A Nonparametric Bayesian Model for Discovering Multiple Hierarchical Clusterings

Yue Guan **Jennifer Dy**
Northeastern University

Abstract

In many domains, humans organize objects hierarchically; thus, the wide usage of hierarchical clustering algorithms. Typical hierarchical clustering algorithms produce a single hierarchy from the data. However, data is often multi-faceted and focusing on different features may lead to different hierarchical groupings, which may be interesting for different purposes. In this paper, we develop an algorithm for automatically finding multiple simultaneous hierarchical solutions. We introduce a probabilistic nonparametric Bayesian model that can discover several possible hierarchical clustering solutions and the feature subset views that generated each cluster hierarchy simultaneously.

1 Introduction

Hierarchical clustering is an important tool for exploring and finding structure from data. Clustering is the process of grouping similar instances together. Clustering can be either partitional or hierarchical. Partitional clustering finds one grouping of the data; whereas, hierarchical clustering provides several levels of partitioning of varying coarseness where samples belonging together in lower levels remain together in higher levels of the hierarchy. A clustering hierarchy can be built either bottom-up (agglomerative) or top-down (divisive).

Typical clustering algorithms find one partitioning of the data [14]. However, data may be multi-faceted in nature; meaning, there may be several possible interpretations for a single data set. Sample instances can be grouped together in several different ways for different purposes. For example, for the same data, what is interesting to physicians might be different from what insurance agencies are interested in; face images of people can be grouped based either on their pose or based on the identity of the person; web-pages from universities may have one clustering interpretation based on which university the web-page comes from or based on the type of the web-page's owner: {*faculty, student, staff*}, or based on the web-page's topic: {*physics, math, engineering, computer science*}; playing cards may be grouped based on suit or based on value. Oftentimes during exploratory analysis, the clustering solution found is not what the analyst or scientist is looking for, but one of the other alternative possible groupings in the data. Sometimes, the analyst may simply want to discover all possible different clustering interpretations of the data.

Algorithms for finding clusters and cluster hierarchies are generally dependent on similarity measures, which in turn are dependent on the features defining similarity. Many clustering algorithms utilize all the features, however not all features are important – some may be irrelevant and some may be redundant. Thus, the need for feature selection for clustering [10, 18]. Since standard clustering algorithms find one partitioning of the data, feature selection algorithms typically choose one subset of features relevant for finding the single clustering solution as well. Why choose one feature subset and clustering interpretation, when all the alternative feature subset views might be interesting to the user? The features that are irrelevant to one clustering interpretation, may be relevant to another clustering solution.

Our goal in this paper is to discover multiple hierarchical clustering solutions in different feature subset views simultaneously. We introduce a probabilistic nonparametric model that assumes each feature comes from an infinite mixture of views generated from a Chinese restaurant process (CRP) [21], where each view generates a hierarchy of clusters of the data with a prior based on Kingman's coalescent [17], for solving this problem. This nonparametric Bayesian model allows

Technical Report, 2011, ECE, Northeastern University

us not only to learn the multiple hierarchical clusterings and feature views but also allows us to automatically learn the number of views.

2 Related Work

There exist only limited work that addresses finding alternative or multiple clustering interpretations. [12], [1] and [22] find an alternative clustering given an existing clustering solution. Gondek and Hofmann [12] find an alternative non-redundant clustering by a conditional information bottleneck approach [5]. Bae and Bailey [1] utilize cannot-link constraints imposed on data points belonging to the same group by a previous clustering and agglomerative clustering in order to find an alternative non-redundant clustering. Qi et al. [22] find an alternative projection of the original data that minimizes the Kullback-Leibler divergence between the distribution of the original space and the projection subject to the constraint that sum squared error between samples in the projected space with the means of the clusters they belong to is smaller than a pre-specified threshold. Their method approximates the clusters from mixtures of Gaussians with components sharing the same covariance matrix. These three only address finding one alternative clustering, but for complex data there may be more than one alternative clustering interpretation. Cui et al. [7] find multiple alternative views by clustering in the subspace orthogonal to the clustering solutions found in previous iterations. This approach discovers several alternative clustering solutions by iteratively finding one alternative solution given the previously found clustering solutions. All these methods find alternative clustering solutions sequentially (or iteratively). Another general way for discovering multiple solutions is by finding them simultaneously. Meta clustering in [4] generates a diverse set of clustering solutions by either random initialization or random feature weighting, which are then meta clustered using an agglomerative clustering based on a Rand index for measuring similarity between pairwise clustering solutions. Jain et al. [15] learn two disparate clusterings by minimizing two k-means type sum-squared error objective for the two clustering solutions while at the same time minimizing the correlation between these two clusterings. Similar to [7, 4, 15], the approach we propose here discovers multiple clustering solutions; and, like [4, 15], our approach finds these solutions simultaneously. However, unlike all these methods, we provide a probabilistic generative nonparametric model. Among the benefits of probabilistic models are that they can handle missing data, answer inference questions (such as predict on new data), and automatically determine the number of views and clusters in each view. There are two recent closely related workshop papers that provide nonparametric Bayesian models for finding multiple partitionings: one utilizes the Chinese restaurant process (CRP) [19] and the other utilizes a multiple clustering stick-breaking construction and variational inference [13]. *In this work, unlike all previous methods, besides providing a probabilistic model, we also learn multiple hierarchical cluster structures.* While there are nonparametric Bayesian models present a hierarchical structure like nested CRP [3], we bring a new aspect that multiple instead of one interesting hierarchical structure may be found.

3 Nonparametric Multiple Hierarchical Clustering Model

Assume data $X \in \mathbb{R}^{N \times D}$, where N is the number of samples and D the number of features. We use the term *view* to refer to a subset of the D features. Our goal is to learn multiple hierarchical clustering solutions and the feature views that define their similarity. We make the following modeling assumptions to solve this problem. We assume that the features in each view are disjoint and the clusters in each view have a hierarchical structure and the cluster hierarchies are generated independently from each other. In our model, the samples in different views are independently partitioned given the view; moreover, samples belonging to the same clusters in one view can belong to different clusters in other views. In this initial work on a probabilistic model for multiple hierarchical clustering, we assume that the features in each view are disjoint; in future work, we will explore models that can allow sharing of features between views. There are several possible ways to cluster the data. We do not want to show an exhaustive list of all possible hierarchical interpretations, because they may overwhelm the data analyst. Rather, we would like to show hierarchical clustering solutions that are both of good quality and non-redundant from each other. Previous models for multiple clustering explicitly enforce non-redundancy, orthogonality or disparity among clustering solutions [12, 1, 7, 22, 15]. Our probabilistic model handles redundancy implicitly, since redundant clusterings offer no probabilistic modelling advantage and are penalized under the prior which assumes that each view is clustered independently.

It is often difficult to find exact parametric models for complex data. Bayesian nonparametric models avoid restricted functional forms and allow the complexity and accuracy of the models to grow with the data.

Let $X = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N]^T = [\mathbf{f}_1, \mathbf{f}_2, \ldots, \mathbf{f}_D]$, where $\mathbf{x}_i \in \mathbb{R}^D$ are the samples, the columns $\mathbf{f}_j \in \mathbb{R}^N$ are the features, and $(\cdot)^T$ is the transpose of a matrix. We

design a nonparametric prior model that can generate a *multiple hierarchical clustering latent structure* for data X. Let Y be a matrix of latent variables representing the partitioning of features into different views, where each element, $y_{j,v} = 1$ if feature $\mathbf{f}_j$ belongs to view v and $y_{j,v} = 0$ otherwise. And, let π_v be a matrix of latent variables representing the hierarchical partitioning of samples for view v. We model the latent variable Y representing the partitioning of the features into different views coming from a Chinese restaurant process (CRP) [21]. Then, we model each π_v as a latent hierarchical tree with a Kingman's coalescent prior [17]. Given this latent multiple hierarchical clustering structure, we then generate our observation variables X. We describe the generative process next.

3.1 Overall Generative Process

The generative process for our nonparametric multiple hierarchical clustering model is summarized below. Details regarding each component of the generative process is described in the next subsections 3.2 to **??**.

1. Generate the latent matrix Y representing the partitioning of features into different views from a CRP with concentration hyperparameter α: $Y \sim CRP(\alpha)$.
2. For each view v, generate hierarchical clusters π_v from a Kingman's coalescent model.
3. For each view v and hierarchy π_v, sample parameters $\boldsymbol{\theta}_v$ from an appropriate prior distribution.
4. For each view v, generate our observation variables X given latent variables Y and π_v. In each view v, we draw the value of the features in view v for sample i: $\mathbf{x}_{i,v} \sim p(\mathbf{x}_{i,v}|\pi_v, \boldsymbol{\theta}_v)$, where $\mathbf{x}_{i,v} = (x_{i,j} : y_{j,v} = 1)$ is the vector of features in view v, and $\boldsymbol{\theta}_v$ are the parameters of our observation for the coalescent tree in view v. We provide two alternative observation probability models, $p(\mathbf{x}_{i,v}|\pi_v, \boldsymbol{\theta}_v)$, one based on a Gaussian transition probability model and another on a multinomial model. The Gaussian model provides us with a model for real-valued data and the multinomial model for discrete data.

3.2 The Chinese Restaurant Process

The Chinese Restaurant Process [21] defines the distributions on partitions in a Dirichlet process mixture whose name is inspired by the seemingly infinite tables in San Francisco's Chinatown restaurants [21]. Here, we apply CRP on the features. The infinite set of tables are analogous to our views and the customers, our features. The jth customer sits at table v with probability, D_v, proportional to the number of customers (features) already seated at table, v, or will sit at a new table (view) with probability proportional to the scalar concentration hyperparameter α. More concretely, this is represented by the following distribution:

$$\begin{aligned} & p(y_{j+1} = v | y_1, \ldots, y_j, \alpha) \\ = \; & \tfrac{1}{\alpha+j}\left(\textstyle\sum_{v=1}^{V} D_v \delta(y_{j+1}, v) + \alpha\delta(y_{j+1}, v_{new})\right) \quad (1) \end{aligned}$$

where v_{new} represents a new table (view), V is the number of unique non-empty views which may be infinite. The CRP is an exchangeable distribution on partitions; meaning that, the distribution is invariant to the order in which customers (features) are assigned to tables (views). Exchangeability of the CRP follows from De Finetti's Theorem [8]. By serving each table a separate dish θ_v, CRP provides a representation for a Dirichlet process mixture [25].

3.3 Kingman's Coalescent Hierarchical Tree

After assigning each feature to the view it belongs to, we are now ready to generate the cluster hierarchy within each view. Kingman's coalescent model serves as a nonparametric prior for a hierarchical latent tree structure [17]. It is used to describe the common genealogy (ancestral) tree of a set of N individuals, as $N \to \infty$. Let's say we have $[N_v] = \{x_{1,v}, \ldots, x_{N,v}\}$ individuals (samples in each view, in our case) at the present time, $t = 0$, where $\mathbf{x}_{i,v} = (x_{i,j} : y_{j,v} = 1)$ is the vector of features in view v. Going back to $t = -\infty$ will provide us with the root of the ancestral tree. We assume that each child has only one parent, thus forming a tree structure. At time $t < 0$, there are m_v $(1 \le m_v \le N)$ ancestors alive at view v; and, at a given time t, $\pi_v(t) = \{p_{1,v}, \ldots, p_{m,v}\}$ forms a partition of $[N_v]$. Here $p_{i,v}$ represents the descendants (samples that belong to partition $p_{i,v}$ at view v). π_v completely defines the ancestral tree (cluster hierarchy at view v).

The Kingman's coalescent is a distribution over hierarchical partitionings, such as π_v. It is a Markov process that builds the hierarchy in an agglomerative (bottom-up) fashion, starting at time $t = 0$ with N samples and evolves backwards in time, merging lineages (clusters) until only one parent root (one cluster) is left. Let $p_{li,v}$ and $p_{ri,v}$ be the ith pair of clusters to coalesce or merge at time t_i, $t_0 = 0 > t_1 > \cdots > t_{N-1}$ be the coelescent times and $\delta_{i,v} = t_{i-1} - t_i$ be the duration of time between two merge events. Every pair of clusters merges independently with an exponential rate of 1. The first pair among m_v clusters merges with a rate equal to m_v choose 2, $\binom{m_v}{2}$, and the duration is distributed from an

exponential distribution as $\delta_{i,v} \sim \text{Exp}(\binom{N-i+1}{2})$. The probability of π_v is thus: $p(\pi_v) = \prod_{i=1}^{N-1} \binom{N-i+1}{2} \exp\left(-\binom{N-i+1}{2}\delta_{i,v}\right) / \binom{N-i+1}{2} = \prod_{i=1}^{N-1} \exp\left(-\binom{N-i+1}{2}\delta_{i,v}\right)$. This N-coalescent model is infinitely exchangeable. Given an N-coalescent model, the hierarchical tree of any m_v individuals that are alive at time $t \leq 0$ is a draw from an m_v-coalescent model. Letting $N \to \infty$, there is an infinite population such that the marginal distribution of any N individuals (samples) is an N-coalescent.

3.4 Inference

In this section, we present the inference method we apply to our multi-coalescents model. We divide the inference problem into two parts: the first part utilizes Gibbs sampling [20] for the Chinese restaurant process, and the second part applies message passing and a greedy algorithm for the coalescent model in each view. [24] provide several strategies for inference of a coalescent tree, including a sequential Monte Carlo and greedy approaches. Here, we apply the simpler greedy scheme that maximizes the marginal probability detailed below.

Let y_j be the view indicator equal to v if $y_{j,v} = 1$, V be the total number of non-empty and unique views, $\mathbf{\Phi}$ be the set of all parameters for each view including the coalescent structure π_v, and λ be the set of all hyperparameters for $\mathbf{\Phi}$. We apply Gibbs sampling to sample our latent variable Y generated from a CRP as follows:

$$p(y_j = v|X) = \begin{array}{l} \frac{D_v}{D-1+\alpha} \int_{\mathbf{\Phi}} p(\mathbf{\Phi}_v, X_v|\lambda) d\mathbf{\Phi}, \text{ if } v \leq V \\ \frac{\alpha}{D-1+\alpha} \int_{\mathbf{\Phi}} p(\mathbf{\Phi}_{V+1}, X_{V+1}|\lambda) d\mathbf{\Phi}, \\ \quad \text{if } v = V+1 \end{array} \tag{2}$$

where $X_v = [x_{1,v}, \ldots, x_{N,v}]^T$ is the data set of N samples with features in view v. For each possible value of v, we have to resample π.

For each view, we present how to find the likelihood score of the coalescent using message passing and a greedy algorithm next. Let i be a coalescent point, meaning the point when $p_{li,v}$ and $p_{ri,v}$ coalesce or merge at time t_i forming the cluster $p_{i,v} = p_{li,v} \cup p_{ri,v}$. Let $z_{i,v}$ be a latent variable taking on the value of the Markov process at p_i. If $z_{i,v}$ is a leaf node, then it's observed sample is $x_{i,v}$. The probability of a coalescent structure is: $p(\pi_v) = \prod_{i=1}^{N-1} \exp\left(\binom{N-i+1}{2}\delta_{i,v}\right)$. The joint distribution follows the conditional independence provided by the tree structure. Thus, given the coalescents, we have the joint probability of the observed samples and internal nodes as: $p(X_v, z_v|\pi) = p(z_{0,v}) \prod_{i=1}^{N-1} p(z_{l,v}|z_{i,v}) p(z_{r,v}|z_{i,v})$. Note that the likelihood function is further defined in following section with two observation model. The probability of the root node at time $t = -\infty$ is $p(z_{0,v}|\gamma)$, where γ is a hyperparameter.

The goal here is to maximize the marginal distribution of the observed samples over the parameters. The marginal distribution is: $p(X_v) = \int p(X_v, z_v|\pi_v) p(\pi_v) dz_v d\pi_v$. In the greedy scheme, we maximize at every step of a message passing algorithm applied to the hierarchy in an agglomerative fashion (from the bottom up). The message at node i in view v, $M(z_{i,v})$ is:

$$\begin{aligned} M(z_{i,v}) &= W_{i,v}^{-1} \int p(z_{li,v}|z_{i,v}) M(z_{li,v}) dz_{li,v} \\ &\quad \int p(z_{ri,v}|z_{i,v}) M(z_{ri,v}) dz_{ri,v} \end{aligned} \tag{3}$$

where $W_{i,v}$ is the normalization constant. When the child node is the observed sample $x_{li,v}$, then we replace $\int p(z_{li,v}|z_{i,v}) M(z_{li,v}) dz_{li,v}$ with $p(x_{li,v}|z_{i,v})$. Similarly, if the right child node is the observed sample $x_{ri,v}$, we replace with $p(x_{ri,v}|z_{i,v})$. We normalize message $M(z_{i,v})$ to a valid probability with the following normalization constant:

$$W_{i,v} = \int \int p(z_{0,v}) p(z_{i,v}|z_{0,v}) M(z_{i,v}) dz_{0,v} dz_{i,v} \tag{4}$$

The joint distribution of the observed samples given the partition tree in each view v is:

$$p(x_v|\pi_v) = W_{0,v} \prod_{i=1}^{N-1} W_{i,v} \tag{5}$$

The joint distribution for the samples X_v and coalescent hierarchy π_v in each view v is:

$$p(X_v, \pi_v) = W_{0,v} \prod_{i=1}^{N-1} \exp(-\binom{N-i+1}{2}\delta_{i,v}) W_{i,v} \tag{6}$$

We find the tree structure that maximizes Equation 6. In our greedy scheme, we maximize each product term one by one from time $t = 0$ to $t = -\infty$ (in an agglomerative fashion); i.e., at each step, we pick the two child nodes that provide us with the largest $W_{i,v}$ to merge, then calculate the new message $M(z_{i,v})$ and the optimal time interval $\delta_{i,v}$. Because we apply a greedy scheme for the cluster hierarchy, instead of the integral in Equation 2, we simply use the marginalized likelihood score for each coalescent structure π_v under view v as provided in Equation 6 to approximate the integration.

Gaussian Observation Model: Suppose we have a Gaussian transition probability model (the probability of a child node given the parent node) in our coalescent tree in each view v, $p(z_{ci,v}|z_{i,v}) = N(z_{i,v}, \delta_{i,v}\Sigma_v)$,

where $z_{ci,v}$ is a child node and $z_{i,v}$ is the corresponding parent node, Σ_v is a positive definite covariance matrix. Then, we also have a Gaussian distributed message function $M(z_{i,v}) \sim N(\hat{z}_{i,v}, \hat{\delta}_{i,v}\Sigma_v)$. Following Equation 4 for the normalization constant $W_{i,v}$: $W_{i,v} = \left|2\pi\hat{\Sigma}_{i,v}\right|^{-\frac{1}{2}} \exp(-\frac{1}{2}\|\hat{z}_{li,v} - \hat{z}_{ri,v}\|^2_{\hat{\Sigma}_{i,v}})$ with covariance matrix: $\hat{\Sigma}_{i,v} = \Sigma(\hat{\delta}_{li,v} + \hat{\delta}_{ri,v} + t_{li,v} + t_{ri,v} - 2t_{i,v})$, where $\|x_v\|^2_\Phi = x_v^T\Phi^{-1}x_v$ is the Mahanalobis norm, $t_{li,v}$ and $t_{ri,v}$ are the times the left and right child node coalesce in view v respectively. Similarly, $\hat{\delta}_{li,v}$ is the estimated duration time for the left child node to coalesce from the previous time step. And the parameters for the message at the new merged node are:

$$\hat{\delta}_{i,v} = ((\hat{\delta}_{li,v} + t_{li,v} - t_{i,v})^{-1} + (\hat{\delta}_{ri,v} + t_{ri,v} - t_{i,v})^{-1})^{-1}$$

$$\hat{z}_{i,v} = \left(\frac{\hat{z}_{li,v}}{\hat{\delta}_{li,v} + t_{li,v} - t_{i,v}} + \frac{\hat{z}_{ri,v}}{\hat{\delta}_{ri,v} + t_{ri,v} - t_{i,v}}\right)\hat{\delta}_{i,v}$$

We apply the inverse Wishart distribution as a prior for our covariance parameter Σ_v. Parameter Σ_v form the observation parameter $\boldsymbol{\theta}_v$ in Section 3.1.

Multinomial Observation Model: Suppose that each feature entry $z_{j,v}$ of the vector takes on K values and evolving independently. The transition rate matrix $Q_{j,v} = \rho_{j,v}(q_{j,v}^T\mathbf{1}_K - I_K)$, where $\rho_{j,v}$ is the rate of evolution for entry j and $q_{j,v}$ is the equilibrium distribution, $\mathbf{1}_K$ is a column vector of length K with entry 1 and I_K is an identity matrix of length K. Then the transition probability matrix for entry j in time interval $\delta_{i,v}$ is $e^{Q_{j,v}\delta_{i,v}}$. Observation parameter $\boldsymbol{\theta}_v$ in Section 3.1 comprises of both $\rho_{j,v}$ and $q_{j,v}$ for all feature j in view v. Let $M^j_{p_{i,v}} = [M^{j1}_{p_{i,v}}, \ldots, M^{jK}_{p_{i,v}}]^T$ represent the message for entry j from p_i to its parent. Let us normalize such that $q_{j,v} \cdot M^j_{p_{i,v}} = 1$. The message and normalization constants are: $M^j_{p_{i,v}} = (1 - e^{\rho_{j,v}(t_{i,v}-t_{li,v})}(1 - M^j_{p_{li,v}}))(1 - e^{\rho_{j,v}(t_{i,v}-t_{ri,v})}(1 - M^j_{p_{ri,v}}))/W^j_{p_{i,v}}(\mathbf{x}_v, \nu_{i,v})$ and $W^j_{p_{i,v}}(\mathbf{x}_v, \nu_{i,v}) = 1 - e^{\rho_{j,v}(2t_{i,v}-t_{li,v}-t_{ri,v})}(1-\sum_{k=1}^K q_{jk,v}M^{jk}_{p_{li,v}}M^{jk}_{p_{ri,v}})$ where $\nu_{i,v} = \{\delta_{s,v}, p_{ls,v}, p_{rs,v} \text{ for } s \leq i\}$. We obtain the optimal duration, $\delta_{v,s}$, by Newton optimization.

4 Experiments

We performed experiments on both synthetic and real data to investigate whether our algorithm gives reasonable multiple hierarchical clustering results. To get a better understanding of our method, we first perform experiments on synthetic data. Then, we test our method on real-world data sets to check whether we can find meaningful clustering views that correspond to human labeling. We select data that have high-dimensionality and multiple possible partitioning interpretations. In particular, we test our method on digit images, face images, and text data. We apply the Gaussian model for the synthetic and two image datasets and the multinomial model for the text data. We compare our multi-coalescent tree model (MT) against a single coalescent tree model (Single) [24] and a single Dirichlet process mixture model (GP). In addition, we compare our approach to two other multiple clustering algorithms, orthogonal projection clustering (OPC) [7] and decorrelated k-means (DCKM) [15]. In orthogonal projection clustering, they first reduced the dimensionality by principal component analysis [16] (retaining 90% of the total variance). Then, instances are clustered in the principal component space by k-means clustering algorithm to find a dominant clustering. Because the means of clusters represent the clustering solution, data are projected to the subspace that is orthogonal to the subspace spanned by the means. In the orthogonal subspace, they use PCA followed by the clustering algorithm again to find an alternative clustering solution. This process is repeated until all the possible views are found. In de-correlated k-means [15], they simultaneously minimize the sum-squared errors in two clustering views and the correlation of the mean vectors and representative vectors between the two views. They apply gradient descent to find the clustering solutions. In de-correlated k-means, both views minimize sum-squared errors in all the original dimensions. The concentration parameter α is set to 10 for all experiments. The covariance matrix for the inverse Wishart distribution in the Gaussian observation model is set to identity I. For the multinomial case, the prior count is set to 1 as a unifrom Dirichlet prior.

To evaluate the effectiveness of the hierarchical structures discovered based on labeled classes, we report purity and subtree scores for each coalescent structure in every view. Purity score is obtained by randomly choosing a leaf from a class and uniformly randomly choosing another leaf in the same class, finding the minimum common subtree, then the score is the number of leaf nodes from the same class divided by the total number of leaf nodes. Another measure is the subtree score, which is the number of interior nodes with all leaves of the same class divided by the number of instances minus the number of classes. Both the purity and subtree scores are always between zero and one and larger values mean better match between the tree and the "true" label. Since existing multiple clustering methods only find flat partitions, we compare our result with these methods based on the normalized mutual information (NMI) [23] of the "true" labeling and the discovered clusters. For our approach, we evaluate at the hierarchy level equal to the known number

of clusters in each view. Let C represent the clustering results and L the labels, $NMI = \frac{MI(C,L)}{\sqrt{H(C)H(L)}}$, where $MI(C, L)$ is the mutual information between random variables C and L and $H(\cdot)$ is the entropy of $(\cdot)$. Note that in all our experiments, labeled information are not used for training. We only use them to measure the performance of our clustering algorithms. Higher NMI values mean the more similar the clustering results are with the labels; and it only reaches its maximum value of one when both clustering and labels are perfectly matched.

Results on Synthetic Data. We generated a synthetic dataset with three independent alternative labels/views to test whether or not our algorithm can deal with high dimensionality and more than two views. It consists of 60 features with 1000 instances. In each feature set $(F_{(1..20)}, F_{(21..40)}, F_{(41..60)})$, a mixture of five Gaussian components is generated. Table 1 reports our results on this data. Our multiple coalescent tree model (MT) found three views. Here, we report the scores of the views that correspond to the appropriate labeling (Label i). The single coalescent model (Single) only found one tree. The scores reflect the score for that tree on each of the true labeling. Note that our multiple coalescent model is able to find the cluster structures in all views with high ($> 93\%$) purity and subtree scores; whereas, a single coalescent model using all the features did not perform as well. The reasons for the single coalescent tree's bad performance are: firstly, it tries to find one cluster structure when there are multiple clustering solutions/views; secondly, the features for the other views misled the clustering results.

Table 1: Purity and Subtree Scores for Synthetic Data

	Label 1		Label 2		Label 3	
Score	MT	Single	MT	Single	MT	Single
Purity	**0.963**	0.548	**0.961**	0.534	**0.947**	0.533
Subtree	**0.976**	0.606	**0.976**	0.542	**0.937**	0.663

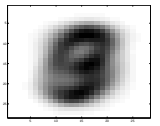
Results on Digits Data. Digits data from the UCI KDD repository [11] are hand-written digits labeled from 0 to 9. The original data set has $60,000$ samples and each image has 28 by 28 pixels. To reduce the running time, we sample 100 samples per digit class, giving us a total of $1,000$ samples. For this data, we first applied principal components analysis (PCA) [16] to extract appearance-based features [26] and reduce the dimensionality by keeping only the first 100 eigenvectors corresponding to the largest eigenvalues. The digits data has only one class label information provided based on the ten digits. However, hand-written digits may be grouped in other ways (for example, based on hand-writing style). Our multiple coalescent model found two views. In Table 2, we provide the purity and subtree scores for the hierarchical structure in each view found by our method and the scores for the single coalescent model compared to the known digit labeling. The table shows that view 1 matches to the known digit-class labels better than view 2. Furthermore, the results for our method with view 1 is better than the single model using all the features maybe because we also learn the appropriate features for clustering the digits. To provide us with a summary of the features selected by our algorithm in each view, we display the top two features (eigenvectors) corresponding to the largest variance in each view and show them as images, we call *eigenimages* in analogy to "eigenfaces" in [26] in Figures 1a and b. Note that the eigenimages in both views make sense – they represent structures that define digits. View 1 captures digits written straight, while view 2 captures the digits written slantly. In Figures 1c and d, we show the coalescent trees learned by our model in views 1 and 2 respectively. At each coalescent point, we also show the mean image for the samples belonging to that cluster. For clarity, we show the trees pruned where the leaves show ten clusters (hopefully, corresponding to the ten digits). The label of the leaf node in the pruned tree is determined by the majority class. The resulting hierarchy for view 1 is $(((3, 8), 5), (6, 0), (((9, (1, 7)), 2), 4))$ and for view 2 is $((((3, (1, 9)), 4), ((0, 8), 6)), ((2, 7), 5))$. Both results make sense. Note that in view 1, 1 and 7 are grouped together, while in view 2, 2 and 7 are grouped together. This makes sense, since 7 is both similar to 1 (long edge) and 2 (curve on top). Similarly for 3 and 8 in view 1 versus 0 and 8 in view 2.

In Table 3, we also report the NMI results of our approach (MT) and the single coalescent tree evaluated at the hierarchy level such that the number of clusters is equal to the "true" (based on known labels) number of classes for each view. We truncated our tree to be able to compare to other multiple clustering methods, OPC and DCKM, which only provide flat partitions. Each column with heading View i reports the NMI between the clusters in the discovered View i with the known label. The results show that our approach had the best NMI match with the digit class labels.

Table 2: Purity and Subtree Scores for the Digits Data

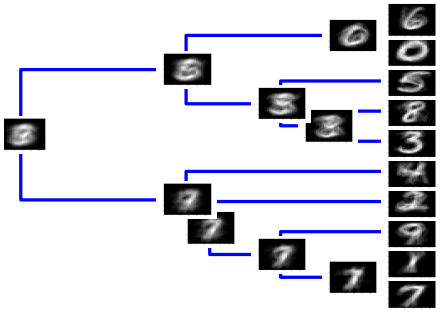
	Digit Label		
Score	View 1	View 2	Single
Purity	**0.538**	0.463	0.436
Subtree	**0.582**	0.514	0.578

Results on Face Data. The face data from the UCI KDD repository [2] has 640 samples with 960 pixels of human faces. This data set has four available labeling interpretations corresponding to identity (20 dif-

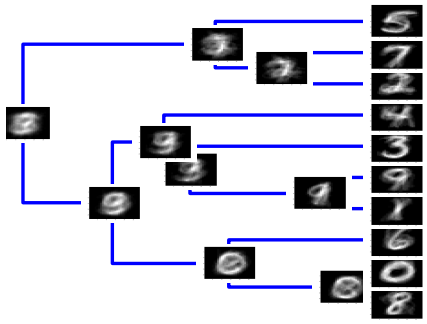


(a) 1st and 2nd eigenimages for view 1

(b) 1st and 2nd eigenimages for view 2



(c) Coalescent tree with mean images for view 1



(d) Coalescent tree with mean images for view 2

Figure 1: The 1st and 2nd eigenimages for views 1 (a) and 2 (b), and the coalescent tree with mean images at the coalescent points for views 1 (c) and 2 (d) learned by our method for the digits data.

Table 3: NMI Results on Digit Data

	Digit Label	
Method	View 1	View 2
MT	**0.347**	**0.358**
Single	0.261	NA
DCKM	0.332	0.316
OPC	0.326	0.318

ferent people), pose (straight, left, right, up), expression (neutral, happy, sad, angry) and w/o sunglasses. The two dominant labeling (high NMI for all methods) are that of identity and pose. To save space, we only show the results for these labeling. Similar to the digits data, we also pre-process the face image data using PCA. We kept only the first 100 eigenvectors corresponding to the largest eigenvalues. The first 100 eigenvectors retains a total of 99.24% of the overall variance. Table 4 reports the purity and subtree scores. The results show that the single tree found clustering structures which are a mixture of the identity and pose views. Our method is able to separate the two views (view 1 with identity and view 2 with pose) with purity and subtree scores higher than the single tree. In Table 5, we report the NMI results of our approach (MT) compared to other multiple clustering methods. Each column with heading View i reports the NMI between the clusters in the discovered View i with the known identity and pose labelings respectively. Note that in this table, all the methods have view 1 correspond to identity (higher NMI values compared to their respective view 2) and view 2 correspond to pose. All the multiple clustering approaches were able to reasonably discover both the identity and pose views. DCKM had the best NMI match with the identity label and our MT had the best NMI match with pose and comparable NMI score for the identity label with the other multiple clustering methods. Unlike the other methods, MT can also discover the features for each view. In Figures 2a and b, we display the first two eigenfaces (the eigenimages for faces) found by MT for views 1 and 2 respectively. Note that the eigenfaces in view 1 make sense. It captures the face and background information important for distinguishing identity. The eigenfaces in view 2 captures the information necessary for distinguishing pose.

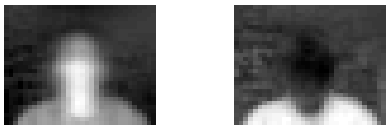
Table 4: Purity and Subtree Scores for the Face Data

	Identity label			Pose		
Score	View 1	View 2	Single	View 1	View 2	Single
Purity	**0.655**	0.441	0.485	0.562	**0.759**	0.512
Subtree	**0.751**	0.563	0.578	0.619	**0.787**	0.538

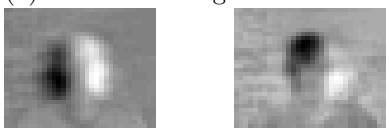
Table 5: NMI Results on the Face Data

	Identity		Pose	
Method	View 1	View 2	View 1	View 2
MT	0.649	0.175	0.157	**0.564**
Single	0.533	NA	0.334	NA
GP	0.563		0.03	
DCKM	**0.704**	0.281	0.127	0.402
OPC	0.673	0.219	0.147	0.384

Results on WebKB Text Data. WebKB data [6] is a webpage text data with two labeling interpretations: one based on four universities (Cornell University, University of Texas, Austin, University of Washington and University of Wisconsin, Madison) and the



(a) 1st and 2nd eigenfaces for view 1



(b) 1st and 2nd eigenfaces for view 2

Figure 2: The 1st and 2nd eigenfaces for views 1 (a) and 2 (b) on the face data.

other based on five owner-types (course, faculty, staff, project and student). We randomly choose 20 samples for each class in each labeling, giving us 400 data points. We pre-processed the data by removing rare words, stop words, and only keeping the words with large variances, giving us 200 word features. For the multinomial model, we truncate the largest count for each word to 20. We set the transition rate matrix and equilibrium distribution based on their maximum likelihood estimates. The purity and subtree scores are provided in Table 6. The results show that the single tree matches the labeling based on university better than based on owner-type and that our multiple coalescent tree have higher scores for both labeling interpretations as shown in bold. This shows that our multiple hierarchical clustering approach is more appropriate for a multi-faceted data, such as WebKB. In Table 7, we report the NMI results of our approach (MT) compared to other multiple clustering methods. Each column with heading View i reports the NMI between the clusters in the discovered View i with the known university and owner-type labelings respectively. Note that in this table, all the methods have view 1 correspond to university and view 2 correspond to owner-type. All the multiple clustering approaches where able to reasonably discover both the identity and pose views at comparable NMI values with DCKM performing slightly better. Note that DCKM and OPC have to be given the number of clusters to find. In addition to discovering multiple cluster hierarchies, MT can also discover the features for each view. The ten most frequent words selected by our method in view 1 are: *Cornell, finance, information, Washington, format, visual, define, Wisconsin, program, Madison.* The top words selected for view 2 are: *student, project, research, theorem, department, report, group, science, faculty, system.* Notice that our multiple coalescent model was able to select the relevant words: the words chosen for view 1 are appropriate for distinguishing different universities; and, the words for view 2 are appropriate for grouping owner-types.

Table 6: Purity and Subtree Scores for the WebKB Data

	University			Owner		
Score	View 1	View 2	Single	View 1	View 2	Single
Purity	**0.381**	0.198	0.265	0.107	**0.212**	0.143
Subtree	**0.390**	0.246	0.257	0.150	**0.291**	0.181

Table 7: NMI Results on WebKB Data

	University		Owner	
Method	View 1	View 2	View 1	View 2
MT	0.542	0.357	0.279	0.459
Single	0.352	NA	0.214	NA
GP	0.392		0.261	
DCKM	**0.572**	0.263	0.123	**0.482**
OPC	0.534	0.167	0.329	0.431

Results on MiniNewsGroups Data. The miniNewsGroups data set is a reduced data set from the NewsGroups data [2]. It contains news about politics, computer, recreation and science. The original data set is comprised of 20000 documents from 20 newsgroups. We removed newsgroups with fewer than 20 documents, leaving us with 13 newsgroup classes. They are: comp.graphics, comp.sys.ibm.pc.hardware, comp.sys.mac.hardware, comp.windows.x, rec.autos, rec.sport.baseball, sci.crypt, sci.electronics, sci.med, sci.space, soc.religion.christian, talk.politics.guns and talk.politics.mideast. To reduce the running time, we subsampled 100 documents for each newsgroup.

After stemming, removing low frequent words, (appears less than 13 times), removing highly frequent words, (appears more than 700 times) and randomly sampling 800 words out of the remaining words, we have a data set of 800 words in 1300 documents. We included this data set to study even though only one labeling is provided, because it has a rich structure. It has a hierarchical structure and there are many ways to interpret the data into clusters. Furthermore, it is a text data (i.e., we can utilize the words to help us analyze whether or not the results make sense). In Table 4, we report the NMI results of our approach (MT) compared to other multiple clustering methods. Our approach found four views on this data. Among these views, view 1 corresponded well to the known newsgroup labeling and has the best NMI match compared to the other methods. The purity and subtree scores are provided in Table 8. The results show that view 1 discovered by MT matches the newsgroup classes well resulting in higher scores compared to the single coalescent tree which tries to discover the newsgroup class hierarchy using all the features. This shows that

our multiple hierarchical clustering approach is more appropriate for discovering clusters on such a multi-faceted data.

In addition to discovering multiple cluster hierarchies, MT can also discover the features for each view. The ten most frequent word selected by our method in view 1 are: *growth, week, side, industry, line, stop, credit, act, company, rule.* The top words selected for view 2 are: *list, control, sense, order, last, curve, design, property, form, market.* The top words selected for view 3 are: *base, important, protest, fact, right, agreement, frequent, necessary, quality, group* And, the top words selected for view 4 are: *interest, operation, education, video, theory, instrument, condition, response, slow, size* Interestingly, our multiple coalescent model was able to select relevant words: the words chosen for view 1 are related to industry; the words for view 2 are related to business; the words for view 3 are related to politics; and the words for view 4 are related to computers.

Score	View 1	View 2	View 3	Single
Purity	**0.278**	0.195	0.215	0.153
Subtree	0.254	0.196	**0.297**	0.231

Table 8: Purity and subtree scores for the MiniNewsGroups data. for the different views found by our multiple coalescent tree and the single coalescent tree.

	NMI with News Groups			
Method	View 1	View 2	View 3	View 4
MT	**0.374**	0.258	0.270	0.234
Single	0.264	NA	NA	NA
GP	0.124			
DCKM	0.369	0.175	NA	NA
OPC	0.276	0.306	0.164	0.137

Table 9: NMI result for the different methods on the MiniNewsGroups data.

Results on Gene Data. In biology, original assumption of last universal common ancestor [9] is no longer a proper one since the evidence of Horizontal gene transfer (HGT) has been observed. The consequence is that tree structure for phylogenetic evolution is not proper anymore. Phylogenetic networks may be used where a bifurcating tree is not suitable. And this paper contribute another approach for the even more complex phylogenetic structure that we use multiple phylogenetic tree instead of one to either discover a possible HGT or present alternative taxonomy.

In this experiment on gene data, we use a filtered data set from National Center for Biotechnology Information (NCBI) data repository. We use bacteria lineage due to the fact that HGT is common found in this case, although discovery of HGT in aukaryota would be great contribution but facing the fact that HGT in eukaryota is rare and unconfirmed. We further reduce to Actinomycetales family, in Bacteria, Actinobacteria, Actinobacteria(class), Actinobacteridae, Actinomycetales as from taxonomy of NCBI. Under this family, NCBI provides 346 species with full genome sequence. Then we use multiple sequence alignment(MSA) tool like MGA to find the match, alignment and gap. To form features. we first apply MSA in genus and collect matches and record this DNA sequence match as feature. By this method, we collect 6753 features. Then we the same MSA tool to find each genome against the selected feature, the value for that feature is set to 1 if the match presents otherwise 0. Then Gaussian observation model is used to build multiple coalescent trees. The result shows 7 coalescent structures. Most significant observation we can make is that 1. Corynebacterineae and Micrococcineae is mixed under one subtree in 3 views and well separated in the other 4 views. 2. Also note that some species under Streptomycineae is also under Corynebacterineae subtree in 2 views may be an proof for gram-positive behavior on DNA level. All these may be serving as evidence of gene sharing among different species and could further serve as show case for HGT on bacteria domain.

5 Conclusions

Many clustering algorithms output a single clustering solution. However, data may be multi-faceted (i.e., the single dataset may have several possible different clustering interpretations). In this paper, we introduced a probabilistic nonparametric Bayesian model for modeling multiple hierarchical clustering solutions. Our model allows us to automatically learn the number of views, the latent features and a hierarchical clustering tree for each view simultaneously. Our experiments on synthetic and real-world data show that a single hierarchical solution is not appropriate for richly structured multi-faceted data. Moreover, features appropriate for discovering the other clustering views can hurt the clustering solution of a single clustering model. Our results also show that our multiple coalescent model is able to learn the features and the clusters for each view that matches the alternative labeling information provided by humans.

References

[1] E. Bae and J. Bailey. Coala: A novel approach for the extraction of an alternate clustering of high quality and high dissimilarity. In *IEEE International Conference on Data Mining*, pages 53–62, 2006.

[2] S. D. Bay. The UCI KDD archive, 1999.

[3] D. M. Blei, T. L. Griffiths, and M. I. Jordan. The nested chinese restaurant process and bayesian inference of topic hierarchies, 2007.

[4] R. Caruana, M. Elhawary, N. Nguyen, and C. Smith. Meta clustering. In *IEEE International Conference on Data Mining*, pages 107–118, 2006.

[5] G. Chechik and N. Tishby. Extracting relevant structures with side information. In *Advances in Neural Information Processing Systems 15 (NIPS-2002)*, pages 857–864, 2003.

[6] CMU. CMU 4 universities WebKB data, 1997.

[7] Y. Cui, X. Z. Fern, and J. Dy. Non-redundant multi-view clustering via orthogonalization. In *IEEE Intl. Conf. on Data Mining*, pages 133–142, 2007.

[8] P. Diaconis and D. Freedman. Finite exchangeable sequences. *The Annals of Probability*, 8(4):745–764, 1980.

[9] W. F. Doolittle. Uprooting the tree of life. *Scientific American*, 72(7):90–95, 2000.

[10] J. G. Dy and C. E. Brodley. Feature selection for unsupervised learning. *Journal of Machine Learning Research*, 5:845–889, August 2004.

[11] A. Frank and A. Asuncion. UCI machine learning repository, 2010.

[12] D. Gondek and T. Hofmann. Non-redundant data clustering. In *Proceedings of the IEEE International Conference on Data Mining*, pages 75–82, 2004.

[13] Y. Guan, J. G. Dy, D. Niu, and Z. Ghahramani. Variational inference for nonparametric multiple clustering. In *Discovering, Summarizing, and Using Multiple Clusterings Workshop at KDD*, 2010.

[14] A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering: A review. *ACM Computing Surveys*, 31(3):264–323, 1999.

[15] P. Jain, R. Meka, and I. S. Dhillon. Simultaneous unsupervised learing of disparate clustering. In *SIAM Intl. Conf. on Data Mining*, pages 858–869, 2008.

[16] I. T. Jolliffe. *Principal Component Analysis*. Springer-Verlag, New-York, 1986.

[17] J. F. C. Kingman. The coalescent. *Stochastic Processes and their Applications*, 13(3):235 – 248, 1982.

[18] M. H. Law, M. Figueiredo, and A. K. Jain. Feature selection in mixture-based clustering. In *Advances in Neural Information Processing Systems 15*, Vancouver, December 2002.

[19] V. Mansinghka, E. Jonas, C. Petschulat, B. Cronin, P. Shafto, and J. Tenenbaum. Cross-categorization: A method for discovering multiple overlapping clusterings. In *Nonparametric Bayes Workshop at NIPS*, 2009.

[20] R. Neal. Probabilistic inference using markov chain monte carlo methods. Technical report, Dept. of Computer Science, University of Toronto, 1993.

[21] J. Pitman. Combinatorial stochastic processes. Technical report, U.C. Berkeley, Department of Statistics, August 2002.

[22] Z. J. Qi and I. Davidson. A principled and flexible framework for finding alternative clusterings. In *ACM SIGKDD Intl. Conf. on Knowledge Discovery and Data Mining*, 2009.

[23] A. Strehl and J. Ghosh. Cluster ensembles – a knowledge reuse framework for combining multiple partitions. *Journal on Machine Learning Research*, 3:583–617, 2002.

[24] Y. W. Teh, H. Daumé III, and D. Roy. Bayesian agglomerative clustering with coalescents. In *Proceedings of the Conference on Neural Information Processing Systems (NIPS)*, Vancouver, Canada, 2007.

[25] Y. W. Teh and M. I. Jordan. Hierarchical bayesian nonparametric models with applications. Technical report, University College London, 2008.

[26] M. Turk and A. Pentland. Face recognition using eigenfaces. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 586–591, 1991.