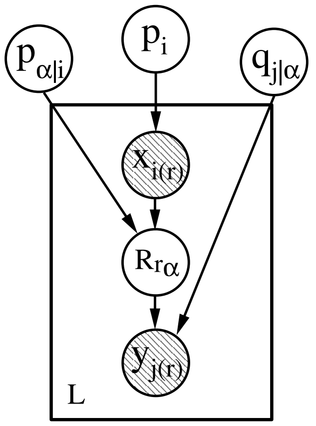

Figure 2: Graphical model representation for the Separable Mixture Model (SMM) using the asymmetric representation.

rewrite the observed data log-likelihood in (2) as

$$\mathcal{L} = -\sum_{i=1}^{N} n_i \; D\big[n_{j|i}|q_{j|i}\big] + \sum_{i=1}^{N} n_i \log p_i - a(\mathcal{S}) \ . \quad (13)$$

Since the estimation of $p_i = n_i$ can be carried out independently, the remaining parameters are obtained by optimizing a sum over cross entropies between conditional distributions weighted with n_i, the frequency of appearance of x_i in $\mathcal{S}$, i.e., by minimizing the cost function

$$\mathcal{H} = \sum_{i=1}^{N} n_i D\big[n_{j|i}|q_{j|i}\big] \quad . \quad (14)$$

Because of the symmetry of the SMM an equivalent decomposition is obtained by interchanging the role of the sets $\mathcal{X}$ and $\mathcal{Y}$.

2.5 Product-Space Mixture Model

The abstract classes $\mathcal{C}_\alpha$ of a fitted SMM correspond to aspects of observations, i.e., pair co-occurrences, and cannot be directly interpreted as a probabilistic grouping in either data space. But often we may want to enforce a *simultaneous* interpretation in terms of groups or probabilistic clusters in both sets of objects, because this may reflect prior belief or is part of the task. This can be achieved by imposing additional structure on the set of labels to enforce a product decomposition of aspects

$$\{1, \ldots, K\} \equiv \{1, \ldots, K_{\mathcal{X}}\} \times \{1, \ldots, K_{\mathcal{Y}}\} \ . \quad (15)$$

Each element α is now uniquely identified with a multi-index (ν_α, μ_α). The resulting *Product-Space Mixture Model* (PMM) has the joint distribution

$$p_{ij} = \sum_{\alpha=1}^{K} \pi_\alpha \; p_{i|\nu_\alpha} \; q_{j|\mu_\alpha}. \quad (16)$$

Here $\pi_\alpha = \pi_{\nu_\alpha \mu_\alpha}$ is the probability to generate an observation from a specific pair combination of clusters from $\mathcal{X}$ and $\mathcal{Y}$. The difference between the SMM and the PMM is the reduction of the number of conditional distributions $p_{i|\alpha}$ and $q_{j|\alpha}$ which also reduces the model complexity. In the SMM we have conditional distributions for each class $\mathcal{C}_\alpha$, while the PMM imposes additional constraints, $p_{i|\alpha} = p_{i|\beta}$, if $\nu_\alpha = \nu_\beta$ and $q_{j|\alpha} = q_{j|\beta}$, if $\mu_\alpha = \mu_\beta$. This is illustrated by the graphical model in Fig. 3.

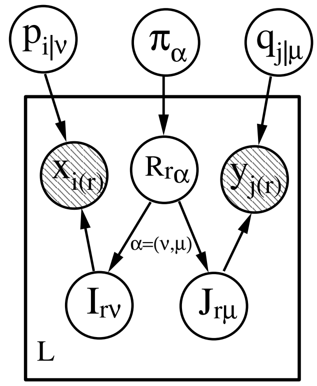

Figure 3: Graphical representation for the Product Space Mixture Model (PMM).

The PMM with $K_{\mathcal{X}}$ $\mathcal{X}$-classes and $K_{\mathcal{Y}}$ $\mathcal{Y}$-classes is thus a constrained SMM with $K = K_{\mathcal{X}} K_{\mathcal{Y}}$ abstract classes. The number of independent parameters in the PMM reduces to

$$(K_{\mathcal{X}} K_{\mathcal{Y}} - 1) + K_{\mathcal{X}}(N-1) + K_{\mathcal{Y}}(M-1) \ . \quad (17)$$

Whether this is an advantage over the unconstrained SMM depends on the specific data generating process. The only difference in the fitting procedure compared to the SMM occurs in the M–step by substituting

$$\hat{p}_{i|\nu}^{(t)} \propto \sum_{r:i(r)=i} \sum_{\alpha:\nu_\alpha=\nu} \langle R_{r\alpha}\rangle^{(t)}, \quad (18)$$

$$\hat{q}_{j|\mu}^{(t)} \propto \sum_{r:j(r)=j} \sum_{\alpha:\mu_\alpha=\mu} \langle R_{r\alpha}\rangle^{(t)} \ . \quad (19)$$

The E–step has to be adapted with respect to the modified labeling convention.

3 Clustering Models

The grouping structure inferred by the SMM corresponds to a probabilistic partitioning of the observation space $\mathcal{X} \times \mathcal{Y}$. Although the conditional probabilities $p_{\alpha|i}$ and $q_{\alpha|j}$ can be interpreted as class membership probabilities of objects in $\mathcal{X}$ or $\mathcal{Y}$, they more precisely correspond to object–specific distributions over aspects. Yet, depending on the application at hand it might be more natural to assume a typically unknown, but nevertheless definitive assignment of objects to clusters, in particular when the main interest lies in extracting *grouping structure* in $\mathcal{X}$ and/or $\mathcal{Y}$ as is often the case in exploratory data analysis tasks. Models where each object is assigned to exactly one cluster are referred to as *clustering models* in the strict sense and they should be treated as models in their own right. As we will demonstrate they have the further advantage to reduce the model complexity compared to the aspect–based SMM approach.

3.1 Asymmetric Clustering Model

A modification of the SMM leads over to the *Asymmetric Clustering Model* (ACM). In the original formulation of the data generation process for the SMM the assumption was made that each observation (x_i, y_j) is generated from a class $\mathcal{C}_\alpha$ according to the class-specific distribution $p_{i|\alpha}\, q_{j|\alpha}$ or, equivalently, the conditional distribution $q_{j|i}$ was a linear combination of probabilities $q_{j|\alpha}$ weighted according to the distribution $p_{\alpha|i}$. Now we restrict this choice for each object x_i to a *single* class. This implies that all y_j occuring in observations (x_i, y_j) involving the same object x_i are assumed to be generated from an identical conditional distribution $q_{j|\alpha}$. Let us introduce an indicator variable $I_{i\alpha}$ for the class membership which allows us to specify a probability distribution by

$$P(x_i, y_j | I, p, q) = p_i \sum_{\alpha=1}^{K} I_{i\alpha} q_{j|\alpha} \quad . \tag{20}$$

The ACM can be understood as a SMM with $I_{i\alpha}$ replacing $p_{\alpha|i}$. The model introduces an asymmetry by clustering only one set of objects $\mathcal{X}$, while fitting class conditional distributions for the second set $\mathcal{Y}$. Obviously, we can interchange the role of $\mathcal{X}$ and $\mathcal{Y}$ and may obtain two distinct models.

For a sample set $\mathcal{S}$ the log–likelihood is given by

$$\mathcal{L} = \sum_{i=1}^{N} n_i \log p_i + \sum_{i=1}^{N}\sum_{\alpha=1}^{K} I_{i\alpha} \sum_{j=1}^{M} n_{ij} \log q_{j|\alpha} \quad . \tag{21}$$

The maximum likelihood equations are

$$\hat{p}_i = n_i / L, \tag{22}$$

$$\hat{I}_{i\alpha} = \begin{cases} 1 & \text{if } \alpha = \arg\min_\nu D\left[n_{j|i} |\, \hat{q}_{j|\nu}\right] \\ 0 & \text{else,} \end{cases} \tag{23}$$

$$\hat{q}_{j|\alpha} = \frac{\sum_{i=1}^{N} \hat{I}_{i\alpha} n_{ij}}{\sum_{i=1}^{N} \hat{I}_{i\alpha} n_i} = \sum_{i=1}^{N} \frac{\hat{I}_{i\alpha} n_i}{\sum_{k=1}^{N} \hat{I}_{k\alpha} n_k} n_{j|i} \,. \tag{24}$$

The class-conditional distributions $\hat{q}_{j|\alpha}$ are linear superpositions of all empirical distributions of objects x_i in cluster $\mathcal{C}_\alpha$. Eq. (24) is thus a simple *centroid condition*, the *distortion measure* being the cross entropy or Kullback–Leibler divergence. In contrast to (13) where the maximum likelihood estimate for the SMM minimizes the cross entropy by fitting $p_{i|\alpha}$, the cross entropy serves as a distortion measure in the ACM. The update scheme to solve the likelihood equations is structurally very similar to the K–means algorithm: calculate assignments for given centroids according to the nearest neighbor rule and recalculate the centroid distributions in alternation.

The ACM is in fact similar to the distributional clustering model proposed in [43] as the minimization of

$$\mathcal{H} = \sum_{i=1}^{N}\sum_{\nu=1}^{K} I_{i\nu} D\left[n_{j|i} | q_{j|\nu}\right] \quad . \tag{25}$$

In distributional clustering, the KL–divergence as a distortion measure for distributions has been motivated by the fact that the centroid equation (24) is satisfied at stationary points[3]. Yet, since after dropping the independent p_i parameters and a data dependent constant we arrive at

$$\mathcal{L} = \sum_{i=1}^{N} n_i \sum_{\alpha=1}^{K} I_{i\alpha} D\left[n_{j|i} | q_{j|\alpha}\right] , \tag{26}$$

showing that the choice of the KL-divergence simply follows from the likelihood principle. We like to point out the non–negligible difference between the distributional clustering cost function in (25) and the likelihood in (26), which weights the object specific contributions by their empirical frequencies n_i. This implies that objects with large sample sets $\mathcal{S}_i$ have a larger influence on the optimization of the data partitioning, since they account for more observations, as opposed to a constant influence in the distributional clustering model.

3.2 EM algorithm for Probabilistic ACM

Instead of interpreting the cluster memberships $I_{i\alpha}$ as model parameters, we may also consider them as *unobserved variables*. In fact, this interpretation is consistent with other common mixture models [35, 59] and might be preferred in the context of statistical modeling, in particular if N scales with L. Therefore consider the following complete data distribution

$$P(\mathcal{S}, I | \rho, p, q) = P(\mathcal{S}|I, p, q) P(I|\rho), \tag{27}$$

$$P(I|\rho) = \prod_{i=1}^{N} \rho_\alpha^{I_{i\alpha}} \,. \tag{28}$$

Here ρ specifies a prior probability for the hidden variables, $P(I_{i\alpha} = 1|\rho) = \rho_\alpha$.

The introduction of unobservable variables $I_{i\alpha}$ yields an EM-scheme and replaces the arg min-evaluation in (23) with posterior probabilities

$$\langle I_{i\alpha}\rangle^{(t+1)} = \frac{\hat{\rho}_\alpha^{(t)} \prod_{j=1}^{M} \left(\hat{q}_{j|\alpha}^{(t)}\right)^{n_{ij}}}{\sum_{\nu=1}^{K} \hat{\rho}_\nu^{(t)} \prod_{j=1}^{M} \left(\hat{q}_{j|\nu}^{(t)}\right)^{n_{ij}}} \tag{29}$$

$$= \frac{\hat{\rho}_\alpha^{(t)} \exp\left(-n_i D\left[n_{j|i} | \hat{q}_{j|\alpha}^{(t)}\right]\right)}{\sum_{\nu=1}^{K} \hat{\rho}_\nu^{(t)} \exp\left(-n_i D\left[n_{j|i} | \hat{q}_{j|\nu}^{(t)}\right]\right)} \,. \tag{30}$$

The M–step is equivalent to (24) with posteriors replacing Boolean variables. Finally, the additional mixing proportions are estimated in the M–step by

$$\hat{\rho}_\alpha^{(t)} = \frac{1}{N} \sum_{i=1}^{N} \langle I_{i\alpha}\rangle^{(t)} \quad . \tag{31}$$

Notice the crucial difference compared to SMM posteriors in (6): Since all indicator variables $R_{r\alpha}$ belonging to the same x_i are identified, the likelihoods for observations in $\mathcal{S}_i$ are collected in a product before they are suitably normalized. To illustrate the difference consider the following example. Let a fraction s of the observed

[3] This is in fact not a unique property of the KL–divergence as it is also satisfied for the Euclidean distance.

pairs involving x_i be best explained by assigning x_i to $\mathcal{C}_\alpha$ while the remaining fraction $1-s$ of the data is best explained by assigning it to $\mathcal{C}_\nu$. Fitting the SMM model approximately results in $p_{\alpha|i}=s$ and $p_{\nu|i}=1-s$, irrespective of the number of observations, because posteriors $\langle I_{r\alpha}\rangle$ are additively collected. In the ACM all contributions first enter a huge product. In particular, for $n_i \to \infty$ the posteriors $\langle I_{i\alpha}\rangle$ approach Boolean values and automatically result in a hard partitioning of $\mathcal{X}$. Compared to the original distributional clustering model as proposed in [43] our maximum likelihood approach naturally includes additional parameters for the mixing proportions ρ.

Notice that in this model, to which we refer as probabilistic ACM, different observations involving the same object x_i are actually not independent, even if conditioned on the parameters (ρ, p, q). As a consequence, considering the predictive probability for an additional observation $s=(x_i, y_j, L+1)$ requires to condition on the given sample set $\mathcal{S}$, more precisely on the subset $\mathcal{S}_i$, which yields

$$\begin{aligned} P(s|\mathcal{S},\rho,p,q) &= \sum_{\alpha=1}^{K} P(s|I_{i\alpha}{=}1,p,q)\, P(I_{i\alpha}{=}1|\mathcal{S}_i,\rho,q) \\ &= p_i \sum_{\alpha=1}^{K} q_{j|\alpha}\langle I_{i\alpha}\rangle \ . \end{aligned} \tag{32}$$

Thus we have to keep the posterior probabilities in addition to the parameters (p,q) in order to define a (predictive) probability distribution on co-occurrence pairs. The corresponding graphical representation in Fig. 4 stresses the fact that observations with identical $\mathcal{X}$–objects are coupled by the hidden variable $I_{i\alpha}$ [4].

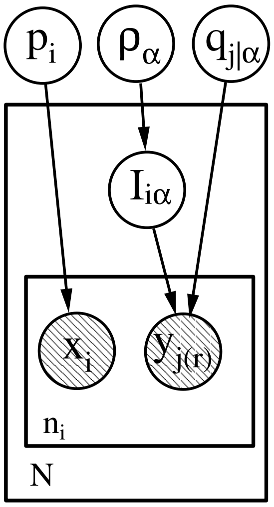

Figure 4: Graphical representation for the Asymmetric Clustering Model (ACM).

[4] Notice that n_i is not interpreted as a random variable, but treated as a given quantity.

3.3 Symmetric Clustering Model

We can classify the models discussed so far w.r.t. the way they model $p_{\alpha|i}$: (i) as an arbitrary probability distribution (SMM), (ii) as an unobserved hidden variable ($p_{\alpha|i}=\langle I_{i\alpha}\rangle$, probabilistic ACM), (iii) as a Boolean variable ($p_{\alpha|i}=I_{i\alpha}\in\{0,1\}$, hard ACM). On the other hand, no model imposes restrictions on the conditional distributions $q_{j|\alpha}$. This introduces the asymmetry in ACM, where $p_{\alpha|i}$ and hence $p_{i|\alpha}$ is actually restricted, while $q_{j|\alpha}$ is not. However, it also indicates a way to derive symmetric clustering models, namely by imposing the same constraints on conditional distributions $p_{i|\alpha}$ and $q_{j|\alpha}$. Unfortunately, naively modifying the SMM for both object sets simultaneously does not result in a reasonable model. Introducing indicator variables $I_{i\alpha}$ and $J_{j\alpha}$ to replace $p_{\alpha|i}$ and $q_{\alpha|j}$ yields a joint probability

$$p_{ij} \quad\propto\quad p_i q_j \sum_{\alpha=1}^{K} I_{i\alpha} J_{j\alpha}\, \pi_\alpha \tag{33}$$

which can be normalized to yield a valid probability distribution, but which is zero for pairs (x_i, y_j), whenever $\sum_\alpha I_{i\alpha}J_{j\alpha}=0$ and therefore results in zero probabilities for each co-occurrence of pairs not belonging to the same cluster $\mathcal{C}_\alpha$.

For the PMM, however, the corresponding clustering model is more interesting. Let us introduce *cluster association* parameters $c_{\nu\mu}$ and define the joint probability distribution of the *symmetric clustering model* (SCM) by

$$p_{ij} \quad=\quad p_i q_j \sum_{\nu,\mu} I_{i\nu} J_{j\mu} c_{\nu\mu}\ , \tag{34}$$

where we have to impose the global normalization constraint

$$\sum_{i=1}^{N}\sum_{j=1}^{M} p_{ij} = \sum_{i=1}^{N}\sum_{j=1}^{M} p_i q_j \sum_{\nu,\mu} I_{i\nu}J_{j\mu}c_{\nu\mu} = 1\ . \tag{35}$$

In the sequel, we will add more constraints to break certain invariances w.r.t. multiplicative constants, but for now no restrictions besides (35) are enforced.

Introducing a Lagrange multiplier λ results in the following augmented log–likelihood

$$\begin{aligned} \mathcal{L} \quad=\quad & \sum_{i,j} n_{ij} \sum_{\nu,\mu} I_{i\nu}J_{j\mu}(\log p_i + \log q_j + \log c_{\nu\mu}) \\ & +\lambda\left(\sum_{i,j} p_i q_j \sum_{\nu,\mu} I_{i\nu}J_{j\mu}c_{\nu\mu} - 1\right)\ . \end{aligned} \tag{36}$$

The corresponding stationary equations for the maximum likelihood estimators are

$$\hat{p}_i = -\frac{n_i}{\lambda\sum_{\nu,\mu} I_{i\nu}c_{\nu\mu}\sum_j q_j J_{j\mu}}, \tag{37}$$

$$\hat{q}_j = -\frac{m_j}{\lambda\sum_{\nu,\mu} J_{j\mu}c_{\nu\mu}\sum_i p_i I_{i\nu}},\quad m_j = \sum_i n_{ij}, \tag{38}$$

$$\hat{c}_{\nu\mu} = -\frac{\sum_{i=1}^{N}\sum_{j=1}^{M} n_{ij} I_{i\nu} J_{j\mu}}{\lambda(\sum_{i=1}^{N} p_i I_{i\nu})(\sum_{j=1}^{M} q_j J_{j\mu})}, \tag{39}$$

together with (35) which is obtained from (36) by differentiation with respect to λ. Substituting the right-hand side of (39) into (35) results in the following equation for λ,

$$\lambda = -\sum_{i,j} n_{ij} \sum_{\nu,\mu} I_{i\nu} J_{j\mu} = -L\,. \tag{40}$$

Inserting $\lambda = -L$ into (39) gives an explicit expression for $\hat{c}_{\nu\mu}$ which depends on p and q. Substituting this expression back into (37,38) yields

$$\hat{p}_i \;=\; n_i \sum_{\nu} I_{i\nu} \frac{\sum_k \hat{p}_k I_{k\nu}}{\sum_k n_k I_{k\nu}} \tag{41}$$

and an equivalent expression for $\hat{q}_j$. Observing that the fraction on the right-hand side in (41) does not depend on the specific index i, the self-consistency equations can be written as $\hat{p}_i = n_i \sum_\nu I_{i\nu} a_\nu$. It is straightforward to verify that: (i) any choice of constants $a_\nu \in \mathbb{R}^+$ gives a valid solution and (ii) each such solution corresponds to a (local) maximum of $\mathcal{L}$. This is because a simultaneous re-scaling of all $\hat{p}_i$ for which $I_{i\nu} = 1$ is compensated by a reciprocal change in $\hat{c}_{\nu\mu}$ as is obvious from (34) or (39), leaving the joint probability unaffected.

We break this scale invariance by imposing the additional conditions $\sum_i p_i I_{i\nu} = \pi^x_\nu \equiv \sum_i n_i I_{i\nu}/L$ and $\sum_j q_j J_{j\mu} = \pi^y_\mu \equiv \sum_j m_j J_{j\mu}/L$ on the parameters. This has the advantage to result in the simple estimators $\hat{p}_i = n_i/L$ and $\hat{q}_j = m_j/L$, respectively. The proposed choice decouples the estimation of p and q from all other parameters. Moreover it supports an interpretation of $\hat{p}_i$ and $\hat{q}_j$ in terms of marginal probabilities, while π^x_ν and π^y_μ correspond to occurrence frequencies of objects from a particular cluster. With the above constraints the final expression for the maximum likelihood estimates for $c_{\nu\mu}$ becomes

$$\hat{c}_{\nu\mu} \;=\; \frac{\pi_{\nu\mu}}{\pi^x_\nu \pi^y_\mu}, \quad \text{where} \tag{42}$$

$$\pi_{\nu\mu} \;\equiv\; \sum_{i=1}^{N}\sum_{j=1}^{M} \frac{n_{ij}}{L} I_{i\nu} J_{j\mu}\,. \tag{43}$$

$\pi_{\nu\mu}$ can be interpreted as an estimate of the joint probability for the cluster pair (ν, μ). Notice that the auxiliary parameters are related by marginalization $\sum_\mu \pi_{\nu\mu} = \pi^x_\nu$ and $\sum_\nu \pi_{\nu\mu} = \pi^y_\mu$. The maximum likelihood estimate $\hat{c}_{\nu\mu}$ is a quotient of the joint frequency of objects from clusters $\mathcal{C}^x_\nu$ and $\mathcal{C}^y_\mu$ and the product of the respective marginal cluster frequencies. If we treat $\hat{c}_{\nu\mu}$ as functions of I, J and insert (42) into (36), this results in a term which represents the average *mutual information* between the random events $x_i \in \mathcal{C}^x_\nu$ and $y_j \in \mathcal{C}^y_\mu$. Maximizing $\mathcal{L}$ w.r.t. I and J thus maximizes the mutual information which is very satisfying, since it gives the SCM a precise interpretation in terms of an information theoretic concept. A similar criterion based on mutual information has been proposed by Brown et al. [4] in their class–based n–gram model. More precisely their model is a special case of the (hard clustering) SCM, where formally $\mathcal{X} = \mathcal{Y}$ and $I_{i\nu} = J_{i\nu}$.[5]

[5] For the bigram model in [4] this implies that the word classes are implicitly utilized in two different ways: as classes for predicting and for being predicted. It is not obvious that this is actually an unquestionable choice.

The coupled K–means like equations for either set of discrete variables are obtained by maximizing the augmented likelihood in (36) from which we deduce

$$\hat{I}_{i\alpha} = \begin{cases} 1 & \text{if } \alpha = \arg\max_\nu h_{i\nu} \\ 0 & \text{else} \end{cases}, \text{ with} \tag{44}$$

$$h_{i\nu} \equiv \sum_j \sum_\mu J_{j\mu} \left(n_{ij} \log \hat{c}_{\nu\mu} - \frac{n_i m_j}{L} \hat{c}_{\nu\mu} \right)$$

The expression for $h_{i\nu}$ further simplifies, because

$$\sum_{j,\mu} \frac{n_i m_j}{L} J_{j\mu} \hat{c}_{\nu\mu} = n_i \sum_\mu \pi^y_\mu \frac{\pi_{\nu\mu}}{\pi^x_\nu \pi^y_\mu} = n_i \frac{\sum_\mu \pi_{\nu\mu}}{\pi^x_\nu} = n_i \tag{45}$$

is a constant independent of the cluster index ν which can be dropped in the maximization in (44). It has to be stressed that the manipulations in (45) have made use of the fact that the J–variables appearing in (42) and (45) can be identified; the $\hat{c}$–estimates have thus to be J–consistent[6]. This can be made more plausible by verifying that for given J and J–consistent parameters, the marginalization $\sum_j p_{ij} = p_i$ holds independently of the specific choice of I. This automatically ensures the global normalization since $\sum_i \hat{p}_i = 1$ which in turn explains why the global constraint does not affect the optimal choices of $\hat{I}$ according to (44). Similar equations can be obtained for J,

$$\hat{J}_{j\alpha} = \begin{cases} 1 & \text{if } \alpha = \arg\max_\mu \sum_i n_{ij} \sum_\nu I_{i\nu} \log \hat{c}_{\nu\mu} \\ 0 & \text{else} \end{cases}.$$

The nature of the simultaneous clustering suggests an alternating minimization procedure where the $\mathcal{X}$–partition is optimized for a fixed $\mathcal{Y}$–partition and vice versa. After each update step for either partition the estimators $\hat{c}_{\nu\mu}$ have to be updated in order to ensure the validity of (45), i.e., the update sequence $(\hat{I}, \hat{c}, \hat{J}, \hat{c})$ is guaranteed to increase the likelihood in every step. Notice that the cluster association parameters effectively decouple the interactions between assignment variables $I_{i\nu}$, $I_{k\nu}$ for different x_i, x_k and between assignment variables $J_{j\mu}$, $J_{l\mu}$ for different y_j, y_l. Although we could in principle insert the expression in (36) directly into the likelihood and derive a local maximization algorithm for I and J (cf. [4]), this would result in much more complicated stationary conditions than (44). The decoupling effect is even more important for the probabilistic SCM derived in the next section.

3.4 Probabilistic SCM

As for the ACM, we now investigate the approach preferred in statistics and treat the discrete I and J as hidden variables. The situation is essentially the same as for the ACM. The complete data distribution for the probabilistic SCM (PSCM) is given by

$$P(\mathcal{S}, I, J|\rho^x, \rho^y, c) = \left[\prod_{i,j} \sum_{\nu,\mu} I_{i\nu} J_{j\mu} c_{\nu\mu}^{n_{ij}} \right] \tag{46}$$

[6] Hereby we mean that $\hat{c}_{\nu\mu}$ depends on J by the formula in (42), where I can be an *arbitrary* partitioning of the $\mathcal{X}$–space.

$$\times \left[\prod_i p_i^{n_i} \sum_\nu I_{i\nu}\rho_\nu^x\right] \left[\prod_j q_j^{m_j} \sum_\mu J_{j\mu}\rho_\mu^y)\right]$$

and the predictive probability for $s = (x_i, y_j, L+1)$ now involves joint posteriors,

$$P(s|\mathcal{S},\ldots) \quad = \quad p_i q_j \sum_{\nu,\mu} \langle I_{i\nu} J_{j\mu}\rangle c_{\nu\mu}\,. \tag{47}$$

The M–step equations are obtained from (42) by replacing (products of) Boolean variables by their posteriors. The estimates for the ρ–parameters are given by

$$\hat{\rho}_\nu^{x(t)} = \frac{\sum_{i=1}^N \langle I_{i\nu}\rangle^{(t)}}{N}, \quad \hat{\rho}_\mu^{y(t)} = \frac{\sum_{j=1}^M \langle J_{i\mu}\rangle^{(t)}}{M}. \tag{48}$$

The coupling of I and J makes the exact computation of posteriors in the E–step intractable. To preserve tractability of the procedure we propose to apply a factorial approximation (called mean–field approximation, cf. Appendix A), $\langle I_{i\nu} J_{j\mu}\rangle \approx \langle I_{i\nu}\rangle\langle J_{j\mu}\rangle$, which results in the following approximations for the marginal posterior probabilities

$$\langle I_{i\nu}\rangle \propto \hat{\rho}_\nu^x \exp\left[-\sum_j n_{ij} \sum_\mu \langle J_{j\mu}\rangle \log \hat{c}_{\nu\mu}\right] \tag{49}$$

and a similar equation for $\langle J_{j\mu}\rangle$. The mean–field equations can be more intuitively understood as a soft-max versions of the hard clustering equations with additional priors ρ^x, ρ^y. Alternatively, one may also apply Markov chain Monte Carlo (MCMC) methods to approximate the required correlations. Yet, the mean–field approximation has the advantage to be more efficient due to its deterministic nature. Notice that the mean–field conditions (49) form a highly non-linear, coupled system of equations. A solution is found by a fixed–point iteration which alternates the update of posterior marginals with an update of the continuous parameters ($\langle I\rangle$, $(\hat{c}, \hat{\rho}^x)$, $\langle J\rangle$, $(\hat{c}, \hat{\rho}^y)$ sequence). This optimizes a common objective function in every step and always maintains a valid probability distribution.[7]

3.5 Overview

Altogether we have derived six different model types for COD[8] which are summarized in the following systematic scheme in Table 1. As can be seen the models span a large range of model complexity by imposing constraints on the class conditional distributions $p_{i|\alpha}$ and $q_{j|\alpha}$.

Model	$p_{i\|\alpha}$ $\alpha = (\nu, \mu)$	$q_{j\|\alpha}$ $\alpha = (\nu, \mu)$
SMM	unconstr.	unconstr.
PMM	$p_{i\|\nu}$	$q_{i\|\mu}$
(hard) ACM	$\frac{I_{i\alpha} p_i}{\pi_\alpha}$	unconstr.
(probabilistic) ACM	$\frac{\langle I_{i\alpha}\rangle p_i}{\pi_\alpha}$	unconstr.
(hard) SCM	$\frac{I_{i\nu} p_i}{\pi_\nu^x}$	$\frac{J_{i\mu} q_j}{\pi_\mu^y}$
(probabilistic) SCM	$\frac{\langle I_{i\nu}\rangle p_i}{\pi_\nu^x}$	$\frac{\langle J_{i\mu}\rangle q_j}{\pi_\mu^y}$

Table 1: Systematic overview of presented COD models.

4 Hierarchical Clustering Model

4.1 Model Specification

In this section we present a novel *hierarchical* generative model, called HACM. Assume therefore a tree topology $\mathcal{T}$ on the clusters to be given, e.g., a complete binary tree. Clusters $\mathcal{C}_\alpha$ are identified with the terminal nodes of $\mathcal{T}$. Conditional probabilities $q_{j|\alpha}$ are not only attached to the leaves, but also to all inner nodes of the tree. The HACM involves two stages. In the first step, which is similar to the ACM case, each object x_i is assigned to one component or cluster $\mathcal{C}_\alpha$ represented by an (unobserved) variable $I_{i\alpha}$. Now, instead of generating all n_i observations from the conditional distribution $q_{j|\alpha}$, a second probabilistic sampling step is involved by selecting a *resolution* or *abstraction level*. Hence, the first step is a selection from a *horizontal mixture* of possible paths for each *object* x_i, while the second step is a probabilistic selection of a component from the *vertical mixture* of nodes on the selected path for each *observation*. To model the vertical selection we introduce a second set of hidden variables $V_{r\nu}$ which encode the resolution level $\mathcal{A}_\nu$ for the r-th observation. Notice the different nature of both sets of variables: the I variables represent a partitioning in the $\mathcal{X}$ space similar to the ACM, while V partitions the co-occurrence space $\mathcal{X} \times \mathcal{Y}$ like in the SMM. Obviously, the hidden variables are not independent and have to fulfill the following set of constraints induced by $\mathcal{T}$:

$$\sum_\alpha \sum_{\mathcal{A}_\nu \uparrow \mathcal{C}_\alpha} I_{i(r)\alpha} V_{r\nu} = 1, \;\; \forall r, \tag{50}$$

where $\mathcal{A}_\nu \uparrow \mathcal{C}_\alpha$ denotes the nodes $\mathcal{A}_\nu$ 'above' $\mathcal{C}_\alpha$, i.e., nodes on the path to $\mathcal{C}_\alpha$. The constraints ensure that once x_i is assigned to a cluster $\mathcal{C}_\alpha$ *all* observation in $\mathcal{S}_i$ are restricted to be generated from one of the abstraction nodes $\mathcal{A}_\nu$ on the path above $\mathcal{C}_\alpha$. A pictorial representation can be found in Fig. 5: for an object x_i assigned to $\mathcal{C}_\alpha$ the choices for abstraction levels of observations (x_i, y_j, r) are restricted to the 'active' (highlighted) vertical path. The 'tension' in this model is imposed by the

[7] The EM approach for SCM has problems when started from a random initialization, because all posteriors typically approach uniform distributions. A remedy which we applied is to utilize a solution of the ACM to initialize one arbitrarily selected set of hidden variables.

[8] We ignore the variants obtained from asymmetric models by reversing the role of $\mathcal{X}$ and $\mathcal{Y}$ for simplicity.

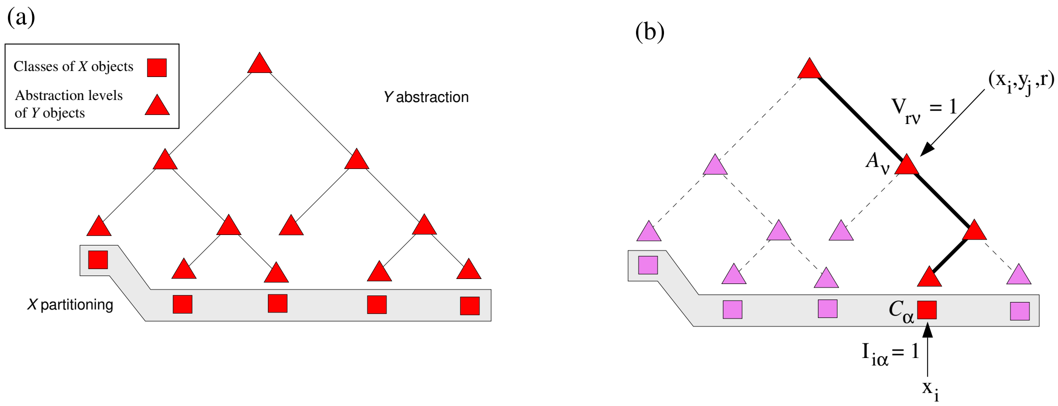

Figure 5: Scheme for data generation with the HACM.

constraints on $I_{i(r)\alpha}V_{r\nu}$ as opposed to the independent choices $R_{r\alpha}$ in the SMM.

To complete the specification of the HACM one has to specify prior probabilities for $V_{r\nu}$. The most general choice is to condition the priors on the terminal node selected by the I variables and on x_i itself, i.e., to introduce parameters $\tau_{\nu|\alpha,i}$. This assumes that each object has a specific distribution over abstraction levels. For simplicity, the constraints in (50) are incorporated in the prior by setting $\tau_{\nu|\alpha,i} = 0$ whenever $\mathcal{A}_\nu \not\uparrow \mathcal{C}_\alpha$. With the above definitions the complete data log–likelihood of the HACM is given by

$$\mathcal{L}^c = \sum_{i,j} n_{ij} \sum_\alpha I_{i\alpha} \sum_{\mathcal{S}_i} \sum_\nu V_{r\nu} \log \tau_{\nu|\alpha,i}\, q_{j|\nu} + \sum_i n_i \log p_i + \sum_i \sum_\alpha I_{i\alpha} \log \rho_\alpha\,. \quad (51)$$

The corresponding representation in terms of a graphical model is shown in Fig. 6.

We may think of the HACM 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. The HACM is more general than the ACM in that it allows to use a different abstraction level for each observation. The class conditional probability $q_{j|\alpha}$ is now modeled by a vertical mixture offering additional degrees of freedom. In fact the ACM is retrieved as a special case by setting $\tau_{\nu|\alpha,i} = \delta_{\nu\alpha}$. On the other hand, the HACM is more constrained than the SMM since the mixing of component densities $q_{j|\alpha}$ is restricted to nodes on a single path through the tree. This restriction precisely expresses why we obtain a hierarchical clustering organization of objects in $\mathcal{X}$ and, simultaneously, an abstractive organization of objects in $\mathcal{Y}$. The dual organization is in particular interesting for information retrieval where the HACM can be used to cluster documents ($\equiv \mathcal{X}$) in a hierarchical fashion and simultaneously assign abstraction levels to the different keywords ($\equiv \mathcal{Y}$).

4.2 EM Algorithm for the HACM

Skipping the derivation of the maximum likelihood equations for the hard clustering case of HACM, we directly continue with the probabilistic EM version.

In the E–step we need to calculate posterior probabilities $\langle I_{i(r)\alpha}V_{r\nu}\rangle$. Since the values of the clustering variables $I_{i\alpha}$ restrict the admissible values of $V_{r\nu}$, we may compute the joint posterior probabilities from the chain rule

$$P(I_{i(r)\alpha}V_{r\nu}{=}\,1|\mathcal{S},\theta) = P(V_{r\nu}{=}\,1|\mathcal{S}_{i(r)},\theta, I_{i(r)\alpha} = 1)\; P(I_{i(r)\alpha}{=}\,1|\mathcal{S}_{i(r)},\theta), \quad (52)$$

where $\theta = (p, q, \rho, \tau)$ summarizes all continuous parameters. The conditional posterior probabilities for $V_{r\nu}$ are given by

$$\langle V_{r\nu|\alpha}\rangle^{(t+1)} = \frac{\hat{\tau}^{(t)}_{\nu|\alpha,i(r)}\hat{q}^{(t)}_{j(r)|\nu}}{\sum_\mu \hat{\tau}^{(t)}_{\mu|\alpha,i(r)}\hat{q}^{(t)}_{j(r)|\mu}}\,. \quad (53)$$

The marginal posteriors can by definition be computed from

$$\langle I_{i\alpha}\rangle = \sum_{\{V\}} P(I_{i\alpha} = 1, V|\mathcal{S},\theta) \quad (54)$$

which yields

$$\langle I_{i\alpha}\rangle^{(t+1)} \propto \hat{\rho}^{(t)}_\alpha \prod_{\mathcal{S}_i} \sum_\nu \hat{\tau}^{(t)}_{\nu|\alpha,i}\, \hat{q}^{(t)}_{j(r)\nu}, \quad (55)$$

completing the E–step.

In the M–step all parameters are updated according to the following set of equations

$$\hat{\rho}^{(t)}_\alpha = \frac{1}{N}\sum_i \langle I_{i\alpha}\rangle^{(t)} \quad (56)$$

$$\hat{q}^{(t)}_{j\nu} \propto \sum_{r:j(r)=j} \sum_\alpha \langle V_{r\nu|\alpha}\rangle^{(t)}\langle I_{i\alpha}\rangle^{(t)} \quad (57)$$

$$\hat{\tau}^{(t)}_{\nu|\alpha,i} = \frac{1}{|\mathcal{S}_i|}\sum_{S_i}\langle V_{r\nu|\alpha}\rangle^{(t)} \quad (58)$$

Using the HACM in large–scale applications requires a closer investigation of the computational complexity

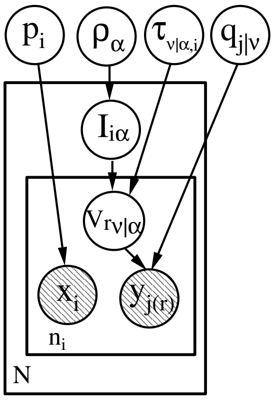

Figure 6: Graphical representation for the Hierarchical Asymmetric Clustering Model (HACM).

of the EM model fitting procedure. The major computational burden of the algorithm is the re–calculation of posterior probabilities $\langle V_{r\nu|\alpha}\rangle$ in the E–step. To accelerate the EM algorithm we exploit the fact that the hierarchical and abstractive organization does not critically depend on the parameters $\tau_{\nu|\alpha,i}$. By this we mean, that even setting $\tau_{\nu|\alpha,i} = const.$ (for all $\mathcal{A}_\nu \uparrow \mathcal{C}_\alpha$) will result in a reasonable model. In fact, to learn x_i–specific distribution of the vertical mixtures is more of a fine tuning which may improve the model performance once the essential structure has been identified. An intermediate choice which worked well in our experiments is to set $\tau_{\nu|\alpha,i} = \tau_{\nu|\alpha}$, i.e., to introduce priors which are shared by all objects x_i belonging to the same cluster $\mathcal{C}_\alpha$. The simplified E–step has the advantage that $\langle V_{r\nu|\alpha}\rangle$ does not depend on $x_{i(r)}$. Moreover it reduces the model complexity and may prevent overfitting. In the simplified model, instead of computing posteriors for all L observations, it suffices to compute posterior probabilities for all M $\mathcal{Y}$–objects. This can result in a significant acceleration, for example, in natural language modeling where M would be the size of the vocabulary as compared to the size R of the corpus of word occurrences which typically differ by several orders of magnitude. For $\tau_{\nu|\alpha} = const.$ an additional speed–up results from a simplified propagation of posteriors in the tree. In our simulation we have thus pursued a three–stage strategy, where the degrees of freedom are incrementally increased.

Like for the ACM it is straightforward to check that the predictive probabilities for $s = (x_i, y_j, L+1)$ are given by

$$P(s|\mathcal{S},\theta) = p_i \sum_\nu p_{\nu|i}\, q_{j|\nu},\; p_{\nu|i} \equiv \sum_\alpha \langle I_{i\alpha}\rangle \tau_{\nu|\alpha,i}\,. \quad (59)$$

4.3 Hierarchies and Abstraction

With other hierarchical mixture models proposed for supervised [3, 29] and unsupervised learning [37] the HACM shares the organization of clusters in a tree structure. It extracts hierarchical relations between clusters, i.e., it breaks the permutation–symmetry of the cluster labeling. Even more important, however, it is capable to perform *statistical abstraction*. Observations which are common to all clusters in the subtree rooted at an inner node are preferably captured at the level of generality represented by that node. Observations being highly specific are 'explained' on the terminal level. Therefore inner nodes do not represent a coarser view on the data which could be obtained, e.g., by a weighted combination of the distributions at successor nodes and as might be expected for a hierarchical model.[9] The vertical mixtures perform a *specialization* in terms of the level of generality most adequate for each of the observations. Thus the HACM incorporates a novel notion of hierarchical modeling, which differs from multiresolution approaches, but also from other hierarchical concepts of unsupervised learning (e.g. [11]). It offers several new possibilities in data analysis and information retrieval tasks like extracting resolution–dependent meaningful keywords for subcollection of documents and gives a satisfying solution to the problem of cluster *summarization* (cf. [8]) since it explicitly finds the most characteristic terms for each (super-)cluster of documents.

There are several additional problems which have to be solved to arrive at a complete algorithm for the HACM. The most important concerns the specification of a procedure to obtain the tree–topology $\mathcal{T}$. Before explaining our heuristic we have to introduce the important concept of *annealing*.

5 Improved EM Variants

5.1 Annealed EM

So far we have mainly focused on the modeling problem of defining mixture models for COD. The standard model fitting procedure has been the EM algorithm and its hard clustering variants. We now discuss two important problems which naturally occur in this context. The first problem is to avoid unfavorable local maxima of the log–likelihood. The second, even more important problem is to avoid overfitting, i.e., maximize the performance on unseen future data. The framework which allows us to improve the presented EM procedures in both aspects is known as *deterministic annealing*.

Deterministic annealing has been applied to many clustering problems, including vectorial clustering [49, 50, 5], pairwise clustering [23], and in the context of COD for distributional clustering [43]. The key idea is to introduce a temperature parameter T and to replace the minimization of a combinatorial objective function by a substitute known as the *free energy*. Details on this topic are given in Appendix A. Here, we present annealing methods without reference to statistical physics. Consider therefore the general case of maximum likelihood estimation by the EM algorithm. The E–step by definition computes a posterior average of the complete data log–likelihood which is maximized in the M–step. The *annealed* E–step at temperature T performs this average w.r.t. a distribution which is obtained by generalizing Bayes' formula such that the likelihood contribution is taken to the power of $1/T$. For $T > 1$ this amounts

[9] In fact, it is also important to develop hierarchical generalizations of the kind. However, we focus on the HACM which is more specific to COD.

to increasing the effect of the prior which in turn will result in a larger entropy of the (annealed) posteriors. For example, in the case of the ACM the annealed E–step generalizing (21) is given by[10]

$$\langle I_{i\alpha}\rangle^{(t)} = \frac{\hat{\rho}_\alpha^{(t)} \exp\left(-\frac{n_i}{T} D\left[n_{j|i}|\hat{q}_{j|\alpha}^{(t)}\right]\right)}{\sum_{\nu=1}^K \hat{\rho}_\nu^{(t)} \exp\left(-\frac{n_i}{T} D\left[n_{j|i}|\hat{q}_{j|\nu}^{(t)}\right]\right)} \quad . \tag{60}$$

For (hard) clustering applications deterministic annealing is utilized in its usual $T \to 0$ limit. Although there is no guarantee that deterministic annealing in general finds the global minimum, many independent empirical studies indicate that the typical solutions obtained are often significantly better than the corresponding 'unannealed' optimization. This is due to the fact that annealing is a *homotopy method*, where the (expected) likelihood as a cost function is smoothed for large T and is recovered in the limit $T \to 1$.

In addition, for fixed $T > 1$ the *annealed* E–step performs a regularization based on entropy, because the posterior probabilities minimize the generalized free energy at $T = 1$ which balances expected costs and (relative) entropy [38] (cf. Appendix A). This is the reason why annealed EM not only reduces the sensitivity to local minima but also controls the effective model complexity. It thereby has the potential to improve the generalization for otherwise overfitting models. The advantages of deterministic annealing are investigated experimentally in Section 6.

In addition, the annealed EM algorithm offers a way to generate tree topologies. As is known from adaptive vector quantization [49], starting at a high value of T and successively lowering T leads through a sequence of phase transitions. At each phase transition the effective number of distinguishable clusters grows until some maximal number is reached or the annealing is stopped. This suggests a heuristic procedure where we start with a single cluster and recursively split clusters. In the course of the annealing one keeps track of the splits and uses the 'phase diagram' as a tree topology $\mathcal{T}$. Note that merely the tree topology is successively grown, while the data partition obtained at a specific temperature is regrouped and may drastically change during the annealing process.

To summarize, annealed EM solves three problems at once:

1. It avoids unfavorable local minima by applying a temperature based continuation method, as the modified likelihood becomes convex at high temperature,
2. it avoids overfitting by discounting the likelihood contribution in the E–step
3. it offers a physically motivated heuristic to produce a meaningful tree topology (for the HACM).

In all experiments and for all statistical models we have therefore utilized the annealed variant of EM.

5.2 Predictive EM

Another modification of the E–step to improve generalization is worth considering. For notational simplicity focus on the E–step in the SMM as given by (6). If the parameters are eliminated by substituting them with their current M–step estimators in terms of posteriors as given by (7-9), we arrive at

$$\langle R_{r\alpha}\rangle^{(t+1)} \propto \frac{\left(\sum_{i(u)=i}\langle R_{u\alpha}\rangle^{(t)}\right)\left(\sum_{j(u)=j}\langle R_{u\alpha}\rangle^{(t)}\right)}{L\sum_u \langle R_{u\alpha}\rangle^{(t)}} \tag{61}$$

where $i = i(r)$ and $j = j(r)$. Eq. (61) reveals that in estimating $\langle R_{r\alpha}\rangle$ in the E–step its old estimator actually appear on the right hand side for $u = r$. This has the effect to systematically overestimate high posterior probabilities while small posteriors are underestimated. This positive feedback on the posteriors may lead to substantial overfitting phenomena. To illustrate this problem consider the extreme case, where $n_i = 1$, e.g., $(x_i, y_l, r) \in \mathcal{S}$ is the only observation with x_i. Then the stationary condition for the E–step is given by

$$\langle R_{r\alpha}\rangle = \frac{\langle R_{r\alpha}\rangle q_{j|\alpha}}{\sum_{\alpha'}\langle R_{r\alpha'}\rangle q_{j|\alpha'}} \tag{62}$$

and hence $q_{j|\alpha} = \sum_{\alpha'}\langle R_{r\alpha'}\rangle q_{j|\alpha'}$ whenever $\langle R_{i\alpha}\rangle > 0$ which is only fulfilled if $\langle R_{r\alpha}\rangle = 1$ (being a stable solution for $\alpha = \arg\max_{\alpha'} q_{j|\alpha'}$). For sparse data the diagonal contribution can thus be dominant and the positive feedback bears the risk of overfitting.

In order to overcome these problems we propose a variant of EM which we refer to as *predictive EM*. The only modification is to exclude the r–th observation in recalculating posteriors $\langle R_{r\alpha}\rangle$. The class membership of the r-th observation is predicted based on the remaining samples.[11] For the SMM this implies that diagonal contributions are excluded in (61). It is obvious that the proposed correction does only have a minor influence on the computational complexity of the fitting procedure. Despite the heuristic flavor caused by modifying an algorithmic step, the predictive EM can be motivated from strict optimization principles. Further details and convergence considerations are given in Appendix B.

In conclusion we would like to stress the fact, that although the positive feedback occurs in other EM algorithms, it is most severe for COD models, because of the inherent sparseness problem. Furthermore, other error measures like the squared error between a data vector and a cluster centroid are far less sensitive to this type of problems then is the cross entropy involving logarithms of small probabilities.

[10] Eq. (60) differs from the original formula by Pereira et al. [43] in that it scales the temperature with the frequency n_i and includes the mixing proportions. As pointed out before this is naturally obtained in the ML framework, while in the distributional clustering cost function (25) the weights n_i are not considered. Canceling these weights may be a reasonable approach to limit the effect of frequently observed 'objects' x_i on the organization of clusters. From a statistical viewpoint, however, the latter is implausible, because more observations should automatically sharpen the posterior distributions at a given temperature level T.

[11] This may result in undefined posterior probabilities due to zeros, as in the above example. In this case we assume the posterior to be uniform.

5.3 Accelerated EM

EM algorithms have important advantages over gradient–based methods, however for many problems the convergence speed of EM may restrict its applicability to large data sets. A simple way to accelerate EM algorithms is by *overrelaxation* in the M–step. This has been discussed early in the context of mixture models [44, 45] and was 'rediscovered' more recently under the title of $EM(\eta)$ in [1]. We found this method useful in accelerating the fitting procedure for all discussed models. Essentially the estimator for a generic parameter θ in the M–step is modified by

$$\hat{\theta}^{(t+1)} = (1-\eta)\hat{\theta}^{(t)} + \eta\overline{\theta}^{(t+1)}, \tag{63}$$

where $\overline{\theta}^{(t+1)}$ is the M–step estimate, i.e., $\eta = 1$ is the usual M-step. Choosing $1 < \eta < 2$ still guarantees convergence, and typically $\eta \approx 1.8$ has been found to be a good choice to speed up convergence. In case that a constraint is violated after performing an overrelaxed M–step, the parameter set is projected back on the admissible parameter space. For an overview on more elaborated acceleration methods for EM we refer to [36].

5.4 Multiscale Optimization

Multiscale optimization [21, 46] is an approach for accelerating clustering algorithms whenever a topological structure exists on the object space(s). In image segmentation, for example, it is a natural assumption that adjacent image sites belong with high probability to the same cluster or image segment. This fact can be exploited to significantly accelerate the estimation process by maximizing over a suitable nested sequence of variable subspaces in a coarse–to–fine manner. This is achieved by temporarily tying adjacent sites in a joint assignment variable. For notational convenience we again restrict the presentation to the ACM, while extensions to the SCM and both probabilistic variants are straightforward.[12]

More formally a coarsening hierarchy for $\mathcal{X}$ is given by a nested sequence of equivalence relations $\mathcal{M}^{(l)}$ over $\mathcal{X}$, where $\mathcal{M}^{(l)} \subset \mathcal{M}^{(l+1)}$ and $\mathcal{M}^{(0)} = \{(x_i, x_i) : x_i \in \mathcal{X}\}$. In the context of image analysis these equivalence relations typically correspond to multi–resolution pixel grids obtained by subsampling. The log–likelihood in (21) is minimized at coarsening level l by imposing constraints of the form $I_{i\alpha} = I_{j\alpha}$ whenever $(x_i, x_j) \in \mathcal{M}^{(l)}$. For all models under consideration this effectively amounts to reducing the number of indicator functions to one set of variables for each equivalence class in $\mathcal{M}^{(l)}$, while preserving the functional form of the likelihood, thus enabling highly accelerated optimization at coarser levels. Once the maximization procedure at a resolution level l is converged, the optimization proceeds at the next level $l-1$ by prolongating the found solution in $\mathcal{M}^{(l)}$ to the subset defined by $\mathcal{M}^{(l-1)}$, thus initializing the optimization at level $l-1$ with the solution at level l. For the probabilistic version multiscale optimization amounts to modifying the E–step of the EM algorithm by computing posteriors for the reduced sets of indicator functions.

[12] Multiscale optimization in its current form is not applicable to the SMM/PMM.

	SMM		ACM		HACM		SCM	
K	β	$\mathcal{P}$	β	$\mathcal{P}$	β	$\mathcal{P}$	β	$\mathcal{P}$
	CRAN							
1	-	685	-	-	-	-	-	-
8	0.88	482	0.09	527	0.18	511	0.67	615
16	0.85	431	0.07	482	0.14	471	0.60	543
32	0.83	386	0.07	**452**	0.12	438	0.53	506
64	0.79	360	0.06	527	0.11	422	0.48	477
128	0.78	**353**	0.04	663	0.10	**410**	0.45	**462**
	PENN							
1	-	639	-	-	-	-	-	-
8	0.73	312	0.08	352	0.13	322	0.55	394
16	0.72	255	0.07	302	0.10	268	0.51	335
32	0.71	205	0.07	254	0.08	226	0.46	286
64	0.69	182	0.07	**223**	0.07	204	0.44	272
128	0.68	**166**	0.06	231	0.06	**179**	0.40	**241**

Table 2: Perplexity results for different models (SMM, ACM, HACM, SCM) on two data sets (CRAN: predicting words conditioned on documents, PENN: predicting nouns conditioned on adjectives) based on ten-fold cross validation ($K^{\mathcal{X}} = K^{\mathcal{Y}} = K$ for SCM).

We like to emphasize that in contrast to most multiresolution optimization schemes, multiscale optimization has the advantage to maximize the *original* log–likelihood at all resolution levels. It is only the set of hidden variables which is effectively reduced by imposing the constraints on the set of hidden variables $\mathcal{M}^{(l)}$. We applied multiscale optimization in all image analysis experiments resulting in typical accelerations by factors 10–100 compared to single–level optimization.

6 Results

6.1 Information Retrieval

Information retrieval in large databases is one of the key topics in *data mining*. The problem is most severe in cases where the query cannot be formulated precisely, e.g., in natural language interfaces for documents or in image databases. Typically, one would like to obtain those entries which best match a given query according to some similarity measure. Yet, it is often difficult to reliably estimate similarities, because the query may not contain enough information, e.g., not all possibly relevant keywords might occur in a query for documents. Therefore, one often applies the *cluster hypothesis* [62]: if an entry is relevant to a query, similar entries may also be relevant to the query although they may not possess a high similarity to the query itself due to the small number of keywords. Clustering thus provides a way of pre-structuring a database for the purpose of improved information retrieval, cf. [63] for an overview. Both types of clustering approaches, for the set of documents as well as for the keywords, have been proposed in the literature. The most frequently used methods in this context are linkage algorithms (single linkage, complete linkage, Wards method, cf. [26]), or hybrid combinations of ag-

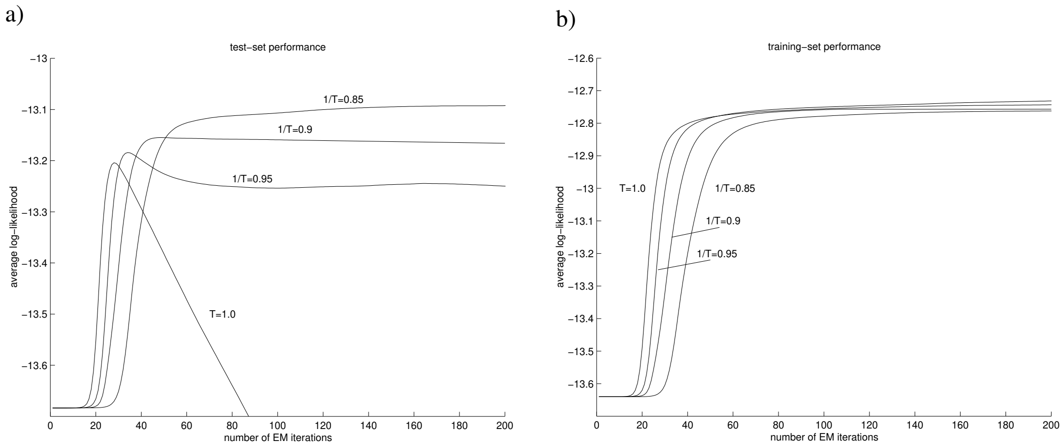

Figure 7: (a) Generalization performance and (b) training likelihood for the annealed EM at different temperatures on the Cranfield collection.

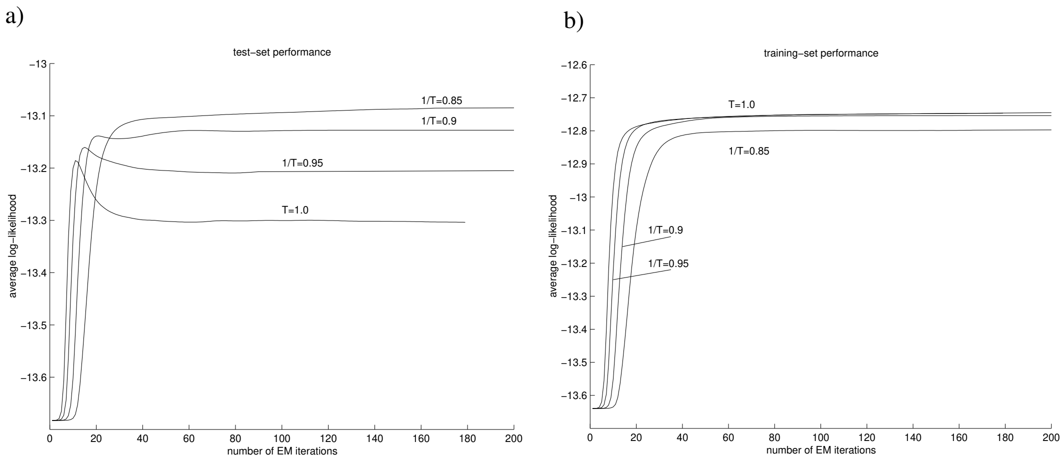

Figure 8: (a) Training likelihood and (b) generalization performance for the (annealed) predictive EM variant at different temperatures on the Cranfield collection.

glomerative and centroid–based methods [8] which have no probabilistic interpretation and have a number of other disadvantages. In contrast, COD mixture models provide a sound statistical basis and overcome the fundamental sparseness problem of proximity–based clustering. In particular the hierarchical clustering model (HACM) has many additional features which make it a suitable candidate for *interactive* or coarse-to-fine information retrieval.

We have performed experiments for information retrieval on different collections of abstracts. The index terms for each dataset have been automatically extracted from all documents with the help of a standard word stemmer. Following [62], a list of stop words has been utilized to exclude frequently used words. Words with few overall occurrences have also been eliminated. The documents are identified with the set of objects $\mathcal{X}$, while index terms correspond to $\mathcal{Y}$.

In the first series of experiments we have investigated how well the different models perform in predicting the occurrences of certain words in the context of a particular document. Therefore, the set of all word occurrences has been divided into a training and a test set. From a statistical point of view the canonical goodness-of-fit measure is the average log–likelihood on the test set. Yet, in the context of natural language processing it is more customary to utilize the perplexity $\mathcal{P}$ which is related to the average test set log–likelihood l by $\mathcal{P} = \exp(-l)$. Since we have used the annealed EM algorithm, a validation set was utilized to determine the optimal choice of the computational temperature. Comparative results for all discussed models[13] on the standard test collection Cranfield (CRAN, N=1400,M=1664,L=111803) are summarized in Table 2. For all experiments we have performed a ten-fold cross validation. The main conclusions are:

- The lowest perplexity is obtained with the SMM. The HACM performs better than the more constrained ACM and SCM. Hence, in terms of perplexity the linear mixture models should be pre-

[13] The performance of the PMM is comparable to the SMM and hence not displayed.

Figure 9: Upper part of a cluster hierarchy (6 levels) for the *CLUSTER* dataset generated by annealed EM for HACM. Each node is described by the index words y_j with the highest probability $q_{j|\nu}$.

ferred over the clustering models.

- The optimal temperature for the SMM is consistently below $T = 1$ which is the standard EM algorithm. For the clustering models the optimal generalization performance even requires a much higher temperature as expected.
- Temperature–based complexity control clearly does much better than restricting the number K of components. Even the SMM with $K = 8$ components suffers from overfitting at $T = 1$.

To stress the advantages of the proposed EM variants, we have investigated the effect of a temperature–based regularization in more detail. Fig. 7 shows log–likelihood curves for typical runs of the annealed EM algorithm at different temperatures.[14] The overfitting phenomenon is clearly visible, e.g., for the SMM at $T = 1$, where the test–set performance degrades after 30 iterations. Notice, that annealing performs much better than *early stopping*. A comparison of the predictive EM variant with the standard EM for the SMM is depicted in Fig. 8. This demonstrates that even the (presumably) slight modification to avoid positive feedback improves the test–set performance. The overrelaxed EM variant has also proven to be a valuable tool in our simulations with a typical acceleration by a factor $2 - 3$.

To facilitate the assessment of the extracted *structure* we have investigated a dataset of $N = 1584$ documents containing abstracts of papers with *clustering* as a title word (CLUSTER). This data is presumably more amenable to an interpretation by the reader than are the standard text collections. The top–levels of a cluster hierarchy generated by HACM are visualized in Fig. 9.

To demonstrate the ability of the HACM to identify abstraction levels in the hierarchy, we have visualized the distribution of the responsibility for observations involving the same index word y_j for some particularly interesting examples in Fig. 10. The first tree for the word 'cluster' shows that, as expected, the occurrences of 'cluster' in documents are explained to be a common feature of all documents, hence most of the occurrences are assigned to the root. The word 'decision' is found on a level 3 node, indicating that it is a typical word for all algorithmically oriented documents assigned to nodes in the subtree, but e.g. not for the left branch of papers from physics and astronomy. The index term 'robust' occurs in two different meanings: first, it has a highly spe-

[14]These simulations have been performed on COD from the CRAN collection. Qualitatively similar results have been obtained for all other data sets.

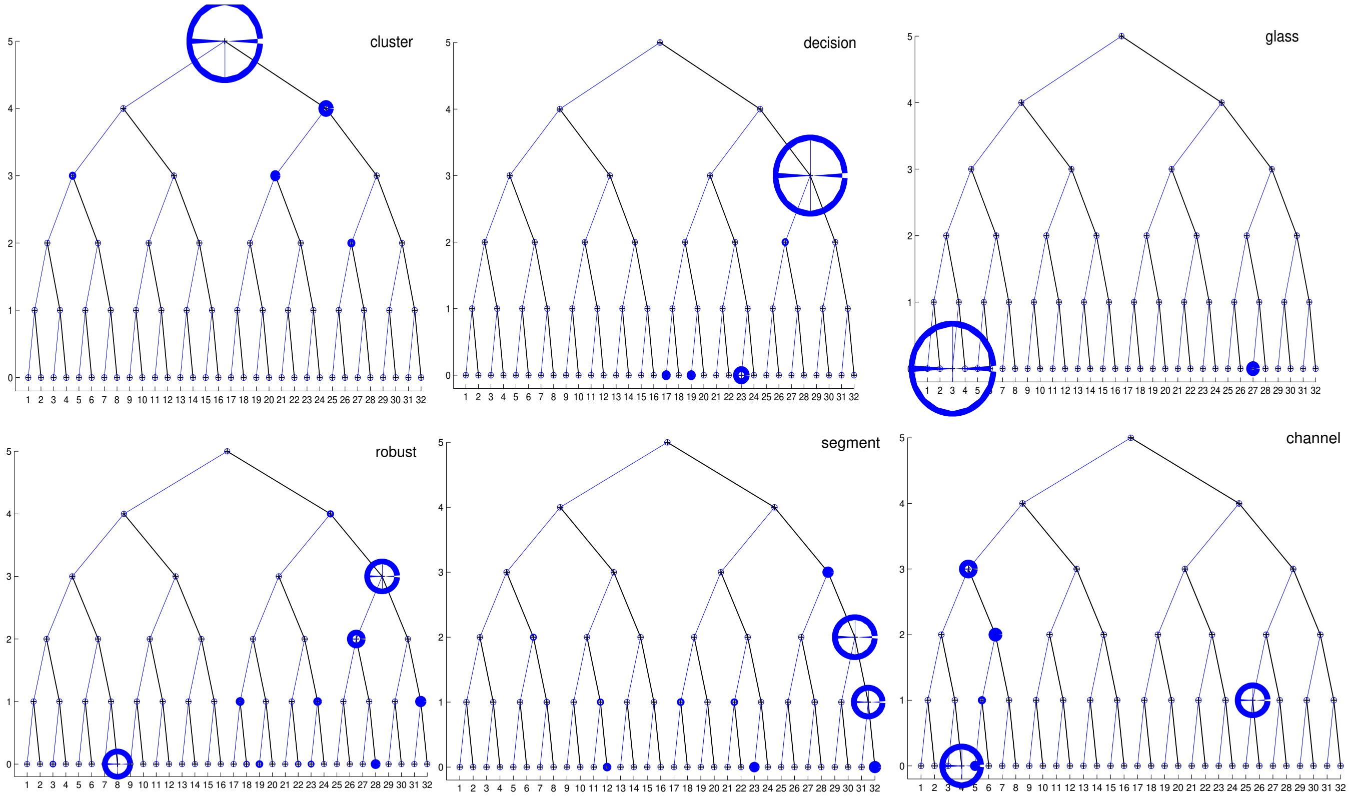

Figure 10: Exemplary relative word distributions over nodes for the *CLUSTER* dataset for the keywords 'cluster', 'decision', 'glass', 'robust', 'segment', and 'channel'.

cific meaning in the context of stability analysis ('plane', 'perturb', 'eigenvalue', 'root', etc.) and a rather broad meaning in the sense of robust methods and algorithms. The word 'segment' occurs mainly in documents about computer vision and language processing, but it is used to a significant larger extend in the first field. 'glass' is a specific term in solid state physics, it thus is found on the lowest level of the hierarchy. 'channel' is again ambivalent, it is used in the context of physics as well as in communication theory. The bimodal distribution clearly captures this fact.

The same experiments have been carried out for a dataset of 1278 documents with abstracts from the journals *Neural Computation* and *Neural Networks* (NN). The solution of a HACM with $K = 32$ clusters is visualized in Fig. 11, each node is again described by the index words with the highest probability.

These examples are only spotlights, but they demonstrate that the hierarchical organization obtained by the HACM is able to extract interesting structure from co-occurrence data. A detailed investigation of its full potential in the context of information retrieval is beyond the scope of this paper and will be pursued in future work.

6.2 Computational Linguistics

In computational linguistics, the statistical analysis of word co-occurrences in lexical structures like adjective/noun or verb/direct object has recently received a considerable degree of attention [22, 43, 9, 10]. Potential applications of these methods are in word–sense disambiguation, a problem which occurs in different linguistic tasks ranging from parsing and tagging to machine translation.

The data we have utilized to test the different models consists of adjective–noun pairs extracted from a tagged version of the Penn Treebank corpus ($N = 6931$, $M = 4995$, $L = 55214$) and the LOB corpus ($N = 5548$, $M = 6275$, $L = 36723$)[15]. Performance results on the Penn dataset are reported in the second half of Table 2. The results are qualitatively very similar to the ones obtained on the CRAN document collection, although this application is quite different from the one in information retrieval.

A result for a simultaneous hard clustering of the LOB data with the SCM is reported in Fig. 12. The visualization of the $\pi_{\nu\mu}$ matrix reveals that many groups in either space are preferably combined with mainly one group in the complementary space. For example the adjective group 'holy', 'divine', 'human' has its occurrences almost exclusively with nouns from the cluster 'life', 'nature', 'being'. Some groups are very much indifferent with respect to the groups in the corresponding set, e.g., the adjective group headed by 'small', 'big', 'suitable'. '

[15] Singular and plural forms have been identified.

Level	Node 1	Node 2	Node 3	Node 4	Node 5	Node 6	Node 7	Node 8
Level5	network 0.102 neural 0.052 function 0.0314 result 0.0311 gener 0.028							
Level4	network 0.0955 algorithm 0.0656 learn 0.0608 train 0.0539 problem 0.0384	model 0.104 neuron 0.0438 process 0.029 author 0.0219 inform 0.0191						
Level3	function 0.0606 weight 0.0327 learn 0.0306 error 0.0276 exampl 0.0235	perform 0.0351 network 0.0279 process 0.0268 data 0.0262 model 0.0233	dynamic 0.0391 time 0.0327 model 0.0295 synaptic 0.0282 neuron 0.0254	learn 0.0608 algorithm 0.035 rule 0.0323 network 0.0263 map 0.0215				
Level2	net 0.0466 state 0.0253 neuron 0.0229 continu 0.0223 arbitrari 0.0202	method 0.052 error 0.0455 gener 0.035 learn 0.0301 perform 0.0292	classif 0.0472 train 0.036 target 0.0306 featur 0.0251 classifi 0.0244	control 0.064 neural 0.0546 design 0.0339 inform 0.0249 network 0.0192	neuron 0.0819 activ 0.0332 spike 0.0319 respons 0.0302 burst 0.0224	network 0.0944 neural 0.064 learn 0.0496 algorithm 0.0338 dynamic 0.0237	visual 0.0786 imag 0.0341 motion 0.0303 cell 0.0247 process 0.022	memori 0.0836 associ 0.0539 capac 0.0411 pattern 0.0399 learn 0.0282
Level1a	function 0.0724 number 0.0434 hidden 0.0361 machin 0.0332 bound 0.0303	algorithm 0.0705 backpropag 0.0367 minima 0.0272 updat 0.0185 rate 0.0179	imag 0.0675 charact 0.0343 classifi 0.0283 featur 0.0171 constraint 0.0161	model 0.0563 nonlinear 0.0318 statist 0.03 optim 0.0215 linear 0.0207	fire 0.0631 cell 0.0418 spike 0.0276 rate 0.0174 record 0.0157	control 0.0579 motor 0.0541 problem 0.0342 movem 0.0328 robot 0.0203	recept 0.032 field 0.027 orient 0.0258 map 0.0255 cortic 0.0222	pattern 0.0563 memori 0.0516 node 0.0337 paramet 0.0306 new 0.0187
Level1b	energi 0.033 delta 0.032 valu 0.0252 memori 0.0238 associ 0.0198	model 0.0521 regular 0.0472 basi 0.0355 term 0.0263 architectur 0.0217	fuzzi 0.0435 map 0.037 decision 0.0301 function 0.0243 boundari 0.0189	time 0.0418 robot 0.0224 domain 0.0223 level 0.0187 algorithm 0.0184	oscil 0.039 circuit 0.0257 local 0.0239 spatial 0.0215 electr 0.0204	inhibi 0.023 cell 0.0171 reinforc 0.0167 extern 0.0157 stochast 0.015	object 0.0761 featur 0.0407 recogni 0.0289 perform 0.0236 segment 0.0211	rule 0.0326 optim 0.0255 size 0.0207 hebbian 0.0204 averag 0.0165
Level0aa	approxim 0.13 continu 0.0227 given 0.0216 art 0.0209 univers 0.0176	quantiz 0.0248 competit 0.0214 order 0.0209 determinist 0.0198 optim 0.0188	neocognitron 0.0385 invari 0.0352 recogn 0.0254 rotat 0.0192 scell 0.0192	stabil 0.03 ann 0.0227 knowledg 0.0212 expert 0.0166 rule 0.0163	paramet 0.034 assembli 0.0165 synchroni 0.0163 steadystat 0.0146 optim 0.0146	coordin 0.0293 limb 0.025 object 0.0208 order 0.0191 target 0.0191	transform 0.0232 spacevari 0.0178 nonlinear 0.0152 term 0.0139 explor 0.0129	motion 0.0397 select 0.0264 flow 0.0148 strategi 0.0148 overlap 0.0148
Level0ab	threshold 0.0313 circuit 0.0275 node 0.027 theta 0.0258 transfer 0.0231	neuron 0.0272 process 0.0271 jacobian 0.017 fam 0.0153 joint 0.0153	filter 0.0302 local 0.0265 digit 0.0134 detect 0.0106 gabor 0.0104	minim 0.0179 cmac 0.0176 unsupervis 0.0153 compon 0.0142 bound 0.0128	pyramid 0.0303 rat 0.02 auditori 0.0178 threshold 0.0134 machin 0.0125	chaotic 0.0399 search 0.0227 solution 0.0209 trajectori 0.0192 loop 0.0189	map 0.0412 maxim 0.0185 retin 0.0143 kohonen 0.0134 format 0.0129	vector 0.0551 kei 0.0138 set 0.0136 tsub 0.0134 stabl 0.0131
Level0ba	sub 0.0772 alpha 0.0439 epsilon 0.037 xsub 0.033 matrix 0.0258	selec 0.0302 bayesian 0.0284 data 0.0237 seri 0.0143 gaussian 0.0142	artmap 0.0362 art 0.0276 node 0.022 cost 0.019 transform 0.0177	echo 0.0656 target 0.0314 bat 0.0246 locat 0.0225 sonar 0.0225	casup 0.0292 center 0.0257 ltp 0.0172 postsynapt 0.0172 conduct 0.0152	net 0.0268 oper 0.0212 respons 0.0201 class 0.0158 delai 0.0148	shape 0.0208 human 0.0189 enforc 0.0179 region 0.0166 oscil 0.0163	regular 0.0244 topologi 0.0184 estim 0.0176 give 0.0164 bag 0.0143
Level0bb	citi 0.0227 list 0.021 path 0.0175 effect 0.0161 tsp 0.0157	shown 0.0207 hybrid 0.0192 nonlinear 0.0176 smooth 0.0157 differenti 0.014	hint 0.0354 processor 0.0214 dfa 0.0161 selforgan 0.0158 determin 0.0157	hardwar 0.0226 sequenti 0.015 chang 0.0136 elem 0.0134 strategi 0.0131	phase 0.0328 oscillatori 0.0287 connec 0.0247 coupl 0.0147 retin 0.0138	nois 0.0319 inhibitori 0.0243 group 0.0204 analog 0.017 interneuron 0.0169	invari 0.0337 pattern 0.0259 line 0.0243 rotat 0.0225 transform 0.0186	retriev 0.0385 error 0.027 valu 0.0225 princip 0.0219 compon 0.0179

Figure 11: Upper part of a cluster hierarchy (6 levels) for the NN dataset generated by annealed EM for HACM. Each node is described by the index words y_j with the highest probability $q_{j|\nu}$.

6.3 Unsupervised Texture Segmentation

The unsupervised segmentation of textured images is one of the most challenging and still only partially solved problems in low level computer vision. Numerous approaches to texture segmentation have been proposed over the past decades, most of which obey a two–stage scheme:

1. A *modeling stage*: characteristic features are extracted from the textured input image, which range from spatial frequencies [25, 24], MRF–models [33, 34], co-occurrence matrices [16] to fractal indices [6].
2. In the *clustering stage* features are grouped into homogeneous segments, where homogeneity of features has to be formalized by a mathematical notion of *similarity*.

Most widely, features are interpreted as vectors in a Euclidean space [25, 33, 34, 65, 40, 6, 31] and a segmentation is obtained by minimizing the K–means criterion, which sums over the square distances between feature vectors and their assigned, group–specific *prototype feature vectors*. K–means clustering can be understood as a statistical mixture model with isotropic Gaussian class distributions.

Occasionally, the grouping process has been based on *pairwise similarity* measurements between image sites, where similarity is measured by a non–parametric statistical test applied to the feature distribution of a surrounding neighborhood [16, 39, 24]. Agglomerative techniques [39] and, more rigorously, optimization approaches [24, 57] have been developed and applied for the grouping of similarity data in the texture segmentation context. Pairwise similarity clustering thus provides an indirect way to group (discrete) feature distributions without reducing information in a distribution to their mean.

COD mixture models, especially the ACM model, formalize the grouping of feature distribution in a more direct manner. In contrast to pairwise similarity clustering, they offer a sound generative model for texture class description which can be utilized in subsequent processing stages like edge localization [56]. Furthermore, there is no need to compute a large matrix of pairwise similarity scores between image sites, which greatly reduces the overall processing time and memory requirements. Compared to the mixture of Gaussian model, ACM provides significantly more flexibility in distribution modeling. Especially in the texture segmentation application class features often exhibit a non–Gaussian, e.g., multi–

1:	2:	3:	4:	5:	6:	7:	8:
voice	work	word	power	behaviour	place	life	worker
face	idea	line	war	floor	field	nature	demand
body	book	impression	size	trip	person	being	informant
head	job	statement	test	leave	section	spirit	acid
light	manner	style	weapon	companion	study	name	glance
9:	**10:**	**11:**	**12:**	**13:**	**14:**	**15:**	**16:**
party	society	experience	position	expression	world	water	attitude
country	family	reason	course	desire	town	eye	thought
system	court	result	unit	atmosphere	method	hair	tradition
sense	child	knowledge	trade	space	movement	fish	programme
market	community	feature	control	note	view	milk	division
17:	**18:**	**19:**	**20:**	**21:**	**22:**	**23:**	**24:**
deal	land	way	authority	figure	soil	number	change
friend	music	thing	government	piece	spot	amount	term
example	metal	hand	paper	shape	iron	scale	property
night	blow	area	council	sound	ship	report	action
bird	stone	side	committee	wind	drink	majority	effort
25:	**26:**	**27:**	**28:**	**29:**	**30:**	**31:**	**32:**
man	value	smile	effect	interest	service	part	feeling
time	level	frame	quality	condition	school	form	love
people	rate	crowd	solution	development	purpose	problem	letter
year	price	baby	picture	meeting	class	case	silence
day	cost	phrase	colour	activity	agreement	type	heart

1:	2:	3:	4:	5:	6:	7:	8:
complete	good	strange	sudden	true	whole	old	high
natural	great	thin	sharp	hard	right	young	final
official	fine	soft	powerful	excellent	wide	little	low
separate	bad	bitter	friendly	ancient	left	early	ordinary
growing	close	plain	curious	famous	opposite	short	average
9:	**10:**	**11:**	**12:**	**13:**	**14:**	**15:**	**16:**
nuclear	human	special	large	common	small	tiny	empty
western	divine	important	certain	foreign	big	perfect	brilliant
apparent	holy	particular	considerable	technical	suitable	native	endless
atomic	peaceful	various	total	open	given	distant	sacred
potential	continued	basic	annual	medical	light	violent	clever
17:	**18:**	**19:**	**20:**	**21:**	**22:**	**23:**	**24:**
slight	similar	electric	double	national	different	new	royal
vertical	single	massive	huge	general	real	present	extended
severe	strong	gentle	tremendous	public	original	modern	closed
awful	regular	cool	occasional	social	possible	serious	heavenly
theatrical	obvious	cheap	firm	private	simple	existing	reciprocal
25:	**26:**	**27:**	**28:**	**29:**	**30:**	**31:**	**32:**
long	red	heavy	full	deep	local	interesting	political
poor	cold	literary	military	daily	white	usual	personal
happy	black	familiar	free	quick	central	successful	economic
golden	dark	pure	professional	mixed	legal	remarkable	individual
beautiful	hot	solid	musical	square	eastern	historical	industrial

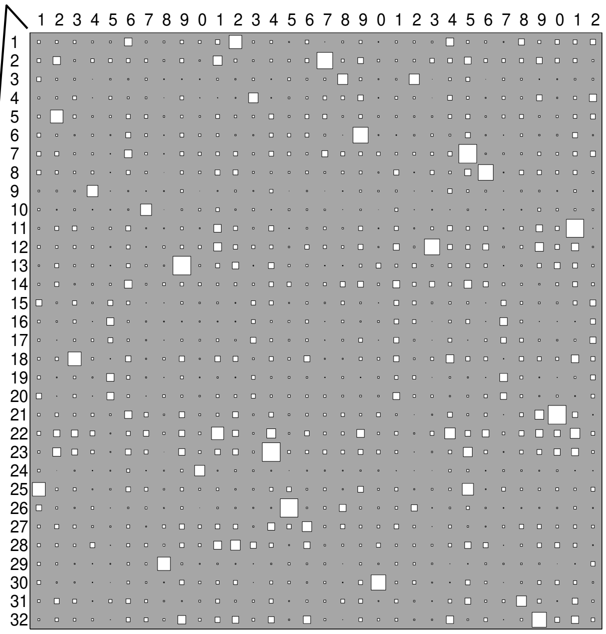

Figure 12: Clustering of LOB using the SCM ($K^{\mathcal{X}} = K^{\mathcal{Y}} = 32$) with a visualization of the $\pi_{\nu\mu}$ matrix and a characterization of clusters by their most probable words.

modal distribution, which is the main reason for the success of pairwise similarity clustering approaches and the ACM compared to standard K–means.

The ACM is applicable to any feature extraction process. As an example we exploit a Gabor filter image representation. More specifically, in all experiments we used the modulus of a bank of Gabor filters with 4 orientations at 3 scales with an octave spacing, resulting in a 12 dimensional feature vector associated to each image site. In our experiments the features were discretized separately for each dimension using 40 bins. Statistical independence of the feature channels has been assumed for simplicity.

The feature generation process is then modeled as a co-occurrence of an image site $x_i \in \mathcal{X}$ and a measured Gabor feature occurrence in one channel $y_j = f_{k_j}^{r_j} \in \mathcal{Y}$, where r_j denotes the Gabor channel and k_j denotes the index of the (discretized) feature. The sample set $\mathcal{S}_i$ for site x_i consists of all Gabor responses in a window centered at x_i, where the size of the window is chosen proportional to the filter scale [25]. Hence each image location x_i is effectively characterized by 12 one-dimensional histograms over Gabor coefficients.

We have applied the ACM–based texture segmentation algorithm to a collection of textured images. Fig. 13 shows exemplary results for images which were randomly generated from the Brodatz texture collection of micro–textures. Fig. 14 shows similar results for mixtures of aerial images. A detailed benchmark study of this novel segmentation algorithm including comparisons with state-of-the-art techniques will appear in a forthcoming paper.

7 Conclusion

As the main contribution of this paper a novel class of statistical models for the analysis of co-occurrence data has been proposed and evaluated. We have introduced and discussed several different models, not only by enumerating them as alternative approaches, but by distinguishing them from a systematic point of view. The criterion to differentiate between these models is the way hidden variables are introduced, which effectively imposes constraints on the component distributions of the mixture. Several recently proposed statistical models have turned out to be special cases. All models have a sound statistical foundation in that they define a generative distribution, and all of them can be fitted by an (approximate) EM algorithm.

Which of these models is the method of choice for a given problem crucially depends on the modeling goal. As we have argued, it is often required to detect groups structure or hierarchical representations. In these situations one may be willing to sacrifice some precision in terms of statistical accuracy (i.e., perplexity reduction) to extract the structure of interest. Within the proposed framework models have been derived to extract group structure on either one or both object spaces and to model hierarchical dependencies of clusters. We strongly believe the proposed framework is flexible enough to be adapted to many different tasks. The generality of the developed methods has been stressed by discussing their benefits in the context of a broad range of potential applications.

In addition to the modeling problem we have also addressed computational issues, in particular focusing on improved variants of the basic EM algorithm. Most im-

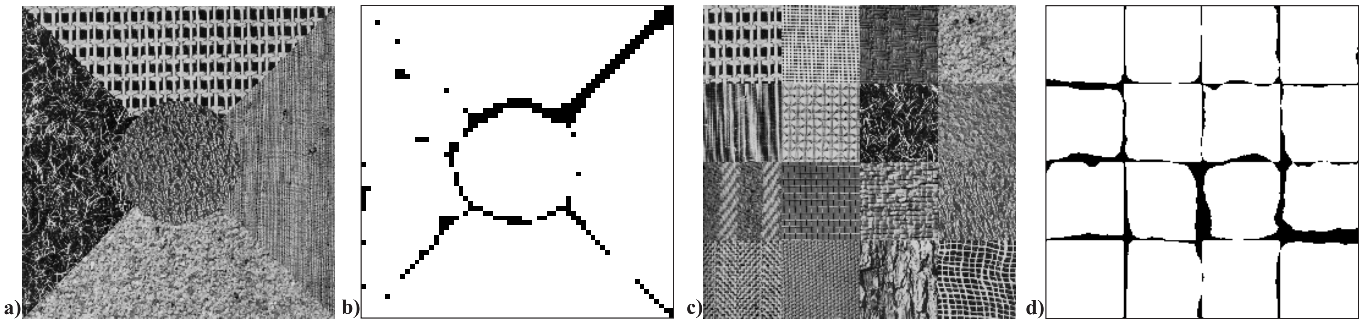

Figure 13: (a), (c) Two mixture images containing 5 textures each. (b), (d) The image segmentation obtained based on the ACM.

portantly, our experiments underline the possible advantages of the annealed version of EM, which is a fruitful combination of ideas and methods from statistics and statistical physics.

Acknowledgment

The authors wish to thank Michael Jordan, Peter Dayan, Tali Tishby, and Joachim Buhmann for helpful comments and suggestions. The authors are grateful to Carl de Marcken and Joshua Goodman for sharing their expertise and data in natural language processing as well as to J.M.H. du Buf for providing the image data depicted in Fig. 14.

Appendix A

First, we establish an important relationship between the log–likelihood and a quantity known as *free energy* in statistical physics [38]. Consider the data log–likelihood $\mathcal{L} = \log P(\mathcal{S}|\theta, R)$ as a function of the discrete hidden states R over $\mathcal{R}$ for fixed parameters, and let $\mathcal{H}(R; \mathcal{S}, \theta) = -\mathcal{L}$ define a cost function on the hidden variable space. Minimizing $\mathbf{E}\left[\mathcal{H}(R; \mathcal{S}, \theta)\right]$ w.r.t. probability distributions over $\mathcal{R}$ subject to a constraint on the entropy yields a quantity which is known as the *free energy* in statistical physics. This is generalized to non–uniform priors by fixing the relative entropy with respect to a prior distribution $\pi(R)$. Introducing a Lagrange parameter T we arrive at the following objective function for probability distributions over the discrete space $\mathcal{R}$

$$\mathcal{F}_T(P|\mathcal{S}, \theta, \pi) = \mathbf{E}_P\left[\mathcal{H}(I)\right] + T\mathbf{E}_P\left[\log\frac{P(R)}{\pi(R)}\right]. \quad (64)$$

The solution of the minimization problem associated with the generalized free energy in (64) is the (tilted) Gibbs distribution

$$\begin{aligned} P(R|\mathcal{S}, \theta, \pi) &\propto \pi(R) \exp\left[-\frac{1}{T}\mathcal{H}(R; \mathcal{S}, \theta)\right] \\ &= \pi(R)\left[P(\mathcal{S}|\theta, R)\right]^{\frac{1}{T}} \quad . \qquad (65) \end{aligned}$$

For $T = 1$ this is exactly the posterior probability of R. The posterior thus minimizes $\mathcal{F}_T$ at $T = 1$. The *annealed* EM algorithm is the generalization defined by an arbitrary choice of (the temperature) T [60]. In the E–step for $T > 1$ this amounts to discounting the likelihood as compared to the prior by taking it to the $1/T$–th power. The M–step performs a minimization over $\mathbf{E}_P\left[\mathcal{H}(I; \mathcal{S}, \theta)\right]$ and therefore $\mathcal{F}_T(P|\mathcal{S}, \theta, \pi)$ with respect to θ for fixed P where P is a Gibbs distribution which does not necessarily correspond to the true posterior. Notice that convergence of annealed EM is guaranteed since $\mathcal{F}_T$ (but not necessarily the likelihood itself) is a Lyapunov function.

If an exact calculation of the posterior in the E–step is intractable, typically because of higher order correlations between the hidden variables as in the SCM, the optimization problem in (64) is restricted to factorial distributions $P(R|p_{r\alpha}) = \prod_r \prod_\alpha p_{r\alpha}^{R_{r\alpha}}$. We make use of the more suggestive notation $\langle R_{r\alpha}\rangle = p_{r\alpha}$ to stress that the variational parameters $p_{r\alpha}$ can actually be thought of as an approximation of the posterior marginals. This variational technique is known as mean-field approximation [42, 2] and has been successfully applied for optimization problems [61, 23], in computer vision [15, 67, 24], and for inference in graphical models [55].

In general, solutions of the *mean–field approximation* have to fulfill the stationary conditions

$$\langle R_{r\alpha}\rangle = \frac{1}{Z}\pi_{r\alpha} \exp\left[-\frac{1}{T}\langle\mathcal{H}(R; \mathcal{S}, \theta, R_{r\alpha} = 1)\rangle\right] \quad (66)$$

where expectations are taken with respect to $P(R|p_{r\alpha})$ [24]. Notice that expected costs appear in the exponent, however the expectation is taken with respect to all hidden variables except $R_{r\alpha}$ itself which is fixed. In order to obtain a convergent iteration procedure in the general case one has to replace the 'synchronous' E–step update by a 'sequential' update. In the SCM, the coupling between hidden variables is restricted to pairs of variables $I_{i\nu}$ and $J_{j\mu}$ with $n_{ij} > 0$ which allows us to recompute all posteriors for the I variables for given posteriors J in one sweep, and vice versa.

Appendix B

In order to preserve strict optimization principles, the predictive E–step variant has to be implemented slightly more careful than naively eliminating diagonal contributions. Inserting the corrected estimates of θ for given R into the complete data log–likelihood function (2) we

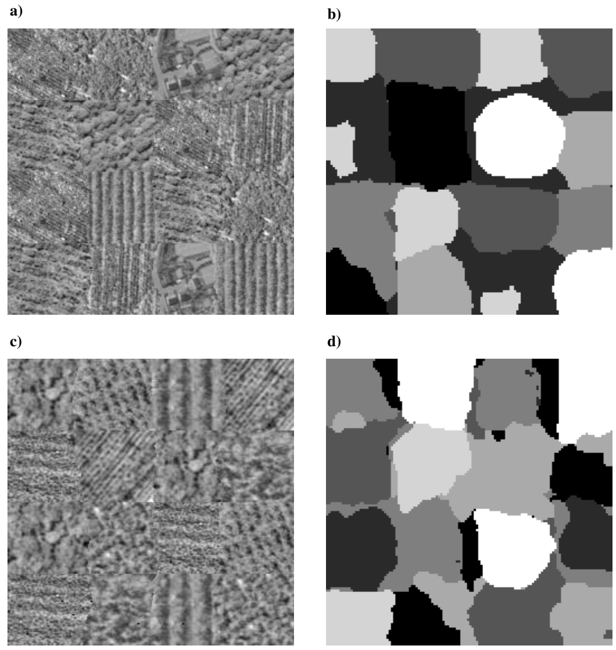

Figure 14: (a), (c) Two mixture images each containing 7 textures extracted from aerial images. (b), (d) The image segmentation obtained based on the ACM.

obtain the following cost function for R

$$\mathcal{H}(R;\mathcal{S}) = \sum_{r=1}^{L}\sum_{\alpha=1}^{K} R_{r\alpha} h_{r\alpha} \qquad (67)$$
$$h_{r\alpha} = \log \sum_{\substack{u\neq r\\ i(u)=i(r)}} R_{u\alpha} + \log \sum_{\substack{u\neq r\\ j(u)=j(r)}} R_{u\alpha} - \log \sum_{u\neq r} R_{u\alpha},$$

which we refer to as *predictive likelihood*. Minimizing the free energy corresponding to (67) in a mean-field approximation yields a direct contribution proportional to $\exp[-h_{r\alpha}/T]$ but also additional terms which result from the indirect effect $R_{r\alpha}$ has on $h_{s\alpha}$ for other variables $R_{s\alpha}$. We omit the details of the derivation which is purely technical. As a consequence one has to utilize sequential update to guarantee convergence of predictive EM. For reasons of efficiency we have ignored the indirect effects in the computation of the posterior probabilities in our experiments which empirically turned out to work well.

References

[1] E. Bauer, D. Koller, and Y. Singer. Update rules for parameter estimation in Bayesian networks. In *Proceedings of the Thirteenth Annual Conference on Uncertainty in Artifical Intelligence*, pages 3–13, 1997.

[2] G. Bilbro and W. Snyder. Mean field approximation minimizes relative entropy. *Journal of the Optical Society of America*, 8(2):290–294, 1991.

[3] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone. *Classification and Regression Trees.* Wadsworth Intern. Group, Belmont, California, 1984.

[4] P.F. Brown, P.V. deSouza, R.L. Mercer, V.J. Della Pietra, and J.C. Lai. Class-based n-gram models of natural language. *Computational Lingustics*, 18(4):467–479, 1992.

[5] J.M. Buhmann and H. Kühnel. Vector quantization with complexity costs. *IEEE Transactions on Information Theory*, 39(4):1133–1145, July 1993.

[6] B. Chaudhuri and N. Sarkar. Texture segmentation using fractal dimension. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(1):72–77, 1995.

[7] S. Chen and J. Goodman. An empirical study of smoothing techniques for language modeling. In *Proceedings of the 34th Annual Meeting of the ACL*, pages 310–318, 1996.

[8] D.R. Cutting, D.R. Karger, and J.O. Pedersen. Constant interaction-time scatter/gather browsing of very large document collections. In *Sixteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Pittsburgh, PA, USA*, pages 126–134, 1993.

[9] I. Dagan, L. Lee, and F.C.N. Pereira. Similarity-based estimation of word cooccurence probabilities. In *Proceedings of the Association for Computational Linguistics*, 1993.

[10] I. Dagan, L. Lee, and F.C.N. Pereira. Similarity-based methods for word sense disambiguation. In *Proceedings of the Association for Computational Linguistics*, 1997.

[11] P. Dayan, G.E. Hinton, R.M. Neal, and R.S. Zemel. The Helmholtz machine. *Neural Computation*, 7(5):889–904, 1995.

[12] S. Deerwester, S. T. Dumais, T. K. Landauer, G. W. Furnas, and R. A. Harshman. Indexing by latent semantics analysis. *Journal of the American Society for Information Science*, 41(6):391–407, 1991.

[13] 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.

[14] U. Essen and V. Steinbiss. Cooccurrence smoothing for stochastic language modeling. In *Proceedings of the IEEE nternational Conference on Acoustics, S peech, and Signal Processing*, pages 161–164, 1992.

[15] Davi Geiger and Federico Girosi. Coupled markov random fields and mean field theory. In *Advances in Neural Information Processing Systems 2*, pages 660–667, 1990.

[16] D. Geman, S. Geman, C. Graffigne, and P. Dong. Boundary detection by constrained optimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(7):609–628, 1990.

[17] W.R. Gilks, S. Richardson, and D.J. Spiegelhalter, editors. *Markov chain Monte Carlo in practice.* Chapman & Hall, 1997.

[18] I.J. Good. *The Estimation of Probabilities.* Research Monograph 30. MIT Press, Cambridge, MA, 1965.

[19] I.J. Good. The population frequencies of species and the estimation of population parameters. *Biometrika*, 40(3):237–264, 1991.

[20] A. Griffiths, H.C. Luckhurst, and P. Willett. Using interdocument similarity information in document retrieval systems. *Journal of the American Society for Information Science*, 37:3–11, 1986.

[21] F. Heitz, P. Perez, and P. Bouthemy. Multiscale minimization of global energy functions in some visual recovery problems. *CVGIP: Image Understanding*, 59(1):125–134, 1994.

[22] D. Hindle. Noun classification from predicate–argument structures. In *Proceedings of the Association for Computational Linguistics*, pages 268–275, 1990.

[23] T. Hofmann and J.M. Buhmann. Pairwise data clustering by deterministic annealing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(1), 1997.

[24] T. Hofmann, J. Puzicha, and J.M. Buhmann. Deterministic annealing for unsupervised texture segmentation. In *Proceedings of the International Workshop on Energy Minimization Methods in Computer Vision and Pattern Recognition*, volume 1223 of *Lecture Notes in Computer Science*, pages 213–228, May 1997.

[25] A. Jain and F. Farrokhnia. Unsupervised texture segmentation using Gabor filters. *Pattern Recognition*, 24(12):1167–1186, 1991.

[26] A.K. Jain and R.C. Dubes. *Algorithms for Clustering Data.* Prentice Hall, Englewood Cliffs, NJ 07632, 1988.

[27] F. Jelinek. The development of an experimental discrete dictation recogniser. *Proceedings of the IEEE*, 73(11), 1985.

[28] 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.

[29] M.I. Jordan and R.A. Jacobs. Hierarchical mixtures of experts and the EM algorithm. *Neural Computation*, 6(2):181–214, 1994.

[30] S.M. Katz. Estimation of probabilities for sparse data for the language model component of a speech recogniser. *ASSP*, 35(3):400–401, 1987.

[31] A. Laine and J. Fan. Frame representations for texture segmentation. *IEEE Transactions on Image Processing*, 5(5):771–779, 1996.

[32] S.L. Lauritzen, editor. *Graphical models.* Clarendon Press & Oxford University Press, 1996.

[33] B. Manjunath and R. Chellappa. Unsupervised texture segmentation using Markov random field models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13:478–482, 1991.

[34] J. Mao and A. Jain. Texture classification and segmentation using multiresolution simultaneous autoregressive models. *Pattern Recognition*, 25:173–188, 1992.

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

[36] G.J. McLachlan and T. Krishnan. *The EM Algorithm and Extensions.* Wiley, New York, 1997.

[37] D. Miller and K. Rose. Hierarchical, unsupervised learning with growing via phase transitions. *Neural Computation*, 8(8):425–450, 1996.

[38] R.M. Neal and G.E. Hinton. A new view of the EM algorithm that justifies incremental and other variants. *Biometrica*, 1993. submitted.

[39] T. Ojala and M. Pietikäinen. Unsupervised texture segmentation using feature distributions. Technical Report CAR-TR-837, Center for Automation Research, University of Maryland, 1996.

[40] D. Panjwani and G. Healey. Markov random field models for unsupervised segmentation of textured color images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(10):939–954, 1995.

[41] J. Pearl. *Probabilistic reasoning in intelligent systems: networks of plausible inference.* Morgan Kaufmann Publishers, San Mateo, CA, 1988.

[42] R. E. Peierls. On a minimum property of the free energy. *Physical Review*, 54:918, 1938.

[43] 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.

[44] B.C. Peters and H.F. Walker. An iterative procedure for obtaining maximum–likelihood estimates of the parameters of a mixture of normal distribution. *SIAM Journal of Applied Mathematics*, 35:362–378, 1978.

[45] B.C. Peters and H.F. Walker. The numerical evaluation of the maximum–likelihood estimates of a subset of mixture proportions. *SIAM Journal of Applied Mathematics*, 35:447–452, 1978.

[46] J. Puzicha and J. Buhmann. Multiscale annealing for real–time unsupervised texture segmentation. Technical Report IAI–97–4, Institut für Informatik III, 1997.

[47] J. Rissanen. Universal coding, information, prediction and estimation. *IEEE Transactions on Information Theory*, 30(4):629–636, 1984.

[48] J. Rissanen. Stochastic complexity and modeling. *The Annals of Statistics*, 14(3):1080–1100, 1986.

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

[50] K. Rose, E. Gurewitz, and G. Fox. Vector quantization by deterministic annealing. *IEEE Transactions on Information Theory*, 38(4):1249–1257, 1992.

[51] G. Salton. Experiments in automatic thesaurus construction for information retrieval. In *Proceedings IFIP Congress, TA-2*, pages 43–49, 1971.

[52] G. Salton. Developments in automatic text retrieval. *Science*, 253:974–979, 1991.

[53] G. Salton and C. Buckley. Term weighting approaches in automatic text retrieval. *Information Processing and Management*, 24(5):513–523, 1988.

[54] L. Saul and F.C.N. Pereira. Aggregate and mixed–order Markov models for statistical language processing. In *Proceedings of the Second Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics*, 1997.

[55] L. K. Saul, T. Jaakkola, and M.I. Jordan. Mean field theory for simoid belief networks. *Journal of Artificial Intelligence Research*, 4:61–76, 1996.

[56] P. Schroeter and J. Bigun. Hierarchical image segmentation by multi-dimensional clustering and orientation-adaptive boundary refinement. *Pattern Recognition*, 28(5):695–709, 1995.

[57] J. Shi and J. Malik. Normalized cuts and image segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'97)*, pages 731–737, 1997.

[58] K. Sparck Jones. *Automatic Keyword Classification for Information Retrieval.* Butterworths, London, 1971.

[59] D.M. Titterington, A.F.M. Smith, and U.E. Makov. *Statistical Analysis of Finite Mixture Distributions.* John Wiley & Sons, 1985.

[60] N. Ueda and R. Nakano. Deterministic annealing variants of EM. In *Advances in Neural Information Processing Systems 7*, pages 545–52, 1995.

[61] D. van den Bout and T. Miller. Graph partitioning using annealed neural networks. *IEEE Transactions on Neural Networks*, 1(2):192–203, 1990.

[62] C.J. van Rijsbergen. *Information retrieval.* Butterworths, London Boston, 1979.

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

[64] I.H. Witten and T.C. Bell. The zero–frequency problem: Estimating the probabilities of novel events in adaptive text compression. *IEEE Transactions on Information Theory*, 37(4):1085–1094, 1991.

[65] C.S. Won and H. Derin. Unsupervised segmentation of noisy and textured images using Markov random fields. *CVGIP: Graphical Models and Image Processing*, 54(4):308–328, July 1992.

[66] Y. Yang and J. Willbur. Using corpus statistics to remove redundant words in text categorization. *Journal of the American Society for Information Science*, 47(5):357–369, 1996.

[67] J. Zhang. The mean–field theory in EM procedures for blind Markov random fields. *IEEE Transactions on Image Processing*, 2(1):27–40, 1993.