Information Theoretic Clustering of Sparse Co-Occurrence Data

Inderjit S. Dhillon and Yuqiang Guan
Department of Computer Sciences
University of Texas
Austin, TX 78712-1188, USA
inderjit, yguan@cs.utexas.edu

Abstract

A novel approach to clustering co-occurrence data poses it as an optimization problem in information theory which minimizes the resulting loss in mutual information. A divisive clustering algorithm that monotonically reduces this loss function was recently proposed. In this paper we show that sparse high-dimensional data presents special challenges which can result in the algorithm getting stuck at poor local minima. We propose two solutions to this problem: (a) a "prior" to overcome infinite relative entropy values as in the supervised Naive Bayes algorithm, and (b) local search to escape local minima. Finally, we combine these solutions to get a robust algorithm that is computationally efficient. We present experimental results to show that the proposed method is effective in clustering document collections and outperforms previous information-theoretic clustering approaches.

1 Introduction

Clustering is a central problem in unsupervised learning [5]. Presented with a set of data points, clustering algorithms group the data into clusters according to some notion of similarity between data points. However, the choice of similarity measure is a challenge and often an *ad hoc* measure is chosen. Information Theory comes to the rescue in the important situations where non-negative co-occurrence data is available. A novel formulation poses the clustering problem as one in information theory: find the clustering that minimizes the loss in (mutual) information [8, 4]. This information-theoretic formulation leads to a "natural" divisive clustering algorithm that uses relative entropy as the measure of similarity and monotonically reduces the loss in mutual information [4].

However, sparse and high-dimensional data presents special challenges and can lead to qualitatively poor local minima. In this paper, we demonstrate these failures and then propose two solutions to overcome these problems. First, we use a prior as in the supervised Naive Bayes algorithm to overcome infinite relative entropy values caused by sparsity. Second, we propose a local search strategy that is highly effective for high-dimensional data. We combine these solutions to get an effective, computationally efficient algorithm. A prime example of high-dimensional co-occurrence data is word-document data; we show that our algorithm returns clusterings that are better than those returned by previous information-theoretic approaches.

The following is a brief outline of the paper. Section 2 presents the information-theoretic framework and divisive clustering algorithm of [4]. The problems due to sparsity and high-dimensionality are illustrated in Section 3. We present our two-pronged solution to the problem in Section 4. Detailed experimental results are given in Section 5.

A word about notation. Upper-case letters such as X, Y will denote random variables, while lower-case letters such as x, y denote individual set elements. $\hat{Y}$ denotes a random variable obtained from a clustering of Y while $\hat{y}$ denotes an individual cluster. Probability distributions will be denoted by $p(X)$, $p(X|y)$. Boldfaced letters, such as $\mathbf{y}, \hat{\mathbf{y}}$, will denote $p(X|y), p(X|\hat{y})$ for brevity. The logarithmic base 2 is used throughout this paper.

2 Divisive Information-Theoretic Clustering

Let X and Y be two discrete random variables that take values in the sets $\{x_1, x_2, \ldots, x_m\}$ and $\{y_1, y_2, \ldots, y_n\}$ respectively. Suppose that we know their joint probability distribution $p(X, Y)$; often this can be estimated using co-occurrence data. Consider the case where we want to cluster Y. Let $\hat{Y}$ denote the "clustered" random variable that ranges over the disjoint clusters $\hat{y}_1, \ldots, \hat{y}_k$, i.e.,

$$\cup_{i=1}^{k} \hat{y}_i = \{y_1, \ldots, y_n\}, \text{ and } \hat{y}_i \cap \hat{y}_j = \phi, \quad i \neq j.$$

A novel information-theoretic approach to clustering is to seek the clustering which gives the smallest loss in mutual information [8, 4], i.e. to minimize

$$I(X;Y) - I(X;\hat{Y}) = \sum_{\hat{y}} \sum_{y \in \hat{y}} p(y) KL(p(X|y), p(X|\hat{y})), \tag{1}$$

where $I(X;Y)$ is mutual information between random variable X and Y and KL stands for Kullback-Leibler divergence [1]. The above expression for the loss in mutual information suggests a "natural" divisive clustering algorithm (DITC), which iteratively (i) re-partitions the distributions $p(X|y)$ by their closeness in KL-divergence to the cluster distributions $p(X|\hat{y})$, and (ii) subsequently, given the new clusters, re-computes the new cluster distributions. This procedure is iterated until change in objective function value as given in (1) is less than, say, 10^{-3}. See [4] for details.

3 Challenges due to Sparsity and High-Dimensionality

Unfortunately, Algorithm DITC can falter in the presence of sparsity and high-dimensionality.

Example 1 *(Sparsity) Consider the three conditional distributions:*

$\mathbf{y}_1$	$\mathbf{y}_2$	$\mathbf{y}_3$
.1	0	0
.9	.9	.1
0	.1	.9

. Suppose we want to cluster $\mathbf{y}_1, \mathbf{y}_2$ and $\mathbf{y}_3$ into two clusters; clearly the optimal clustering puts $\{\mathbf{y}_1, \mathbf{y}_2\}$ in one cluster and $\{\mathbf{y}_3\}$ in the other. However suppose the initial clusters are $\hat{\mathbf{y}}_1 = \{\mathbf{y}_1\}$ and $\hat{\mathbf{y}}_2 = \{\mathbf{y}_2, \mathbf{y}_3\}$. Then the cluster distributions will be $\hat{\mathbf{y}}_1 = (.1, .9, 0)$ and $\hat{\mathbf{y}}_2 = (0, .5, .5)$, respectively. The Kullback-Leibler divergences $KL(\mathbf{y}_1, \hat{\mathbf{y}}_2)$, $KL(\mathbf{y}_2, \hat{\mathbf{y}}_1)$ and $KL(\mathbf{y}_3, \hat{\mathbf{y}}_1)$ are infinite. Therefore Algorithm DITC gets stuck in this initial clustering and misses the optimal partition due to the presence of zeros in the cluster distributions $\hat{\mathbf{y}}_1$ and $\hat{\mathbf{y}}_2$ that result in infinite KL-divergences.

Example 2 *(High dimensionality) For the second example, we took a collection of 30 documents consisting of 10 documents each from the three distinct* classes *MEDLINE, CISI and CRAN (see Section 5 for details). These 30 documents contain a total of 1073 words and so the data is very high-dimensional. However, when we run DITC using the word-document co-occurrence data, there is hardly any movement of documents between clusters irrespective of the starting partition.*

4 Proposed Algorithm

In this section, we propose a computationally efficient algorithm that avoids the above problems due to sparsity and high-dimensionality. As in the supervised Naive Bayes method, we wish to perturb $p(X|\hat{y})$ to avoid zero probabilities. Recall that in our unsupervised case $p(X|\hat{y})$ refers to a cluster distribution. The important question is: what should be the perturbation? For reasons outlined below, we perturb the cluster distribution to:

$$p'(X|\hat{y}) = \frac{1}{1+\alpha} \left(p(X|\hat{y}) + \alpha \cdot u(X) \right), \tag{2}$$

where α is a constant and $u(X)$ is the uniform distribution $(\frac{1}{m}, ..., \frac{1}{m})$. The value of this prior has a pleasing property: the perturbed cluster distribution $p'(X|\hat{y})$ can be interpreted as the mean distribution for $\hat{y}$ obtained after perturbing each element of the input joint distribution $p(x,y)$ to

$$p'(x,y) = \frac{1}{1+\alpha} \left(p(x,y) + \frac{\alpha}{m} p(y) \right).$$

Note that if $\alpha = \frac{m}{\sum_x N(x,\hat{y})}$ in (2), where $N(x,\hat{y})$ is the frequence of x in cluster $\hat{y}$, we get Laplace's rule of succession used in supervised Naive Bayes, i.e., $p'(x|\hat{y}) = \frac{1+N(x,\hat{y})}{m+\sum_x N(x,\hat{y})}$. What should be the value of α in our clustering algorithm? Experimental results reveal that an "annealing" approach helps, i.e., start the algorithm with a large value of α and decrease α progressively as the number of iterations increase. Algorithm DITC_prior is same as DITC except that the cluster distributions are computed as in (2) and α is halved at every iteration. Our prior has the same influence as the temperature in deterministic annealing [7] through a slightly different mechanism: when the prior is big all the $p(X|\hat{y})$'s are uniform, i.e., the joint entropy $H(X,\hat{Y})$ is large, thus $KL(p(X|y), p(X|\hat{y}))$ is almost the same for all y and $\hat{y}$. As the prior decreases $H(X,\hat{Y})$ is decreased.

To further improve our algorithm, we turn to a local search strategy, called *first variation* in [3], that allows us to escape undesirable local minimum, especially in the case of high-dimensionality. Precisely, a first variation of a partition $\{\hat{y}_j\}_{j=1}^k$ is a partition $\{\hat{y}'_j\}_{j=1}^k$ obtained by removing a distribution y from a cluster $\hat{y}_j$ and assigning it to an existing cluster $\hat{y}_l$. Among all the kn possible first variations, corresponding to each combination of y and $\hat{y}_l$, we choose the one that gives the smallest loss in mutual information. As in [3], a chain of first variations are implemented for our DITC_LocalSearch algorithm, which iterates over DITC followed by a chain first variations. Finally, our algorithm DITC_PLS incorporates both the ideas of priors and local search, i.e., it iteratively runs DITC_prior and a chain of first variations till it converges. Lack of space prevents us from giving a more detailed description of the algorithm which may be found in [2].

	MED	CRAN	CISI
$\hat{y}_1$	**847**	41	275
$\hat{y}_2$	142	**954**	86
$\hat{y}_3$	44	405	**1099**

DITC results

	MED	CRAN	CISI
$\hat{y}_1$	**1016**	1	2
$\hat{y}_2$	1	**1389**	1
$\hat{y}_3$	16	9	**1457**

DITC_prior results

Table 1. Confusion matrices for 3893 documents, 4303 words (CLASSIC3)

	MED	CRAN	CISI
$\hat{y}_1$	1	15	**29**
$\hat{y}_2$	**13**	11	8
$\hat{y}_3$	**36**	24	13

DITC_prior results

	MED	CRAN	CISI
$\hat{y}_1$	**50**	0	1
$\hat{y}_2$	0	**50**	0
$\hat{y}_3$	0	0	**49**

DITC_LocalSearch results

Table 2. Algorithm DITC_LocalSearch yields better confusion matrix than DITC_prior (150 documents, 3652 words)

	MED	CRAN	CISI
$\hat{y}_1$	**45**	38	35
$\hat{y}_2$	31	26	**33**
$\hat{y}_3$	24	**36**	32

DITC_prior results

	MED	CRAN	CISI
$\hat{y}_1$	**97**	0	0
$\hat{y}_2$	1	**100**	0
$\hat{y}_3$	2	0	**100**

DITC_LocalSearch results

Table 3. Algorithm DITC_LocalSearch yields better confusion matrix than DITC_prior (300 documents, 5577 words)

5 Experimental Results

We now present experimental results for our information-theoretic algorithm applied to the task of clustering document collections using word-document co-occurrence data.

For our test data, we use various subsets of the 20-newsgroup data (NG20) [6] and the SMART collection (ftp://ftp.cs.cornell.edu/pub/smart). NG20 consists of approximately 20,000 newsgroup postings collected from 20 different usenet newsgroups. We report results on NG20 and various subsets of this data set of size 500 each: Binary, Multi5, Multi10 and NG10 (see [2] for details). In order for our results to be comparable, we applied the same preprocessing as in [8] to all the news group data sets, i.e. we removed stopwords and selected the 2000 words with the highest contribution to the mutual information, removed documents with less than 10 word occurrences and removed all the headers except the subject line.

From SMART, we used MEDLINE, CISI, and CRANFIELD subcollections, which consist of 1033, 1460 and 1400 abstracts respectively. We also created 3 subsets of 30, 150, and 300 documents respectively; each data set was created by equal sampling of the three collections. After removing stopwords, the number of words for the 30, 150 and 300 document data sets is 1073, 3658 and 5577 respectively. We refer to the entire data set as CLASSIC3 and the subsets as C30, C150 and C300 respectively.

Since we know the underlying class labels for our data sets, we can evaluate clustering results by forming a confusion matrix where entry(i, j) gives the number of documents in cluster i that belong to the true class j. For an objective evaluation measure, we use micro-averaged precision which was also used in [8].

We first demonstrate that Algorithm DITC_PLS (with prior and local search) is superior to Algorithms DITC_prior and DITC_LocalSearch.

Algorithm DITC_prior cures the problem of sparsity to some extent and its results are superior to DITC, for example, Table 1 shows the confusion matrices resulting from the two algorithms. An interesting option in DITC_prior is the starting value of α. Indeed, as Figures 1 show, the starting values of α can result in quite different values of mutual information preserved, $\frac{I(X;\hat{Y})}{I(X;Y)}$, and micro-averaged precision. The trend for DITC_prior in Figures 1 appears to be that larger starting values of α lead to better results (we observe this trend over other data sets too). This behavior is interesting and needs further study. Note that larger α values correspond to starting with "smeared" cluster distributions, or in other words, with high joint entropy values $H(X, \hat{Y})$.

However, the starting values of α cease to be an issue when we use DITC_PLS, which is seen to be "immune" to different starting values in Figure 1. Note that these figures validate our optimization criterion: there is a definite correlation between the mutual information preserved and micro-averaged precision, which was also observed in [8]. DITC_PLS is seen to be more stable than DITC_prior in addition to yielding higher quality results. Tables 2 and 3 further show that DITC_LocalSearch also yields better clustering than DITC_prior. However, DITC_PLS is computationally more efficient than DITC_LocalSearch since it has better starting partitions before invoking the slower local search procedure; hence DITC_PLS is our method of choice.

We now compare our Algorithm DITC_PLS with previously proposed information-theoretic algorithms. [9] proposed the use of an agglomerative algorithm that first clusters words, and then uses this clustered feature space to cluster documents using the same agglomerative information bottleneck method. More recently [8] improved the clustering results in [9] by using sequential information bottleneck (sIB). We implemented the sIB algorithm for purpose of comparison; since the sIB method starts with a random partition we ran 10 trials and report the average per-

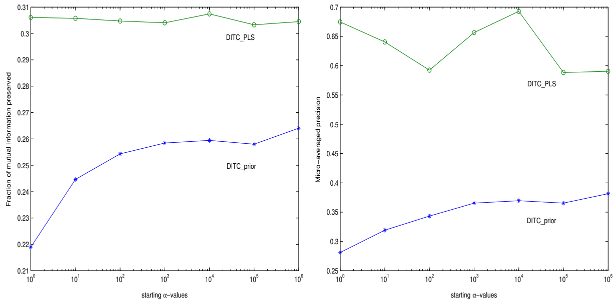

Figure 1. Mutual information preserved and Micro-averaged precision for DITC_prior with various starting α-values on Multi10

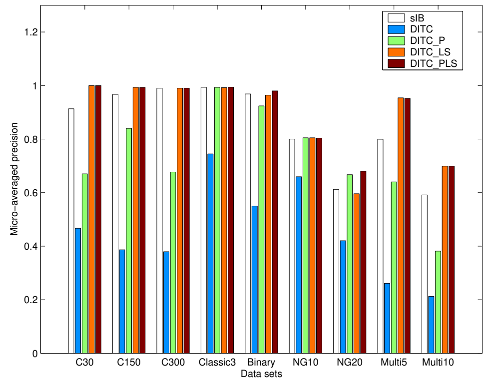

Figure 2. Micro-averaged precision results.

formance numbers in Figures 2 and 3, which also contain performance results for our algorithms (recall that our algorithm is deterministic due to the deterministic initialization shceme we use). Figures 2 and 3 again reveal the correlation between the preserved mutual information and micro-averaged precision. DITC_PLS is seen to be the best algorithm, and beats sIB on at least 3 of the data sets; for example, the average micro-averaged precision of sIB on Multi5 is .8 while DITC_PLS yields .95. Note that numbers for our sIB implementation are averages of 10 runs while the published numbers in [8] are the best among 15 restarts. Also, the Binary, Multi10 and Multi5 datasets in our work and in [8] are formed by a random sampling of the newsgroups, so the data sets are a bit different. However, the NG10 and NG20 data sets used by us and [8] are identical, and so are the micro-averaged precision values (see [8, Table 2]).

For the large data sets, CLASSIC3, NG10, NG20, DITC_prior gives results that are comparable to those with prior and local search, see Figures 2 and 3. This leads to considerable savings in time since DITC_prior is much faster than sIB as shown in Table 5.

Acknowledgments. This research was supported by NSF CAREER Award No. ACI-0093404 and Texas Advanced Research Program grant 003658-0431-2001.

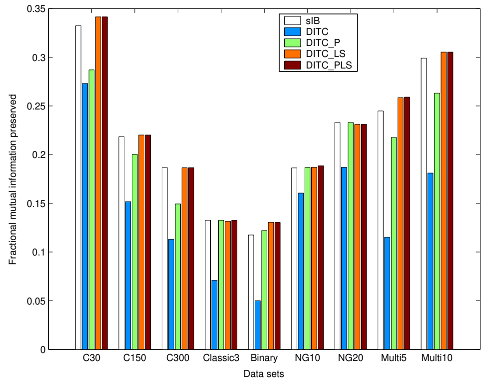

Figure 3. Fraction of mutual information preserved

Data set	sIB	DITC	DITC_prior
Classic3 (3893 documents)	95	1.35	1.67
NG10 (20000 documents)	2459	16.71	14.75
NG20 (20000 documents)	6244	35.87	29.92

Table 4. Computation time (in seconds) on large data sets (≥ 3000 documents)

References

[1] T. Cover and J. Thomas. *Elements of Information Theory*. John Wiley & Sons, New York, USA, 1991.

[2] I. S. Dhillon and Y. Guan. Information theoretic clustering of sparse co-occurrence data. Technical Report TR-03-39, Dept. of Computer Sciences, University of Texas, Sept 2003.

[3] I. S. Dhillon, Y. Guan, and J. Kogan. Iterative clustering of high dimensional text data augmented by local search. In *Proc. of IEEE International Conf. on Data Mining*, 2002.

[4] I. S. Dhillon, S. Mallela, and R. Kumar. A divisive information-theoretic feature clustering algorithm for text classification. *J. of Mach. Learning Res.*, 3:1265–1287, 2003.

[5] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. John Wiley & Sons, 2nd edition, 2000.

[6] K. Lang. News Weeder: Learning to filter netnews. In *Proc. 12th Int'l Conf. Machine Learning*, pages 331–339, 1995.

[7] K. Rose. Deterministic annealing for clustering, compression, classification, regression, and related optimization problems. *Proc. IEEE*, 86(11):2210–2239, 1998.

[8] N. Slonim, N. Friedman, and N. Tishby. Unsupervised document classification using sequential information maximization. In *ACM SIGIR*, 2002.

[9] N. Slonim and N. Tishby. Document clustering using word clusters via the information bottleneck method. In *ACM SIGIR*, pages 208–215, 2000.