Supervised clustering with the Dirichlet process

Hal Daumé III and **Daniel Marcu**
Information Sciences Institute
4676 Admiralty Way, Suite 1001
Marina del Rey, CA 90292
{hdaume,marcu}@isi.edu

Abstract

The task of learning to partition data into similar sets occurs frequently in many disciplines. We construct a Bayesian model for learning to partition from labeled data. Our model is based on the nonparametric Dirichlet process prior. Experimental results show that our model is able to outperform existing solutions on real world datasets.

1 Introduction

Problem definition.   In this paper, we explore the task of *supervised clustering*; like clustering, we are given data, a subset of $\mathcal{X}$, and must split it into like subsets. Unlike clustering, we also have as training data subsets of $\mathcal{X}$ and a desired partitions of these subsets. The learning task is to predict the correct partition of an unseen (typically disjoint) subset of $\mathcal{X}$. This problem has been investigated in many domains including identity uncertainty, record linkage, reference matching, coreference resolution and schema matching. We take our terminology and notation from the reference matching task (eg., the CiteSeer/ResearchIndex problem). Specifically, we assume that we are given a list of *citations* to *publications* and we need to identify which citations correspond to the same publication.

Prior work.   The most common solution to the supervised clustering problem is to build a binary classifier over pairs, and apply a heuristic algorithms to deal with non-transitivity [1, 2], but this separation of learning and clustering is not ideal. Another approach is to learn a distance metric to seed a clustering algorithm [3, 4]. One other recent work considers a generative approach in the style of relational learning [5].

2 A generative model for supervised clustering

Model.   We formulate the task in a generative Bayesian framework. We assume that there is an underlying set of possible publications $\{y_i\}$ and a set of ways to refer to each publication $\{t_j\}$, called reference types. In the reference matching task, one reference type might be good at identifying long, journal-style citations, another might be good at recognizing shorter conference-style citations, and another might be able to recognize the first initial, versus full name distinction. Of course, the intuitive interpretation of the reference types varies across different tasks.
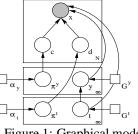

Figure 1: Graphical model

In our supervised clustering model (SCM), shown in Figure 1, we posit that a citation x_n is generated from a single publication y_{c_n} and a single reference type t_{d_n}. The c_n and d_n components are indicator variables for $\boldsymbol{y}$ and $\boldsymbol{t}$, respectively. Publications y_i are

chosen with probability π_i^y and types t_j are chosen with probability π_j^t. The πs are drawn according to their corresponding αs. Publications y is drawn from a base distribution G^y and reference types t are drawn from a base distribution G^t. Additionally, the publications $\boldsymbol{y}$ are conditioned on the chosen publications, $\boldsymbol{t}$. This is crucial: experiments based on a model without this dependence perform poorly. The reference types are the key to the learning component of our algorithm: they are global across the training and test data.

By making the πs multinomial r.v.s, the standard conjugate prior used would be a Dirichlet. However, since π is now a random *distribution*, as it is defined over infinitely many values, the Dirichlet distribution is no longer adequate, so instead we use a Dirichlet process (DP). The DP is, formally, a measure over measures. F^p is a DP if p is a finite measure over $\mathcal{B}$ and if $B_1, \ldots, B_k$ is a finite partition of $\mathcal{B}$ then $\langle F^p(B_1), \ldots, F^p(B_k) \rangle$ is distribution Dirichlet with parameter $p(B_1), \ldots, p(B_k)$ (under mild technical restrictions).

One can understand our model as an extension of the standard (Bayesian) naïve Bayes classifier. If the number of publications K were known and all publications appeared in the training data, this would be multiclass classification and a Dirichlet prior could be placed on the (finite) π. By removing the second assumption, we would prefer to use a symmetric Dirichlet distribution and finally to relax the assumption that K is known, we take the limit as $K \to \infty$, which corresponds to the DP [6].

The DP can be understood in terms of Pòlya Urns [7]: We start with an urn containing a black ball. We draw a ball from the urn: if it is black, we add a ball of a novel color to the urn and replace the black ball; if it is not black, we replace it along with a new ball of the same color. This distribution corresponds to samples from a DP with $\alpha = 1$.

Model Parameterization. As usual, we must make assumptions about the distributions underlying the data generating process. Here, we assume that the x_ns are normally distributed with mean y_{c_n} and precision (i.e., inverse variance) t_{d_n}. In the terminology of reference matching, we view all citations as located somewhere in space around the publication mean. We make G^p a normal distribution to preserve conjugacy. For computational reasons, we will assume that the t matrices are diagonal, thus enabling us to make G^t a gamma distribution. For the parameterization of $p(\boldsymbol{y}|\boldsymbol{t})$, to preserve generalization ability, we wish to ascribe no meaning to actual values of the $\boldsymbol{y}$s, so we define the conditional probability of the publications x under the reference types t in terms of their relative distances to each other, approximated as $\prod_{j=1}^{J-1} \prod_{j'=j+1}^{J} \mathcal{G}am(||y_j - y_{j'}||_t^{-1}; 1, 1)$.

Inference. The full Bayesian approach to inference would simultaneously consider both training data and testing data, integrating out the reference types to find a MAP solution to the cluster labels c_i for the test data. This involves integrating out the types plate and the representation of the publications. This approach has the disadvantage that it is computationally costly, especially when prediction is done for many data sets. An alternative is to split inference into a training phase and a prediction phase. The training phase samples from the posterior distribution of the reference types, conditioned on the training data (for which the $\boldsymbol{c}$ are observed). During prediction, the training data is ignored and the sample reference types are used to replace the bottom plate and the $\boldsymbol{c}$ variables are estimated.

Prediction. The training phase (discussed below) will leave us with a finite number of finite samples for reference types. Thus, we may consider that the t plate has been reduced to a finite size. Furthermore, the parameters of G^y have been estimated. Given the assumption on the form of $p(x_n|y_{c_n}, t_{d_n})$, prediction becomes a mixture of Gaussians problem, where each Gaussian can have one of J possible diagonal precision matrices, with an unknown (or infinite) number of means.

The algorithm we use is Algorithm 2 from [6], and involves resampling each indicator variable according to its conditional distribution, given the rest of the variables. Then, the y values (essentially the means) are drawn according to their posterior. The draws for the

indicator variables are according to:

$$c_n = c_j \mid \boldsymbol{c}_{-i} \sim \delta_{\boldsymbol{c}_{-i},c_j} \mathcal{N}or(\boldsymbol{x}_n; \boldsymbol{y}_{c_j}, \boldsymbol{t})$$
$$c_n \neq c_j \mid \boldsymbol{c}_{-i} \sim \alpha^y \int \mathrm{d}\mathcal{N}or(0, \sigma_0) \mathcal{N}or(\boldsymbol{x}_n; \boldsymbol{y}_{c_j}, \boldsymbol{t}) \tag{1}$$

Training. The general technique for training this model is identical to that of prediction, though the algorithm differs slightly due to the the distribution placed on the citations. The distribution for the model is broken down into two cases as before, one for shared reference types and one for new/unique reference types:

$$d_i = d \mid \boldsymbol{d}_{-i} \sim \delta_{\boldsymbol{d}_{-i},d} \prod_{j<j'} \mathcal{G}am(||y_j - y_{j'}||_t^{-1}) \mathcal{N}or(\boldsymbol{x}; \boldsymbol{y}, t)$$
$$d_i \neq d_j \mid \boldsymbol{d}_{-i} \sim \alpha^t \int \mathrm{d}\mathcal{G}am(a, b) \prod_{j<j'} \mathcal{G}am(||y_j - y_{j'}||_t^{-1}) \mathcal{N}or(\boldsymbol{x}; \boldsymbol{y}, t) \tag{2}$$

Unfortunately, these distributions are no longer conjugate, so we can neither efficiently sample from them nor compute the integral in closed form. Thus, we must use a different sampling algorithm from that described previously. The changes we make are few: Instead of computing the marginal for each point under the integral, we approximate this integral by a Student's t-distribution, which arises when one does not consider the effect of the reference types t on the publications y. Then, in order to rectify this approximation, during the initial sampling process, we sample R many extra reference types from the underlying distribution $G^t = \mathcal{G}am(a, b)$; in our experiments, $R = 6$. α is estimated as described in [8]. We run $10k$ iterations of burnin, and take $1k$ samples at a spacing of 100 in all experiments.

3 Results

Metrics. The standard metric used in the clustering literature, when a gold-standard clustering is available, is the Rand index, which views clustering as a binary classification problem; its value is the number of correct decisions made, divided by the total number of decisions. This metric unfortunately obscures many aspects of the problem, so we also report precision, recall and F-score, as well as a variant of the *cluster edit distance* (*CED*) metric, which counts the number of moves, merges and creates needed to transform the hypothesis clustering into the gold standard. Our variant, the normalized edit score (*NES*) is $NES(g, h) = 1 - (CED(g, h) + CED(h, g))/(2N)$, and falls between 0 (bad) and 1 (good). Other metrics are discussed in [9], but we have not experimented with these.

Baseline systems. For now, we evaluate against three baseline systems; in the future, this list will likely grow to include more recent methodologies proposed in the literature. The first two, which are straw men, simply either put all the elements in their own cluster (FINE) or put all the elements in a single cluster (COARSE). The third baseline system (SVM) is an SVM using an RBF kernel, tuned by cross-validation; clustering is done using the technique [1], tuned through more cross-validation. In addition, we also compare our system with a "partially trained" (PT-SCM) version of our system, where only one reference type is used (identity matrix), but α_y is estimated from the training data.

Digits data. We apply our system to the USPS handwritten digits dataset by using digits $\{1, 3, 5, 8, 9\}$ as testing data and the other digits as training data; with the task of identifying identical numbers (subsets were randomly selected). For brevity, the full results are omitted, but a selection are shown in Table 1. These results show that the trained SCM model consistently outper-

Table 1: Digits results.

# Model	RI	P	R	F	NES
COARSE	.116	.116	1.00	.207	.176
FINE	.885	.000	.000	.000	.053
PT-SCM	.886	.970	.016	.031	.000
2 SVM	.911	.802	.304	.441	.394
SCM	.893	.542	.501	.521	.476
5 SVM	.921	.730	.497	.592	.537
SCM	.937	.622	.658	.639	.618

forms the partially trained model, showing that the model is able to learn valuable information from the training data. Furthermore, our model outperforms the SVM-based system,

according to the NES metric in all cases. It also achieves higher RI and F measures on most training sizes.

Reference matching. One advantage of our parameterization is that all formulae depend only on pairwise distances, means distances, and sample variance of any set of points. Given two sets of points $\{a_i\}$ and $\{b_j\}$ we can compute the distance between the means as $\frac{1}{IJ}\sum_i \sum_j ||a_i - b_j||^2 - \frac{1}{I^2}\sum_{i<i'} ||a_i - a_{i'}||^2 - \frac{1}{J^2}\sum_{j<j'} ||b_b - b_{j'}||^2$. This derivation assumes that pairwise distances are calculated in Euclidean space. To deal with non-Euclidean distances, one could first *embed* the data into Euclidean space using a known algorithm, then compute distances in this space. Such an embedding can only lose information, so instead we advocate the direct use of any metric, Euclidean or not. (Some care must be taken: the Gaussian assumption makes no sense in the case of discrete values; however, in this case, G^y can be replaced with a Beta or Dirichlet distribution.) Generalization to the other relevant cases and multidimensional inputs is straightforward.

We evaluate on the Cora reference matching data [10]. This data consists of 1916 citations from 121 publications by M. Kearns, R. Schapire and Y. Freund. This data is noisy: the labeling of the fields has been done automatically and there are many errors. We treat the labeled data for two of these authors as

Table 2: Reference matching results.

Model	RI	P	R	F	NES
COARSE	.118	.118	1.00	.568	.010
FINE	.882	.000	.000	.000	.000
PT-SCM	.782	.141	.094	.113	.097
SVM	.936	.714	.616	.686	.721
SCM	.977	.779	.884	.828	.749

training data, using the third author as testing data. As features, we use several string edit distance computations (on publication names, primary author names, full names and conference names) and Euclidean distance between publication years, conference publication count and number of coauthors. The results are shown in Table 2, averaged over the three runs. In this data, our system outperforms any of the other approaches we compare against on all metrics. We cannot compare directly to [10] because their test data is not available.

4 Discussion

We have presented a graphical model framework for the supervised clustering task, based on the Dirichlet process prior, using a Gaussian parameterization. We have applied our model to artificial problems on naturally occurring data and to a full-fledged reference matching problem. On all data, our model has consistently outperformed all baselines and competing approaches. In the future, we wish to apply this model to more problems, such as identity uncertainty, explore other features and parameterizations, and experiment with different evaluation criteria.

References

[1] W. Cohen and J. Richman. Learning to match and cluster large high-dimensional data sets for data integration. In *KDD*, 2002.

[2] A. Bar-Hillel and D. Weinshall. Learning with equivalence constraints and the relation to multiclass learning. In *COLT*, 2003.

[3] A. Bar-Hillel, T. Hertz, N. Shental, and D. Weinshall. Learning distance functions using equivalence relations. In *ICML*, 2003.

[4] E.P. Xing, A.Y. Ng, M.I. Jordan, and S. Russel. Distance metric learning, with application to clustering with side-information. In *NIPS 15*, 2003.

[5] H. Pasula, B. Marthi, B. Milch, S. Russell and I. Shpitser. Identity uncertainty and citation matching. In *NIPS 15*, 2003.

[6] R. M. Neal. Markov chain sampling methods for Dirichlet process mixture models. Technical Report 9815, University of Toronto. 1998.

[7] D. Blackwell and J. B. MacQueen. Ferguson distributions via Pòlya urn schemes. *The Annals of Statistics*, 1(2):353–355, March 1973.

[8] M. D. Escobar. Estimating normal means with a Dirichlet process prior. *JASA*, 89(425):268–277, March 1994.

[9] M. Meila. Comparing clusterings. In *COLT*, 2003.

[10] A. McCallum, K. Nigam, and L. Ungar. Efficient clustering of high-dimensional data sets with application to reference matching. In *KDD*, 2000.