Estimation of Mixture Models using Co-EM

Steffen Bickel BICKEL@INFORMATIK.HU-BERLIN.DE
Tobias Scheffer SCHEFFER@INFORMATIK.HU-BERLIN.DE
Humboldt-Universität zu Berlin, Department of Computer Science, Unter den Linden 6, 10099 Berlin, Germany

Abstract

We study estimation of mixture models for problems in which multiple views of the instances are available. Examples of this setting include clustering web pages or research papers that have intrinsic (text) and extrinsic (references) attributes. Our optimization criterion quantifies the likelihood and the consensus among models in the individual views; maximizing this consensus minimizes a bound on the risk of assigning an instance to an incorrect mixture component. We derive an algorithm that maximizes this criterion. Empirically, we observe that the resulting clustering method incurs a lower cluster entropy than regular EM for web pages, research papers, and many text collections.

1. Introduction

In many application domains, instances can be represented in two or more distinct, redundant views. For instance, web pages can be represented by their text, or by the anchor text of inbound hyperlinks ("miserable failure"), and research papers can be represented by their references from and to other papers, in addition to their content. In this case, multi-view methods such as co-training (Blum & Mitchell, 1998) can learn two initially independent hypotheses. These hypotheses bootstrap by providing each other with conjectured class labels for unlabeled data. Multi-view learning has often proven to utilize unlabeled data effectively, increase the accuracy of classifiers (*e.g.,* Yarowsky, 1995; Blum & Mitchell, 1998) and improve the quality of clusterings (Bickel & Scheffer, 2004).

Nigam and Ghani (2000) propose the co-EM procedure that resembles semi-supervised learning with EM (McCallum & Nigam, 1998), using two views that alternate after each iteration. The EM algorithm (Dempster et al., 1977) is very well understood. In each iteration, it maximizes the expected joint log-likelihood of visible and invisible data given the parameter estimates of the previous iteration — the Q function. This procedure is known to greedily maximize the likelihood of the data. By contrast, the primary justification of the co-EM algorithm is that it often works very well; it is not known which criterion the method maximizes.

We take a top down approach on the problem of mixture model estimation in a multi-view setting. A result of Dasgupta et al. (2001) motivates our work by showing that a high consensus of independent hypotheses implies a low error rate. We construct a criterion that quantifies likelihood and consensus and derive a procedure that maximizes it. We contribute to an understanding of mixture model estimation for multiple views by showing that the co-EM algorithm is a special case of the resulting procedure. Our solution naturally generalizes co-EM for more than two views. We show that a variant of the method in which the consensus term is annealed over time is guaranteed to converge.

The rest of this paper is organized as follows. Section 2 discusses related work. In Section 3, we define the problem setting. Section 4 motivates our approach, discusses the new Q function, the unsupervised co-EM algorithm, and its instantiation for mixture of multinomials. We conduct experiments in Section 5 and conclude with Section 6.

2. Related Work

Most studies on multi-view learning address semi-supervised classification problems. de Sa (1994) observes a relationship between consensus of multiple hypotheses and their error rate and devised a semi-supervised learning method by cascading multi-view vector quantization and linear classification. A multi-view approach to word sense disambiguation combines a classifier that refers to the local context of a word with a second classifier that utilizes the document in which words co-occur (Yarowsky, 1995). Blum and

Appearing in *Proceedings of the Workshop on Learning with Multiple Views*, 22^{nd} ICML, Bonn, Germany, 2005.

Mitchell (1998) introduce the co-training algorithm for semi-supervised learning that greedily augments the training set of two classifiers. A version of the AdaBoost algorithm boosts the agreement between two views on unlabeled data (Collins & Singer, 1999).

Dasgupta et al. (2001) and Abney (2002) give PAC bounds on the error of co-training in terms of the disagreement rate of hypotheses on unlabeled data in two independent views. This justifies the direct minimization of the disagreement. The co-EM algorithm for semi-supervised learning probabilistically labels all unlabeled examples and iteratively exchanges those labels between two views (Nigam & Ghani, 2000; Ghani, 2002). Muslea et al. (2002) extend co-EM for active learning. Brefeld and Scheffer (2004) study a co-EM wrapper for the Support Vector Machine.

For unsupervised learning, several methods combine models that are learned using distinct attribute subsets in a way that encourages agreement. Becker and Hinton (1992) maximize mutual information between the output of neural network modules that perceive distinct views of the data. Models of images and their textual annotations have been combined (Barnard et al., 2002; Blei & Jordan, 2003). Reinforcement clustering (Wang et al., 2003) exchanges cluster membership information between views by artificial attributes. Bickel and Scheffer (2004) use the co-EM algorithm for clustering of data with two views. Clustering by maximizing the dependency between views is studied by Sinkkonen et al. (2004). Also, the density-based DBSCAN clustering algorithm has a multi-view counterpart (Kailing et al., 2004).

3. Problem Setting

The *multi-view* setting is characterized by available attributes X which are decomposed into views $X^{(1)}, \ldots, X^{(s)}$. An instance $x = (x^{(1)}, \ldots, x^{(s)})$ has representations $x^{(v)}$ that are vectors over $X^{(v)}$. We focus on the problem of estimating parameters of a generative mixture model in which data are generated as follows. The *data generation process* selects a mixture component j with probability α_j. Mixture component j is the value of a random variable Z. Once j is fixed, the generation process draws the s independent vectors $x^{(v)}$ according to the likelihoods $P(x^{(v)}|j)$. The likelihoods $P(x^{(v)}|j)$ are assumed to follow a parametric model $P(x^{(v)}|j, \Theta)$ (distinct views may of course be governed by distinct distributional models).

The *learning task* involved is to estimate the parameters $\Theta = (\Theta^{(1)}, \ldots, \Theta^{(s)})$ from data. The *sample* consists of n observations that usually contain only the visible attributes $x_i^{(v)}$ in all views v of the instances x_i. The vector Θ contains priors $\alpha_j^{(v)}$ and parameters of the likelihood $P(x_i^{(v)}|j, \Theta^{(v)})$, where $1 \leq j \leq m$ and m is the number of mixture components assumed by the model (clusters). Given Θ, we will be able to calculate a posterior $P(j|x^{(1)}, \ldots, x^{(s)}, \Theta)$. This posterior will allow us to assign a cluster membership to any instance $x = (x^{(1)}, \ldots, x^{(s)})$. The *evaluation metric* is the impurity of the clusters as measured by the entropy; the elements of each identified cluster should originate from the same true mixture component.

4. Derivation of the Algorithm

Dasgupta et al. (2001) have studied the relation between the consensus among two independent hypotheses and their error rate. Let us review a very simple result that motivates our approach, it can easily be derived from their general treatment of the topic. Let $h^{(v)}(x) = \text{argmax}_j \; P(j|x^{(v)}, \Theta^{(v)})$ be two independent clustering hypotheses in views $v = 1, 2$. For clarity of the presentation, let there be two true mixture components. Let x be a randomly drawn instance that, without loss of generality belongs to mixture component 1, and let both hypotheses $h^{(1)}$ and $h^{(2)}$ have a probability of at least 50% of assigning x to the correct cluster 1. We observe that

$$P(h^{(1)}(x) \neq h^{(2)}(x)) \geq \max_v P(h^{(v)}(x) \neq 1).$$

That is, the probability of a disagreement $h^{(1)}(x) \neq h^{(2)}(x)$ is an upper bound on the risk of an error $P(h^{(v)}(x) \neq 1)$ of either hypothesis $h^{(v)}$.

We give a brief *proof* of this observation. In Equation 1 we distinguish between the two possible cases of disagreement; we utilize the independence assumption and order the summands such that the greater one comes first. In Equation 2, we exploit that the error rate be at most 50%: both hypotheses are less likely to be wrong than just one of them. Exploiting the independence again takes us to Equation 3.

$$\begin{aligned}
& P(h^{(1)}(x) \neq h^{(2)}(x)) \\
&= P(h^{(v)}(x) = 1, h^{(\bar{v})}(x) = 2) + \\
&\quad P(h^{(v)}(x) = 2, h^{(\bar{v})}(x) = 1) && (1) \\
&\quad \text{where } v = \text{argmax}_u P(h^{(u)}(x) = 1, h^{(\bar{u})}(x) = 2) \\
&\geq P(h^{(v)}(x) = 2, h^{(\bar{v})}(x) = 2) + \\
&\quad P(h^{(v)}(x) = 2, h^{(\bar{v})}(x) = 1) && (2) \\
&= \max_v P(h^{(v)}(x) \neq 1) && (3)
\end{aligned}$$

In unsupervised learning, the risk of assigning instances to wrong mixture components cannot be minimized directly, but with the above argument we can minimize an upper bound on this risk.

The Q function is the core of the EM algorithm. We will now review the usual definition, include a consensus term, and find a maximization procedure.

4.1. Single-View Optimization Criterion

Even though the goal is to maximize $P(X|\Theta)$, EM iteratively maximizes an auxiliary (single-view) criterion $Q^{SV}(\Theta, \Theta_t)$. The criterion refers to the visible variables X, the invisibles Z (the mixture component), the optimization parameter Θ and the parameter estimates Θ_t of the last iteration. Equation 4 defines $Q^{SV}(\Theta, \Theta_t)$ to be the expected log-likelihood of $P(X, Z|\Theta)$, given X and given that the hidden mixture component Z be distributed according to $P(j|x, \Theta_t)$.

The criterion $Q^{SV}(\Theta, \Theta_t)$ can be determined as in Equation 5 for mixture models. It requires calculation of the posterior $P(j|x_i, \Theta_t)$ as in Equation 6; this is referred to as the E step of the EM algorithm. In the M step, it finds the new parameters $\Theta_{t+1} = \text{argmax}_\Theta\, Q^{SV}(\Theta, \Theta_t)$ that maximize Q^{SV} over Θ. The parameters Θ occur in Equation 5 only in the prior probabilities α_j and likelihood terms $P(x_i|j, \Theta)$.

$$Q^{SV}(\Theta, \Theta_t) = E[\log P(X, Z|\Theta)|X, \Theta_t] \quad (4)$$

$$= \sum_{i=1}^{n}\sum_{j=1}^{m} P(j|x_i, \Theta_t) \log(\alpha_j P(x_i|j, \Theta)) \quad (5)$$

$$P(j|x_i, \Theta_t) = \frac{\alpha_j P(x_i|j, \Theta_t)}{\sum_k \alpha_k P(x_i|k, \Theta_t)} \quad (6)$$

The EM algorithm starts with some initial guess at the parameters Θ_0 and alternates E and M steps until convergence. Dempster et al. (1977) prove that, in each iteration, $P(X|\Theta_{t+1}) - P(X|\Theta_t) \geq 0$. Wu (1983) furthermore proves conditions for the convergence of the sequence of parameters $(\Theta)_t$.

4.2. Multi-View Criterion

We want to maximize the likelihood in the individual views and the consensus of the models because we know that the disagreement bounds the risk of assigning an instance to an incorrect mixture component. Equations 7 and 8 define our *multi-view Q function* as the sum over s single-view Q functions minus a penalty term $\Delta(\cdot)$ that quantifies the disagreement of the models $\Theta^{(v)}$ and is regularized by η.

$$Q^{MV}(\Theta^{(1)}, \ldots, \Theta^{(s)}, \Theta_t^{(1)}, \ldots, \Theta_t^{(s)}) = \sum_{v=1}^{s} Q^{SV}(\Theta^{(v)}, \Theta_t^{(v)}) - \eta\Delta(\Theta^{(1)}, \ldots, \Theta^{(s)}, \Theta_t^{(1)}, \ldots, \Theta_t^{(s)}) \quad (7)$$

$$= \sum_{v=1}^{s} E\left[\log P(X^{(v)}, Z^{(v)}|\Theta^{(v)})|X^{(v)}, \Theta_t^{(v)}\right] - \eta\Delta(\Theta^{(1)}, \ldots, \Theta^{(s)}, \Theta_t^{(1)}, \ldots, \Theta_t^{(s)}) \quad (8)$$

When the regularization parameter η is zero, then $Q^{MV} = \sum_v Q^{SV}$. In each step, co-EM then maximizes the s terms Q^{SV} independently. It follows immediately from Dempster et al. (1977) that each $P(X^{(v)}|\Theta^{(v)})$ increases in each step and therefore $\sum_v P(X^{(v)}|\Theta^{(v)})$ is maximized.

The disagreement term Δ should satisfy a number of desiderata. Firstly, since we want to minimize Δ, it should be convex. Secondly, for the same reason, it should be differentiable. Given Θ_t, we would like to find the maximum of $Q^{MV}(\Theta, \Theta_t)$ in one single step. We would, thirdly, appreciate if Δ was zero when the views totally agree.

We construct Δ to fulfill these desiderata in Equation 9. It contains the pairwise cross entropy $H(P(j|x_i^{(v)}, \Theta_t^{(v)}), P(j|x_i^{(u)}, \Theta^{(u)}))$ of the posteriors of any pair of views u and v. The second cross entropy term $H(P(j|x_i^{(v)}, \Theta_t^{(v)}), P(j|x_i^{(v)}, \Theta^{(v)}))$ scales Δ down to zero when the views totally agree. Equation 10 expands all cross-entropy terms. At an abstract level, Δ can be thought of as all pairwise Kullback-Leibler divergences of the posteriors between all views. Since the cross entropy is convex, Δ is convex, too.

$$\Delta(\Theta^{(1)}, \ldots, \Theta^{(s)}, \Theta_t^{(1)}, \ldots, \Theta_t^{(s)}) = \frac{1}{s-1}\sum_{v\neq u}\sum_{i=1}^{n}\Big(H(P(j|x_i^{(v)}, \Theta_t^{(v)}), P(j|x_i^{(u)}, \Theta^{(u)})) - H(P(j|x_i^{(v)}, \Theta_t^{(v)}), P(j|x_i^{(v)}, \Theta^{(v)}))\Big) \quad (9)$$

$$= \frac{1}{s-1}\sum_{v\neq u}\sum_{i=1}^{n}\sum_{j=1}^{m} P(j|x_i^{(v)}, \Theta_t^{(v)}) \log\frac{P(j|x_i^{(v)}, \Theta^{(v)})}{P(j|x_i^{(u)}, \Theta^{(u)})} \quad (10)$$

In order to implement the M step, we have to maximize $Q^{MV}(\Theta, \Theta_t)$ given Θ_t. We have to set the derivative to zero. Parameter Θ occurs in the logarithmized posteriors, so we have to differentiate a sum of likelihoods within a logarithm. Theorem 1 solves this problem and rewrites Q^{MV} analogously to Equation 5.

Equation 12 paves the way to an algorithm that maximizes Q^{MV}. The parameters Θ occur only in the log-likelihood terms $\log P(x_i^{(v)}|j, \Theta^{(v)})$ and $\log \alpha_j^{(v)}$ terms, and Q^{MV} can be rewritten as a sum over local functions Q_v^{MV} for the views v. It now becomes clear that the M step can be executed by finding parameter estimates of $P(x_i^{(v)}|j, \Theta^{(v)})$ and $\alpha_j^{(v)}$ independently in each view v. The E step can be carried out by calculating and averaging the posteriors $P^{(v)}(j|x_i, \Theta_t, \eta)$ according to Equation 13; this equation specifies how the views interact.

Theorem 1 *The multi-view criterion Q can be expressed as a sum of local functions Q_v^{MV} (Equation 11) that can be maximized independently in each view v. The criterion can be calculated as in Equation 12, where $P^{(v)}(j|x_i, \Theta_t, \eta)$ is the averaged posterior as detailed in Equation 13 and $P(j|x_i^{(v)}, \Theta_t^{(v)})$ is the local posterior of view v, detailed in Equation 14.*

$$Q^{MV}(\Theta^{(1)}, \ldots, \Theta^{(s)}, \Theta_t^{(1)}, \ldots, \Theta_t^{(s)})$$

$$= \sum_{v=1}^{s} Q_v^{MV}(\Theta^{(v)}, \Theta_t^{(1)}, \ldots, \Theta_t^{(s)}) \quad (11)$$

$$= \sum_{v=1}^{s} \left(\sum_{i=1}^{n} \sum_{j=1}^{m} P^{(v)}(j|x_i, \Theta_t, \eta) \log \alpha_j^{(v)} \right. \quad (12)$$

$$\left. + \sum_{i=1}^{n} \sum_{j=1}^{m} P^{(v)}(j|x_i, \Theta_t, \eta) \log P(x_i^{(v)}|j, \Theta^{(v)}) \right)$$

$$P^{(v)}(j|x_i, \Theta_t^{(1)}, \ldots, \Theta_t^{(s)}, \eta) \quad (13)$$

$$= (1-\eta) P(j|x_i^{(v)}, \Theta_t^{(v)}) + \frac{\eta}{s-1} \sum_{\bar{v} \neq v} P(j|x_i^{(\bar{v})}, \Theta_t^{(\bar{v})})$$

$$P(j|x_i^{(v)}, \Theta_t^{(v)}) = \frac{\alpha_j^{(v)} P(x_i^{(v)}|j, \Theta_t^{(v)})}{\sum_k \alpha_k^{(v)} P(x_i^{(v)}|k, \Theta_t^{(v)})} \quad (14)$$

The proof for Theorem 1 is given in Appendix A.

4.3. Generalized Co-EM Algorithm

Theorem 1 describes the unsupervised co-EM algorithm with arbitrarily many views mathematically. The M steps can be executed independently in the views but Theorem 1 leaves open how the E and M steps should be interleaved. Co-EM can be implemented such that a global E step is followed by M steps in all views or, alternatively, we can iterate over the views in an outer loop and execute an E and an M step in the current view in each iteration of this loop.

We implement the latter strategy because consecutive M steps in multiple views impose the following risk. Cases can arise in which Q_1^{MV} can be maximized by changing $\Theta_{t+1}^{(1)}$ such that it agrees with $\Theta_t^{(2)}$. A consecutive M step in view 2 can then maximize Q_2^{MV} by changing $\Theta_{t+1}^{(2)}$ such that it agrees with $\Theta_t^{(1)}$. As a result, the two models flip their dissenting opinions. We observe empirically that this effect slows down the convergence; if the Q function consisted of only the Δ term, then this could even lead to alternation.

The unsupervised co-EM algorithm with multiple views is shown in Table 1. When the execution has reached time step t and view v, the parameters $\Theta_{t+1}^{(1)}, \ldots, \Theta_{t+1}^{(v-1)}$ and $\Theta_t^{(v)}, \ldots, \Theta_t^{(s)}$ have already been estimated. In the E step, we can therefore determine the posterior $P^{(v)}(j|x_i, \Theta_{t+1}^{(1)}, \ldots, \Theta_{t+1}^{(v-1)}, \Theta_t^{(v)}, \ldots, \Theta_t^{(s)}, \eta)$ using the most recent parameter estimates. In the succeeding M step, the local Q_v^{MV} function is maximized over the parameter $\Theta^{(v)}$. Note that the co-EM algorithm of Nigam and Ghani (2000) is a special case of Table 1 for two views, $\eta = 1$, and semi-supervised instead of unsupervised learning.

Table 1. Unsupervised Co-EM Algorithm with Multiple Views.

Input: Unlabeled data $(x_i^{(1)}, \ldots, x_i^{(s)}) \in D$. Regularization parameter η (by default, 1).

1. Initialize $\Theta_0^{(1)}, \ldots, \Theta_0^{(s)}$ at random; let $t = 1$.
2. Do until convergence of Q^{MV}:
   (a) For $v = 1 \ldots s$:
      i. E step in view v: Compute the posterior $P^{(v)}(j|x_i, \Theta_{t+1}^{(1)}, \ldots, \Theta_{t+1}^{(v-1)}, \Theta_t^{(v)}, \ldots, \Theta_t^{(s)}, \eta)$ in view v using Equation 13.
      ii. M step in view v: maximize Q^{MV}; $\Theta_{t+1}^{(v)} = \text{argmax}_{\Theta^{(v)}} Q_v^{MV}(\Theta^{(v)}, \Theta_{t+1}^{(1)}, \ldots, \Theta_{t+1}^{(v-1)}, \Theta_t^{(v)}, \ldots, \Theta_t^{(s)})$.
   (c) Increment t.
3. Return $\Theta = (\Theta_t^{(1)}, \ldots, \Theta_t^{(s)})$.

In every step 2(a)ii, the local function Q_v^{MV} increases. Since all other $Q_{\bar{v}}^{MV}$ are constant in $\Theta^{(v)}$, this implies that also the global function Q^{MV} increases. In each iteration of the regular EM algorithm, $P(X|\Theta_{t+1}) - P(X|\Theta_t) \geq 0$. For co-EM, this is clearly not the case since the Q function has been augmented by a dissent penalization term. Wu (1983) proves conditions for the convergence of the sequence $(\Theta)_t$ for regular EM. Sadly, the proof does not transfer to co-EM.

We study a variant of the algorithm for which convergence can be proven. In an additional step 2(b), η is decremented towards zero according to some annealing scheme. This method can be guaranteed to converge; the proof is easily derived from the convergence guarantees of regular EM (Dempster et al., 1977; Wu, 1983). We can furthermore show that co-EM with annealing of η maximizes $\sum_v P(X^{(v)}|\Theta)$. In the beginning of the optimization process, Δ contributes strongly to the criterion Q^{MV}; the dissent Δ is convex and we know that it upper-bounds the error. Therefore, Δ guides the search to a parameter region of low error. The contribution of Δ vanishes later; $\sum_v P(X^{(v)}|\Theta)$ usually has many local maxima and having added Δ earlier now serves as a heuristic that may lead to a good local maximum.

4.4. Global Prior Probabilities

According to our generative model we have one global prior for each mixture component, but in step 2(a)ii the co-EM algorithm so far estimates priors in each view v from the data. We will now focus on maximization of Q subject to the constraint that the estimated priors of all views be equal.

We introduce two sets of Lagrange multipliers and get Lagrangian $L(\alpha, \lambda, \gamma)$ in Equation 15. Multiplier $\lambda^{(v)}$ guarantees that $\sum_j \alpha_j^{(v)} = 1$ in view v and $\gamma^{(j,v)}$ enforces the constraint $\alpha_j^{(1)} = \alpha_j^{(v)}$ for component j.

$$L(\alpha, \lambda, \gamma) = \sum_{v=1}^{s}\sum_{i=1}^{n}\sum_{j=1}^{m} P^{(v)}(j|x_i, \Theta_t, \eta) \log \alpha_j^{(v)} + \sum_{v=1}^{s} \lambda^{(v)} \left(\sum_{j=1}^{m} \alpha_j^{(v)} - 1 \right) + \sum_{v=2}^{s}\sum_{j=1}^{m} \gamma^{(j,v)}(\alpha_j^{(1)} - \alpha_j^{(v)}) \quad (15)$$

Setting the partial derivatives of $L(\alpha, \lambda, \gamma)$ to zero and solving the resulting system of equations leads to Equation 16. Expanding $P^{(v)}(j|x_i, \Theta_t, \eta)$, the regularization parameter η cancels out and we reach the final M step for $\alpha_j^{(v)}$ in Equation 17. We can see that the estimated prior is an average over all views and is therefore equal for all views.

$$\alpha_j^{(v)} = \frac{1}{sn}\sum_{v=1}^{s}\sum_{i=1}^{n} P^{(v)}(j|x_i, \Theta_t, \eta) \quad (16)$$

$$= \frac{1}{sn}\sum_{v=1}^{s}\sum_{i=1}^{n} P(j|x_i^{(v)}, \Theta_t^{(v)}) = \alpha_j \quad (17)$$

4.5. Cluster Assignment

For cluster analysis, an assignment of instances to clusters has to be derived from the model parameters. The risk of deciding for an incorrect cluster is minimized by choosing the *maximum a posteriori* hypothesis as in Equation 18. Bayes' rule and the conditional independence assumption lead to Equation 19.

$$h(x_i) = \text{argmax}_j P(j|x_i, \Theta) \quad (18)$$

$$= \underset{j}{\text{argmax}} \frac{\alpha_j \prod_{v=1}^{s} P(x_i^{(v)}|j, \Theta^{(v)})}{\sum_k^m \alpha_k \prod_{v=1}^{s} P(x_i^{(v)}|k, \Theta^{(v)})} \quad (19)$$

4.6. Mixture of Multinomials

In step 2(a)ii the co-EM algorithm estimates parameters in view v from the data. This step is instantiated for the specific distributional model used in a given application. We will detail the maximization steps for multinomial models which we use in our experimentation because they model both text and link data appropriately.

A multinomial model j is parameterized by the probabilities $\theta_{lj}^{(v)}$ of word w_l in view v and mixture component j. The likelihood of document $x_i^{(v)}$ is given by Equation 20. Parameters $n_{il}^{(v)}$ count the occurrences of word w_l in document $x_i^{(v)}$. $P(|x_i^{(v)}|)$ is the prior on the document length. The factorials account for all possible sequences that result in the set of words $x_i^{(v)}$.

$$P(x_i^{(v)}|j, \Theta^{(v)}) = P(|x_i^{(v)}|)|x_i^{(v)}|! \prod_l \frac{(\theta_{lj}^{(v)})^{n_{il}^{(v)}}}{n_{il}^{(v)}!} \quad (20)$$

We will now focus on maximization of Q^{MV} over the parameters $\theta_{lj}^{(v)}$. Lagrangian $L(\theta, \lambda)$ in Equation 21 guarantees that the word probabilities sum to one.

$$L(\theta, \lambda) = \sum_{v=1}^{s}\sum_{i=1}^{n}\sum_{j=1}^{m} P^{(v)}(j|x_i, \Theta_t, \eta) \left(\log P(|x_i^{(v)}|)|x_i^{(v)}|! + \sum_l n_{il}^{(v)} (\log \frac{\theta_{lj}^{(v)}}{n_{il}^{(v)}!} \right) + \sum_{v=1}^{s}\sum_{j=1}^{m} \lambda_j^{(v)} \left(\sum_l \theta_{lj}^{(v)} - 1 \right) \quad (21)$$

Setting the partial derivatives to zero and solving the resulting system of equations yields Equation 22.

$$\theta_{lj}^{(v)} = \frac{\sum_i P^{(v)}(j|x_i, \Theta_t, \eta) n_{il}^{(v)}}{\sum_k \sum_i P^{(v)}(j|x_i, \Theta_t, \eta) n_{ik}^{(v)}} \quad (22)$$

5. Empirical Studies

We want to find out (1) whether co-EM with multiple views finds better clusters in sets of linked documents with mixture of multinomials than regular single-view EM; (2) whether co-EM is still beneficial when there is no natural feature split in the data; (3) whether there are problems for which the optimal number of views lies above 2; and (4) whether the consensus regularization parameter η should be annealed or fixed to some value. To answer these questions, we experiment on archives of linked and plain text documents. All data sets that we use contain labeled instances; the labels are not visible to the learning method but we use them to measure the impurity of the returned clusters. Our quality measure is the average entropy over all clusters (Equation 23). This measure corresponds to the average number of bits needed to code the real class labels given the clustering result. The frequency $\hat{p}_{i|j}$ counts the number of elements of class i in cluster j, and n_j is the size of cluster j.

$$H = \sum_{j=1}^{m} \frac{n_j}{n} \left(-\sum_i \hat{p}_{i|j} \log \hat{p}_{i|j} \right) \quad (23)$$

The mixture of multinomials model for text assumes that a document is generated by first choosing a component j, and then drawing a number of words with

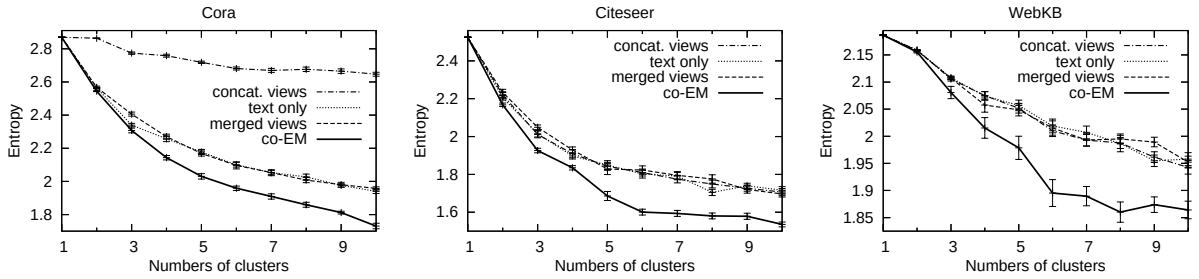

Figure 1. Average cluster impurity over varying numbers of clusters.

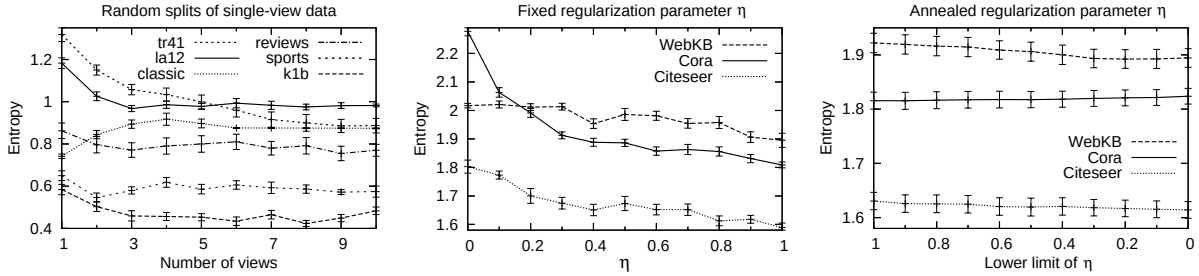

Figure 2. Six single-view data sets with random feature splits into views (left); tuning the regularization parameter η to a fixed value (center); annealing η during the optimization process (right).

replacement according to a component-specific likelihood. The multinomial link model analogously assumes that, for a document x, a number of references *from* or *to* other documents are drawn according to a component-specific likelihood. We first use three sets of linked documents for our experimentation. The *Citeseer* data set contains 3,312 entries that belong to six classes. The text view consists of title and abstract of a paper; the two link views are inbound and outbound references. The *Cora* data set contains 9,947 computer science papers categorized into eight classes. In addition to the three views of the Citeseer data set we extract an anchor text view that contains three sentences centered at the occurrence of the reference in the text. The *WebKB* data set is a collection of 4,502 academic web pages manually grouped into six classes. Two views contain the text on the page and the anchor text of all inbound links, respectively. The total number of views are 2 (WebKB), 3 (Citeseer), and 4 (Cora).

Note that web pages or publications do not necessarily have inbound or outbound links. We require only the title/abstract and web page text views to contain attributes. The other views are empty in many cases; the inbound link view of 45% of the Cora instances is empty. In order to account for this application-specific property, we include only non-empty views in the averaged posterior $P^{(v)}(j|x_i, \Theta_t, \eta)$.

We use two single-view baselines. The first baseline applies single-view EM to a concatenation of all views (caption "concat. views"). The second baseline merges all text views (anchor text and intrinsic text are merged into one bag) and separately merges all link views (corresponding to an undirected graphical model). Single-view EM is than applied to the concatenation of these views ("merged views"). All results are averaged over 20 runs and error bars indicate standard error. Figure 1 details the clustering performance of the algorithm and baselines for various numbers of clusters (mixture components assumed by the model). Co-EM outperforms the baselines for all problems and any number of clusters.

In order to find out how multi-view co-EM performs when there is no natural feature split in the data, we randomly draw six single-view document data sets that come with the cluto clustering toolkit (Zhao & Karypis, 2001). We randomly split the available attributes into s subsets and average the performance over 20 distinct attribute splits. We set the number of clusters to the respective number of true mixture components. Figure 2 (left) shows the results for several numbers of views. We can see that in all but one case the best number of views is greater than one. In four of six cases we can reject the null hypothesis that one view incurs a lower entropy than two views at a significance level of $\alpha = 0.01$. Additionally, in 2 out of six cases, three views lead to significantly better clusters than two views; in four out of six cases, the entropy has its empirical minimum for more than two views.

In all experiments so far, we fixed $\eta = 1$. Let us study whether tuning or annealing η improves the cluster quality. Figure 2 (center) shows the entropy for various

fixed values of η; we see that 1 is the best setting ($\eta > 1$ would imply negative word probabilities $\theta_{lj}^{(v)}$).

Let us finally study whether a fixed value of η or annealing η results in a better cluster quality. In the following experiments, η is initialized at 1 and slowly annealed towards 0. Figure 2 (right) shows the development of the cluster entropy as η approaches towards 0. We see that fixing and annealing η empirically works equally well; annealing η causes a slight improvement in two cases and a slight deterioration of the quality in one case. The distinction between co-EM with and without annealing of η lies in the fact that convergence can only be proven when η is annealed; empirically, these variants are almost indistinguishable.

6. Conclusion

The Q^{MV} function defined in Equation 7 augments the single-view optimization criterion Q^{SV} by penalizing disagreement among distinct views. This is motivated by the result that the consensus among independent hypotheses upper-bounds the error rate of either hypothesis. Theorem 1 rewrites the criterion $Q^{MV}(\Theta, \Theta_t)$ such that it can easily be maximized over Θ when Θ_t is fixed: an M step is executed locally in each view. Maximizing Q^{MV} naturally leads to a version of the co-EM algorithm for arbitrarily many views and unlabeled data. Our derivation thus explains, motivates, and generalizes the co-EM algorithm.

While the original co-EM algorithm cannot be shown to converge, a variant of the method that anneals η over time can be guaranteed to converge and to (locally) maximize $\sum_v P(X^{(v)}|\Theta)$. Initially amplifying the convex error bound Δ in the criterion Q^{MV} serves as a heuristic that guides the search towards a better local optimum.

Our experiments show that co-EM is a better clustering procedure than single-view EM for actual multi-view problems such as clustering linked documents. Surprisingly, we also found that in most cases the impurity of text clusters can be reduced by splitting the attributes at random and applying multi-view clustering. This indicates that the consensus maximization principle may contribute to methods for a broader range of machine learning problems.

Acknowledgment

This work has been supported by the German Science Foundation DFG under grant SCHE540/10-1.

References

Abney, S. (2002). Bootstrapping. *Proc. of the 40th Annual Meeting of the Association for Comp. Linguistics.*

Barnard, K., Duygulu, P., Forsyth, D., de Freitas, N., Blei, D., & Jordan, M. (2002). Matching words and pictures. *Journal of Machine Learning Research, 3*, 1107–1135.

Becker, S., & Hinton, G. (1992). A self-organizing neural network that discovers surfaces in random-dot stereograms. *Nature, 355*, 161–163.

Bickel, S., & Scheffer, T. (2004). Multi-view clustering. *Proc. of the IEEE International Conf. on Data Mining.*

Blei, D., & Jordan, M. (2003). Modeling annotated data. *Proceedings of the ACM SIGIR Conference on Information Retrieval.*

Blum, A., & Mitchell, T. (1998). Combining labeled and unlabeled data with co-training. *Proceedings of the Conference on Computational Learning Theory.*

Brefeld, U., & Scheffer, T. (2004). Co-EM support vector learning. *Proc. of the Int. Conf. on Machine Learning.*

Collins, M., & Singer, Y. (1999). Unsupervised models for named entity classification. *Proc. of the Conf. on Empirical Methods in Natural Language Processing.*

Dasgupta, S., Littman, M., & McAllester, D. (2001). PAC generalization bounds for co-training. *Proceedings of Neural Information Processing Systems.*

de Sa, V. (1994). Learning classification with unlabeled data. *Proc. of Neural Information Processing Systems.*

Dempster, A., Laird, N., & Rubin, D. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B, 39.*

Ghani, R. (2002). Combining labeled and unlabeled data for multiclass text categorization. *Proceedings of the International Conference on Machine Learning.*

Kailing, K., Kriegel, H., Pryakhin, A., & Schubert, M. (2004). Clustering multi-represented objects with noise. *Proceedings of the Pacific-Asia Conference on Knowledge Discovery and Data Mining.*

McCallum, A., & Nigam, K. (1998). Employing EM in pool-based active learning for text classification. *Proc. of the International Conference on Machine Learning.*

Muslea, I., Kloblock, C., & Minton, S. (2002). Active + semi-supervised learning = robust multi-view learning. *Proc. of the International Conf. on Machine Learning.*

Nigam, K., & Ghani, R. (2000). Analyzing the effectiveness and applicability of co-training. *Proceedings of the Workshop on Information and Knowledge Management.*

Sinkkonen, J., Nikkilä, J., Lahti, L., & Kaski, S. (2004). Associative clustering. *Proceedings of the European Conference on Machine Learning.*

$$\sum_{v\neq u}\sum_{i=1}^{n}\sum_{j=1}^{m} P(j|x_i^{(v)},\Theta_t^{(v)})\log\frac{P(x_i^{(v)}|\Theta^{(v)})}{P(x_i^{(u)}|\Theta^{(u)})} = \sum_{v\neq u}\sum_{i=1}^{n}\log\frac{P(x_i^{(v)}|\Theta^{(v)})}{P(x_i^{(u)}|\Theta^{(u)})} \tag{24}$$

$$= \sum_{v<u}\sum_{i=1}^{n}\left(\log\frac{P(x_i^{(v)}|\Theta^{(v)})}{P(x_i^{(u)}|\Theta^{(u)})}+\log\frac{P(x_i^{(u)}|\Theta^{(u)})}{P(x_i^{(v)}|\Theta^{(v)})}\right) = \sum_{v<u}\sum_{i=1}^{n}\log\frac{P(x_i^{(v)}|\Theta^{(v)})P(x_i^{(u)}|\Theta^{(u)})}{P(x_i^{(u)}|\Theta^{(u)})P(x_i^{(v)}|\Theta^{(v)})} = \sum_{v<u}\sum_{i=1}^{n}\log 1 = 0 \tag{25}$$

$$\Delta(\Theta^{(1)},\ldots,\Theta^{(s)},\Theta_t^{(1)},\ldots,\Theta_t^{(s)})$$

$$= \frac{1}{s-1}\sum_{v\neq u}\sum_{i=1}^{n}\sum_{j=1}^{m} P(j|x_i^{(v)},\Theta_t^{(v)})\log\frac{P(j|x_i^{(v)},\Theta^{(v)})}{P(j|x_i^{(u)},\Theta^{(u)})} + \frac{1}{s-1}\sum_{v\neq u}\sum_{i=1}^{n}\sum_{j=1}^{m} P(j|x_i^{(v)},\Theta_t^{(v)})\log\frac{P(x_i^{(v)}|\Theta^{(v)})}{P(x_i^{(u)}|\Theta^{(u)})} \tag{26}$$

$$= \frac{1}{s-1}\sum_{v\neq u}\sum_{i=1}^{n}\sum_{j=1}^{m} P(j|x_i^{(v)},\Theta_t^{(v)})\log\frac{P(x_i^{(v)},j|\Theta^{(v)})}{P(x_i^{(u)},j|\Theta^{(u)})} = \frac{1}{s-1}\sum_{v\neq u}\sum_{i=1}^{n}\sum_{j=1}^{m} P(j|x_i^{(v)},\Theta_t^{(v)})\log\frac{\alpha_j^{(v)}P(x_i^{(v)}|j,\Theta^{(v)})}{\alpha_j^{(u)}P(x_i^{(u)}|j,\Theta^{(u)})} \tag{27}$$

$$= \sum_{v=1}^{s}\sum_{i=1}^{n}\sum_{j=1}^{m} P(j|x_i^{(v)},\Theta_t^{(v)})\log(\alpha_j^{(v)}P(x_i^{(v)}|j,\Theta^{(v)})) - \frac{1}{s-1}\sum_{v=1}^{s}\sum_{i=1}^{n}\sum_{j=1}^{m}\left(\sum_{\bar{v}} P(j|x_i^{(\bar{v})},\Theta_t^{(\bar{v})})\right)\log(\alpha_j^{(v)}P(x_i^{(v)}|j,\Theta^{(v)})) \tag{28}$$

$$Q^{MV}(\Theta^{(1)},\ldots,\Theta^{(s)},\Theta_t^{(1)},\ldots,\Theta_t^{(s)})$$

$$= \sum_{v=1}^{s}\sum_{i=1}^{n}\sum_{j=1}^{m} P(j|x_i^{(v)},\Theta_t^{(v)})\log(\alpha_j^{(v)}P(x_i^{(v)}|j,\Theta^{(v)})) - \eta\sum_{v=1}^{s}\sum_{i=1}^{n}\sum_{j=1}^{m} P(j|x_i^{(v)},\Theta_t^{(v)})\log(\alpha_j^{(v)}P(x_i^{(v)}|j,\Theta^{(v)}))$$
$$+\frac{\eta}{s-1}\sum_{v=1}^{s}\sum_{i=1}^{n}\sum_{j=1}^{m}\left(\sum_{\bar{v}} P(j|x_i^{(\bar{v})},\Theta_t^{(\bar{v})})\right)\log(\alpha_j^{(v)}P(x_i^{(v)}|j,\Theta^{(v)})) \tag{29}$$

$$= \sum_{v=1}^{s}\sum_{i=1}^{n}\sum_{j=1}^{m}\log(\alpha_j^{(v)}P(x_i^{(v)}|j,\Theta^{(v)}))\left((1-\eta)P(j|x_i^{(v)},\Theta_t^{(v)}) + \frac{\eta}{s-1}\sum_{\bar{v}} P(j|x_i^{(\bar{v})},\Theta_t^{(\bar{v})})\right) \tag{30}$$

$$= \sum_{v=1}^{s}\sum_{i=1}^{n}\sum_{j=1}^{m} P^{(v)}(j|x_i,\Theta_t,\eta)\log(\alpha_j^{(v)}P(x_i^{(v)}|j,\Theta^{(v)})) \tag{31}$$

$$= \sum_{v=1}^{s}\sum_{i=1}^{n}\sum_{j=1}^{m} P^{(v)}(j|x_i,\Theta_t,\eta)\log\alpha_j^{(v)} + \sum_{v=1}^{s}\sum_{i=1}^{n}\sum_{j=1}^{m} P^{(v)}(j|x_i,\Theta_t,\eta)\log P(x_i^{(v)}|j,\Theta^{(v)}) \tag{32}$$

Wang, J., Zeng, H., Chen, Z., Lu, H., Tao, L., & Ma, W. (2003). Recom: Reinforcement clustering of multi-type interrelated data objects. *Proceedings of the ACM SIGIR Conference on Information Retrieval*.

Wu, J. (1983). On the convergence properties of the EM algorithm. *The Annals of Statistics*, *11*, 95–103.

Yarowsky, D. (1995). Unsupervised word sense disambiguation rivaling supervised methods. *Proc. of the Annual Meeting of the Association for Comp. Ling.*.

Zhao, Y., & Karypis, G. (2001). *Criterion functions for document clustering: Experiments and analysis* (Technical Report TR 01-40). Department of Computer Science, University of Minnesota, Minneapolis, MN.

Appendix

A. Proof of Theorem 1

In order to prove Theorem 1 we first prove two additional equations. Firstly, we prove that the left term of Equation 24 equals zero. Equation 24 holds because $\sum_{j=1}^{m} P(j|x_i,\Theta_t)\log C = \log C$ when C is independent of j. Instead of summing over all two-way combinations of views we sum only once over each pairwise combination in the left term of Equation 25 and merge the logarithms. The terms in the resulting fraction cancel to one.

Secondly, we simplify the dissent function Δ. In Equation 26 we add a term (Equation 24) that we proved to be zero in Equation 25. Equations 27 merge the logarithms and apply the chain rule to extract $\alpha_j^{(v)}$. In Equation 28 the logarithm is split up and the sum over all pairwise view combinations is substituted with a nested sum.

Now we can prove Theorem 1. We write Q^{MV} as a sum of single-view Q^{SV} criteria (Equation 5) and the transformed Δ of Equation 28, resulting in Equation 29. With Equation 30 the $\log(\alpha_j^{(v)}P(x_i^{(v)}|j,\Theta^{(v)}))$ terms are factored out. We introduce the abbreviation $P^{(v)}(j|x_i,\Theta_t,\eta)$ in Equation 31. Finally the logarithm is split up (Equation 32) and the proof is finished. □