The Sound of an Album Cover: Probabilistic Multimedia and IR

Eric Brochu
Department of Computer Science
University of British Columbia
Vancouver, BC, Canada
ebrochu@cs.ubc.ca

Nando de Freitas
Department of Computer Science
University of British Columbia
Vancouver, BC, Canada
nando@cs.ubc.ca

Kejie Bao
Department of Computer Science
University of British Columbia
Vancouver, BC, Canada
kbao@cs.ubc.ca

Abstract

We present a novel, flexible, statistical approach to modeling music, images and text jointly. The technique is based on multi-modal mixture models and efficient computation using online EM. The learned models can be used to browse multimedia databases, to query on a multimedia database using any combination of music, images and text (lyrics and other contextual information), to annotate documents with music and images, and to find documents in a database similar to input text, music and/or graphics files.

1 INTRODUCTION

An essential part of human psychology is the ability to identify music, text, images or other information based on associations provided by contextual information of different media. Think no further than a well-chosen image on a book cover, which can instantly establish for the reader the contents of the book, or how the lyrics to a familiar song can instantly bring the song's melody to mind. In this paper, we describe an attempt to reproduce this effect by presenting a method for querying on a multimedia database with words, music and/or images.

Musically, we focus on monophonic and polyphonic musical pieces of known structure (MIDI files, full music notation, etc.). Retrieving these pieces in multimedia databases, such as the Web, is a problem of growing interest (Hoos, Renz and Gorg 2001, Huron and Aarden 2002, Pickens 2000). A significant step was taken by Downie (Downie 1999), who initially applied standard text IR techniques to retrieve music by converting music to text format. Most research (including (Downie 1999)) has, however, focused on plain music retrieval. To the best of our knowledge there has been no attempt to model text and music jointly.

To this we also add the capability of modeling images. While many different models of image features could be incorporated, we use a simple multinomial model based on image histograms. Our work is motivated by similar approaches to image and text modeling (Barnard and Forsyth 2001, Barnard, Duygulu and Forsyth 2001, Duygulu, Barnard, de Freitas and Forsyth 2002), and our earlier research in modeling music and text (Brochu and de Freitas 2003). The inclusion of music adds a new interesting dimension.

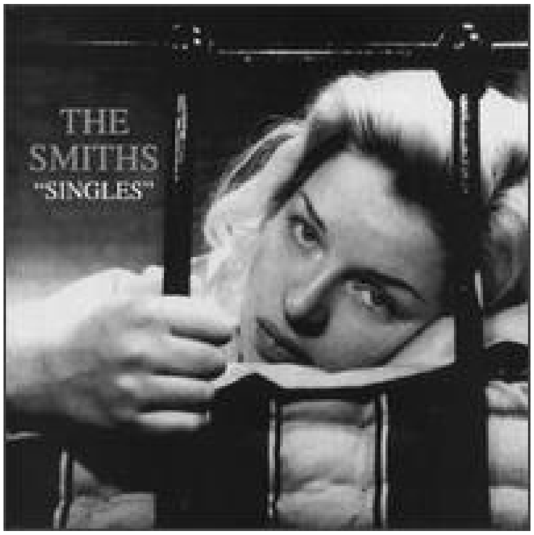


Figure 1: *The CD cover art for "Singles" by The Smiths. Using this image as input, our querying method returns the songs "How Soon is Now?" and "Bigmouth Strikes Again" – also by The Smiths – by probabilistically clustering the query image, finding database images with similar histograms, and returning songs associated with those images. In this case, the high-contrast black-and-white cover of "Singles" matches the similarly stark cover art of the CD associated with other Smiths songs (see figure 9).*

We propose a joint probabilistic model for documents with any combination of music, images and/or text. This model is simple, easily extensible, flexible and powerful. It allows users to query multimedia databases using text, images and/or music as input. It is well suited for browsing applications as it organizes the documents into "soft" clusters. The document of highest probability in each cluster can serve as a music or graphical thumbnail for automated summarization. The model allows one to query with text, music, and/or image documents to automatically annotate the document with appropriate musical and/or images, or to find similar documents (figure 1). It can be used to automatically recommend or identify similar songs and pictures. It allows for the inclusion of different types of text, including website content, lyrics, and meta-data such as hyper-text links. And finally, it is flexible enough to easily handle new media such as video, and to incorporate different models of extant media.

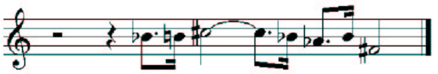
We use an online EM algorithm to train the models as the data arrives sequentially. This enables us to process vast quantities of data efficiently and faster than traditional batch EM.

2 MODEL SPECIFICATION

Our current model is based on documents with text (lyrics or information about the song), musical scores in GUIDO notation[1] (Hoos et al. 2001), and JPEG image files. We model the data with a Bayesian multi-modal mixture model. Words, images and scores are assumed to be conditionally independent given the mixture component label.

We model musical scores with first-order Markov chains, in which each state corresponds to a note, rest, or the start of a new voice. Notes' pitches are represented by the interval change (in semitones) from the previous note, rather than by absolute pitch, so that a score or query transposed to a different key will still have the same Markov chain. Rhythm is represented using the standard fractional musical measurement of whole-note, half-note, quarter-note, etc. Rest states are represented similarly, save that pitch is not represented. See Figure 2 for an example.

Polyphonic scores are represented by chaining the beginning of a new voice to the end of a previous one. In order to ensure that the first note in each voice appears in both the row and column of the Markov transition matrix, a special "new voice" state with no interval or rhythm serves as a dummy state marking the beginning of a new voice. The first note of a voice has a distinguishing "first note" interval value.



[_*3/4 b&1*3/16 b1/16 c#2*11/16 b&1/16
a&1*3/16 b&1/16 f#1/2]

S	INTERVAL	DURATION
0	newvoice	0
1	rest	3/4
2	firstnote	3/16
3	+1	1/16
4	+2	11/16
5	-2	1/16
6	-2	3/16
7	+3	1/16
8	-5	1/2

Figure 2: *Sample melody – the opening notes to "The Yellow Submarine" by The Beatles – in different notations. From top: standard musical notation (generated from GUIDO notation), GUIDO notation, and as a series of states in a first-order Markov chain (also generated from GUIDO notation).*

The Markov chain representation of a piece of music k is then mapped to a transition frequency table M_k, where $M_{i,j,k}$ denotes the number of times we observe the transition from state i to state j in document k. We use $M_{k,0}$ to denote the initial state of the Markov chain. In essence, this Markovian approach is analogous to a text bigram model, save that the states are musical notes and rests rather than words.

Images are represented as image intensity histograms of 256 equally-spaced bins, each representing a range of colour values. Each bin's value is initially equal to the number of pixels of the image that fall into that range, and then the entire histogram is normalized to find the relative frequencies of the bins, G_k, where $G_{k,b}$ indicated the relative frequency value of bin b in image k.

The associated text is modeled using a standard term frequency vector T_k, where $T_{w,k}$ denotes the number of times word w appears in document k.

For notational simplicity, we group together the image, music and text variable as follows: $X_k \triangleq \{G_k, M_k, T_k\}$. Note that there is no requirement of uniqueness in the database for the elements of different X_k. One graphic document can be associated with any number of text or music documents, though under this model, each association requires a separate instance of X. We rely on a human expert to provide the groupings for each instance of X in the database.

Our multi-modal mixture model is as follows:

$$X_k|\theta \overset{iid}{\sim} \sum_{c=1}^{n_c} p(c) \left[\prod_{b=1}^{n_b} p(b|c)^{G_{b,k}} \prod_{j=1}^{n_s} p(j|c)^{\mathbb{I}_j(M_{k,0})} \cdot \prod_{j=1}^{n_s} \prod_{i=1}^{n_s} p(j|i,c)^{M_{i,j,k}} \prod_{w=1}^{n_w} p(w|c)^{T_{w,k}} \right] \qquad (1)$$

where $\theta \triangleq \{p(c), p(b|c), p(j|c), p(j|i,c), p(w|c)\}$ encom-

[1] GUIDO is a powerful language for representing musical scores in an HTML-like notation. MIDI files, plentiful on the World Wide Web, can be easily converted to this format.

passes all the model parameters and where $\mathbb{I}_j(M_{k,0}) = 1$ if the first entry of M_k belongs to state j and is 0 otherwise. The three-dimensional matrix $p(j|i,c)$ denotes the estimated probability of transitioning from state i to state j in cluster c, the matrix $p(j|c)$ denotes the initial probabilities of being in state j, given membership in cluster c. The vector $p(c)$ denotes the probability of each cluster. The two-dimensional matrices $p(b|c)$ and $p(w|c)$ denote the probabilities of bin b and word w in cluster c.

The mixture model is defined on the standard probability simplex $\{\forall c, p(c) \geq 0 \text{ and } \sum_{c=1}^{n_c} p(c) = 1\}$. We introduce the latent allocation variables $z_k \in \{1, \ldots, n_c\}$ to indicate that a particular sequence $\mathbf{x}_k$ belongs to a specific cluster c. These indicator variables $\{z_k; k = 1, \ldots, n_x\}$ correspond to an i.i.d. sample from the distribution $p(z_k = c) = p(c)$.

This simple model is easy to extend. For browsing applications, we might prefer a hierarchical structure with levels l:

$$X_k|\theta \overset{iid}{\sim} \sum_{c=1}^{n_c} p(c) \sum_{l=1}^{n_l} p(l|c)p(G_k|c,l)p(M_k|c,l)p(T_k|c,l) \tag{2}$$

This is still a multinomial model, but by applying appropriate parameter constraints we can produce a tree-like browsing structure (Barnard and Forsyth 2001). It is also easy to formulate the model in terms of aspects and clusters as suggested in (Hofmann 1999, Blei, Ng and Jordan 2002).

2.1 PRIOR SPECIFICATION

We follow a hierarchical Bayesian strategy, where the unknown parameters θ and the allocation variables z are regarded as being drawn from appropriate prior distributions. We acknowledge our uncertainty about the exact form of the prior by specifying it in terms of some unknown parameters (hyperparameters). The allocation variables z_k are assumed to be drawn from a multinomial distribution, $z_k \sim \mathcal{M}_{n_c}(1; p(c))$. We place a conjugate Dirichlet prior on the mixing coefficients $p(c) \sim \mathcal{D}_{n_c}(\alpha)$. Similarly, we place Dirichlet prior distributions $\mathcal{D}_{n_j}(\beta)$ on each $p(j|c)$, $\mathcal{D}_{n_j}(\gamma)$ on each $p(j|i,c)$, $\mathcal{D}_{n_w}(\rho)$ on each $p(w|c)$, $\mathcal{D}_{n_b}(\nu)$ on each $p(b|c)$, and assume that these priors are independent.

The posterior for the allocation variables will be required. It can be obtained easily using Bayes' rule:

$$\begin{aligned} \xi_{ck} \triangleq & p(z_k = c|\theta, X_k) = \frac{p(X_k|c,\theta)p(c|\theta)}{p(X_k|\theta)} \\ \propto & p(c) \left(\prod_{b=1}^{n_b} p(b|c)^{G_{b,k}} \prod_{j=1}^{n_s} p(j|c)^{\mathbb{I}_j(M_{k,0})} \right. \\ & \left. \prod_{j=1}^{n_s} \prod_{i=1}^{n_s} p(j|i,c)^{M_{i,j,k}} \prod_{w=1}^{n_w} p(w|c)^{T_{w,k}} \right) \end{aligned} \tag{3}$$

3 COMPUTATION

The parameters of the mixture model cannot be computed analytically unless one knows the mixture indicator variables. We have to resort to numerical methods. One can implement a Gibbs sampler to compute the parameters and allocation variables. This is done by sampling the parameters from their Dirichlet posteriors and the allocation variables from their multinomial posterior. However, this algorithm is too computationally intensive for the applications we have in mind. Instead we opt for expectation maximization (EM) algorithms. We derive batch EM algorithms for maximum likelihood (ML) and *maximum a posteriori* (MAP) estimation. We also derive a very efficient on-line EM algorithm, which can be interpreted as a quasi-Bayes procedure (Smith and Makov 1978).

3.1 ML ESTIMATION WITH EM

After initialization, the EM algorithm for ML estimation iterates between the following two steps:

1. E step: Compute the expectation of the complete log-likelihood with respect to the distribution of the allocation variables $\mathbf{Q}^{\text{ML}} = \mathbb{E}_{p(z|G,M,T,\theta^{(\text{old})})}[\log p(z,G,M,T|\theta)]$, where $\theta^{(\text{old})}$ represents the value of the parameters at the previous time step.

2. M step: Maximize over the parameters: $\theta^{(\text{new})} = \arg\max_{\theta} \mathbf{Q}^{\text{ML}}$

The $\mathbf{Q}^{\text{ML}}$ function expands to

$$\begin{aligned} \mathbf{Q}^{\text{ML}} = & \sum_{k=1}^{n_x} \sum_{c=1}^{n_c} \xi_{ck} \log \left[p(c) \prod_{b=1}^{n_b} p(b|c)^{G_{b,k}} \right. \\ & \left. \prod_{j=1}^{n_s} p(j|c)^{\mathbb{I}_j(M_{k,0})} \prod_{j=1}^{n_s} \prod_{i=1}^{n_s} p(j|i,c)^{M_{i,j,k}} \prod_{w=1}^{n_w} p(w|c)^{T_{w,k}} \right]. \end{aligned}$$

In the E step, we have to compute ξ_{ck} using equation (3). The corresponding M step requires that we maximize $\mathbf{Q}^{\text{ML}}$ subject to the constraints that all probabilities for the parameters sum up to 1. This constrained maximization can be carried out by introducing Lagrange multipliers. The resulting parameter estimates are:

$$\begin{aligned} \widehat{p}(c) &= \frac{1}{n_x} \sum_{k=1}^{n_x} \xi_{ck} \\ \widehat{p}(b|c) &= \frac{\sum_{k=1}^{n_x} G_{b,k} \xi_{ck}}{\sum_{k=1}^{n_x} \sum_{b'} G_{b',k} \xi_{ck}} \\ \widehat{p}(j|c) &= \frac{\sum_{k=1}^{n_x} \mathbb{I}_j(M_{k,0}) \xi_{ck}}{\sum_{k=1}^{n_x} \xi_{ck}} \end{aligned}$$

$$\begin{aligned}\widehat{p}(j|i,c) &= \frac{\sum_{k=1}^{n_x} M_{i,j,k}\xi_{ck}}{\sum_{k=1}^{n_x}\sum_{j'=1}^{n_s} M_{i,j',k}\xi_{ck}}\\ \widehat{p}(w|c) &= \frac{\sum_{k=1}^{n_x} T_{w,k}\xi_{ck}}{\sum_{k=1}^{n_x}\sum_{w'} T_{w',k}\xi_{ck}}\end{aligned}$$

3.2 MAP ESTIMATION WITH EM

The EM formulation for MAP estimation is straightforward. One simply has to augment the objective function in the M step, $\mathbf{Q}^{\text{ML}}$, by adding to it the log prior densities. That is, the MAP objective function is

$$\mathbf{Q}^{\text{MAP}} = \mathbb{E}_{p(z|X,\theta^{(\text{old})})}\left[\log p(z,X,\theta)\right] = \mathbf{Q}^{\text{ML}} + \log p(\theta)$$

The MAP parameter estimates are:

$$\begin{aligned}\widehat{p}(c) &= \frac{\alpha_c - 1 + \sum_{k=1}^{n_x}\xi_{ck}}{\sum_{c'=1}^{n_c}\alpha_{c'} - n_c + n_x}\\ \widehat{p}(b|c) &= \frac{\nu_{b,c} - 1 + \sum_{k=1}^{n_x} G_{b,k}\xi_{ck}}{\sum_{b'=1}^{n_b}\nu_{b',c} - n_b + \sum_{k=1}^{n_x}\sum_{b'} G_{b',k}\xi_{ck}}\\ \widehat{p}(j|c) &= \frac{\beta_{j,c} - 1 + \sum_{k=1}^{n_x}\mathbb{I}_j(M_{k,0})\xi_{ck}}{\sum_{j'=1}^{n_s}\beta_{j',c} - n_s + \sum_{k=1}^{n_x}\xi_{ck}}\\ \widehat{p}(j|i,c) &= \frac{\gamma_{i,j,c} - 1 + \sum_{k=1}^{n_x} M_{i,j,k}\xi_{ck}}{\sum_{j'=1}^{n_s}\gamma_{i,j',c} - n_s + \sum_{j'=1}^{n_s}\sum_{k=1}^{n_x} M_{i,j',k}\xi_{ck}}\\ \widehat{p}(w|c) &= \frac{\rho_{w,c} - 1 + \sum_{k=1}^{n_x} T_{w,k}\xi_{ck}}{\sum_{w'=1}^{n_w}\rho_{w',c} - n_w + \sum_{k=1}^{n_x}\sum_{w'} T_{w',k}\xi_{ck}}\end{aligned}$$

These expressions can also be derived by considering the posterior modes and by replacing the cluster indicator variable with its posterior estimate ξ_{ck}. This observation opens up room for various stochastic and deterministic ways of improving EM.

3.3 ON-LINE EM ALGORITHM

The batch EM algorithms fail to scale well as the number of database entries becomes very large. To surmount this problem, we derive an on-line EM algorithm. In this setting, the training data are supplied one by one and the model parameters are updated within each time frame using the current data. A simpler version of this algorithm was originally proposed as a quasi-Bayes procedure in (Smith and Makov 1978). There, it was shown that the algorithm can be interpreted as a stochastic approximation procedure and, hence, it is possible to prove convergence in stationary regimes like ours (Deylon, Lavielle and Moulines 1999). This algorithm has been re-invented a few times in the machine learning literature (Sato 1999, Sato and Ishii 1998).

Let θ_t denote the parameter after the t^{th} observation X_t. We define $\langle f(X)\rangle_T$ as the weighted mean of $f(X)$ at time T with respect to the posterior probability of the cluster allocation variables. Our goal is to derive online updates for the sufficient statistics required to compute the model parameters. The updates will be of the form

$$\begin{aligned}\langle f(X)\rangle_t &= \langle f(X)\rangle_{t-1}\\ &+ \eta_t\left(f(X_t)\xi_{ct} - \langle f(X)\rangle_{t-1}\right)\end{aligned}$$

where η_t denotes the learning rate and $\xi_{ct} \triangleq p(z_t = c|X_t,\theta_{t-1})$. As in the batch scenario, the E step involves computing the posterior of the allocation variables:

$$\begin{aligned}\xi_{ct} \propto p(c)_t \Bigg(&\prod_{b=1}^{n_b} p(b|c)_t^{G_{b,t}}\prod_{j=1}^{n_s} p(j|c)_t^{\mathbb{I}_j(M_0)_t}\\ &\prod_{j=1}^{n_s}\prod_{i=1}^{n_s} p(j|i,c)_t^{M_{i,j,t}}\prod_{w=1}^{n_w} p(w|c)_t^{T_{w,t}}\Bigg)\end{aligned}$$

In the M step, we compute the parameters

$$\begin{aligned}\widehat{p}(c)_t &= \langle 1\rangle_{c,t}\\ \widehat{p}(b|c)_t &= \frac{\langle G_b\rangle_{c,t}}{\langle\sum_b G_b\rangle_{c,t}}\\ \widehat{p}(j|c)_t &= \frac{\langle\mathbb{I}_j(M_0)\rangle_{c,t}}{\langle 1\rangle_{c,t}}\\ \widehat{p}(j|i,c)_t &= \frac{\langle M_{i,j}\rangle_{c,t}}{\langle\sum_j M_{i,j}\rangle_{c,t}}\\ \widehat{p}(w|c)_t &= \frac{\langle T_w\rangle_{c,t}}{\langle\sum_w T_w\rangle_{c,t}}\end{aligned}$$

where the online expectations are given by

$$\begin{aligned}\langle 1\rangle_{c,t} &= \langle 1\rangle_{c,t-1} + \eta_{1t}\left[\xi_{ct} - \langle 1\rangle_{c,t-1}\right]\\ \langle G_b\rangle_{c,t} &= \langle G_b\rangle_{c,t-1} + \eta_{2t}\left[G_{b,t}\xi_{ct} - \langle G_b\rangle_{c,t-1}\right]\\ \langle\sum_b G_b\rangle_{c,t} &= \langle\sum_b G_b\rangle_{c,t-1}\\ &+\eta_{3t}\left[\sum_b G_{b,t}\xi_{ct} - \langle\sum_b G_b\rangle_{c,t-1}\right]\\ \langle T_w\rangle_{c,t} &= \langle T_w\rangle_{c,t-1} + \eta_{4t}\left[T_{w,t}\xi_{ct} - \langle T_w\rangle_{c,t-1}\right]\\ \langle\sum_w T_w\rangle_{c,t} &= \langle\sum_w T_w\rangle_{c,t-1}\\ &+\eta_{5t}\left[\sum_w T_{w,t}\xi_{ct} - \langle\sum_w T_w\rangle_{c,t-1}\right]\\ \langle\mathbb{I}_j(M_0)\rangle_{c,t} &= \langle\mathbb{I}_j(M_0)\rangle_{c,t-1}\\ &+\eta_{6t}\left[\mathbb{I}_j(M_0)_t\xi_{ct} - \langle\mathbb{I}_j(M_0)\rangle_{c,t-1}\right]\\ \langle M_{i,j}\rangle_{c,t} &= \langle M_{i,j}\rangle_{c,t-1}\\ &+\eta_{7t}\left[M_{i,j,t}\xi_{ct} - \langle M_{i,j}\rangle_{c,t-1}\right]\\ \langle\sum_j M_{i,j}\rangle_{c,t} &= \langle\sum_j M_{i,j}\rangle_{c,t-1}\\ &+ \eta_{8t}\left[\sum_j M_{i,j,t}\xi_{ct} - \langle\sum_j M_{i,j}\rangle_{c,t-1}\right]\end{aligned} \tag{4}$$

To ensure convergence, one should choose decaying learning rates $\eta_{it} = (at+b_i)^{-1}$. As shown in (Smith and Makov 1978), for the upate of $\widehat{p}(c)$, a and b are functions of the Dirichlet hyperparameters. This suggests that one could use the priors to specify the learning rates: we are currently investigating this avenue.

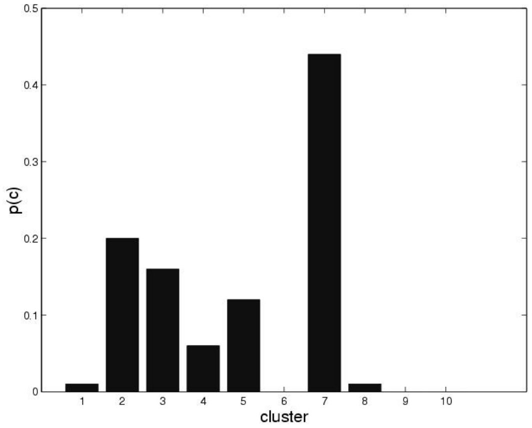


Figure 3: *Probability mass distribution across clusters discovered from our test dataset using online EM.*

4 EXPERIMENTS

To test the model with text, images and music, we clustered on a database of musical scores with associated text documents and JPEG images. The database is composed of various types of musical scores – jazz, classical, television theme songs, and contemporary pop and electronic music – each of which has an associated text file and image file, as represented by the combined media variable X_k. The scores are represented in GUIDO notation. The associated text files are a song's lyrics, where applicable, or textual commentary on the score for instrumental pieces, all of which were extracted from the World Wide Web. The image file for each piece is an image of the cover of the CD on which the song appears.

The experimental database contains 100 scores, each with a single associated text document and image. There is nothing in the model, however, that requires this one-to-one association of text documents and scores – this was done solely for testing and pedagogical simplicity. In a deployment such as the world wide web, one would routinely expect one-to-many or many-to-many mappings between the scores and text.

We tested our database with the various versions of EM described above. We found that standard batch ML EM gave the least satisfactory results, distributing probability mass across the maximum number of clusters in a nondeterministic fashion. Batch MAP and online EM give better results, and regularize to a smaller number of more intuitive clusters. There is little difference in the clusters found by batch MAP and online EM. In our experiments, we found online EM to run in significantly less time than batch EM, and online EM is the clustering method used in the following sections. Figures 3 and 4 show some representative cluster probability assignments obtained with online EM estimation.

CLUSTER	SONG	ξ_{ck}
1	*The Beatles – Good Day Sunshine*	0.1667
1	*other – 'The Addams Family' theme*	0.0043
2	*J. S. Bach – Invention #1*	1.0000
2	*J. S. Bach – Invention #2*	1.0000
2	*other – 'The Jetsons' Theme*	1.0000
⋮	⋮	⋮
3	*Nine Inch Nails – Down In It*	1.0000
3	*Nine Inch Nails – The Perfect Drug*	0.9998
3	*Nine Inch Nails – Wish*	1.0000
⋮	⋮	⋮
4	*The Cure – 10:15 Saturday Night*	1.0000
4	*Moby – Flower*	0.6667
4	*other – 'The Addams Family' theme*	0.9957
⋮	⋮	⋮
5	*The Smiths – Girlfriend in a Coma*	1.0000
5	*The Cure – Push*	0.9753
5	*Nine Inch Nails – The Perfect Drug*	0.0002
⋮	⋮	⋮
7	*The Prodigy – One Love*	1.0000
7	*PJ Harvey – Down by the Water*	1.0000
7	*Rogers & Hart – Blue Moon*	1.0000
⋮	⋮	⋮
8	*Soft Cell – Tainted Love*	1.0000

Figure 4: *Representative probabilistic cluster allocations using ML estimation via online EM.*

By and large, the clusters are intuitive. The 15 pieces by J. S. Bach each have very high ($p > 0.999$) probabilities of membership in the same cluster, as do the 13 pieces from the band Nine Inch Nails. A few curious anomalies exist. The theme song to the television show *The Jetsons* is included in the same cluster as the Bach pieces, for example.

4.1 DEMONSTRATING THE UTILITY OF MULTI-MODAL QUERIES

A major intended use of the text-score-image model is for searching documents on a combination of text, images and/or music. Consider a hypothetical example, using text and music only: A music fan is struggling to recall a dimly-remembered song with a strong repeating single-pitch, dotted-eight-note/sixteenth-note bass line, and lyrics containing the words "*come on, get down.*" We can use our database non-probabilistically to find all instances of X_k for which T_k contains all the words at least once and M_k contains each of the desired transitions at least once.

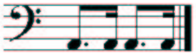
QUERY	RETRIEVED SONGS
“come on, get down”	*Erksine Hawkins – Tuxedo Junction* *Moby – Bodyrock* *Nine Inch Nails – Last* *Sherwood Schwartz – ‘The Brady Bunch’ theme song*
	*The Beatles – Got to Get You Into My Life* *The Beatles – I’m Only Sleeping* *The Beatles – Yellow Submarine* *Moby – Bodyrock* *Moby – Porcelain* *Gary Portnoy – ‘Cheers’ theme song* *Rodgers & Hart – Blue Moon*
 “come on, get down”	*Moby – Bodyrock*

Figure 5: *Examples of query matches, using only text, only musical notes, and both text and music. The combined query is more precise.*

A search on the text portion alone turns up four documents which have matching lyrics. A search on the notes alone returns seven documents which have matching transitions. But a combined search returns only the correct document (figure 5).

4.2 PROBABILISTIC QUERYING

While retrieving documents on a deterministic, ‘all-or-nothing’ basis can be useful for small datasets, or on very precise queries, it is often desirable for a query to return not a complete subset of matching responses, but an arbitrary number of ranked responses. This also allows us to query using graphic files as input.

To perform a probabilistic multimodal query, we simply sample probabilistically without replacement from the clusters. A query $\mathbf{Q}$ is composed of one or more media – a text string $\mathbf{Q}_T$, a series of musical transitions $\mathbf{Q}_M$, and/or an image $\mathbf{Q}_G$. The probability of sampling from each cluster, $p(c|\mathbf{Q})$, is computed using equation 3, and assigning a value of 1 to any multinomial probability modeling a non-occurring medium. Sampling probabilistically from clusters allows us to search the database without checking every entry.

In each iteration i, a cluster c is selected by randomly sampling from $p(c|\mathbf{Q})$, and the matching criteria are applied against each instance of X_k for which $p(X_k|c) > \epsilon$ in which ϵ is some small threshold to ensure that we do not inefficiently examine documents with negligible degrees of membership in the cluster. The matching criteria consists of using each instance of X as a generative multinomial model, and calculating $p(\mathbf{Q}|X_k)$.

$$\begin{aligned} p(\mathbf{Q}|X_k) = & \prod_{b=1}^{n_b}\left(\frac{G_{b,k}+\beta_G-1}{\sum_{b'=1}^{n_b}[G_{b',k}+\beta_G-1]}\right)^{\mathbf{Q}_{G,b}} \\ & \prod_{j=1}^{n_s}\left(\frac{M_{j,k}+\beta_M-1}{\sum_{j'=1}^{n_s}[M_{j',k}+\beta_M-1]}\right)^{\mathbb{I}_j(\mathbf{Q}_{M,0})} \\ & \prod_{j=1}^{n_s}\prod_{i=1}^{n_s}\left(\frac{M_{i,j,k}+\beta_M-1}{\sum_{j'=1}^{n_s}[M_{j',k}+\beta_M-1]}\right)^{\mathbf{Q}_{M,i,j}} \\ & \prod_{w=1}^{n_w}\left(\frac{T_{w,k}+\beta_T-1}{\sum_{w'=1}^{n_w}[T_{w',k}+\beta_T-1]}\right)^{\mathbf{Q}_{T,w}} \qquad (5) \end{aligned}$$

β_G, β_M, and β_T are independent conjugate Dirichlet priors, typically very close to one. The selected X_k is then $\arg\max_k p(\mathbf{Q}|X_k)$. Once selected, a given X_k cannot be reselected. If this results in a cluster no longer having any values of X_k such that $p(X_k|c) > \epsilon$, the cluster is assigned a probability of zero and the remaining cluster probabilities are renormalized.

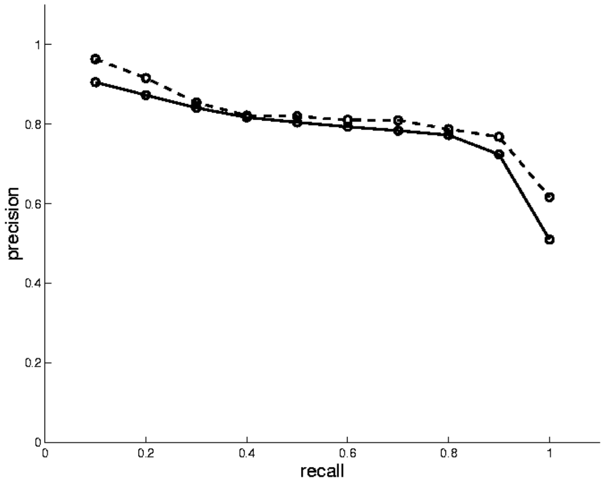


Figure 6: *Precision-recall curve showing average results, over 1000 randomly-generated queries, combining music and text matching criteria (solid line), and music, text and image criteria (dashed).*

4.3 PRECISION AND RECALL

We evaluated our retrieval system with randomly generated queries. In our first evaluation, we defined a query $\mathbf{Q}$ as

INPUT	**CLOSEST MATCH**
J. S. Bach – Toccata and Fugue in D Minor (score)	*J. S. Bach – Invention #5*
Nine Inch Nails – Closer (score & lyrics)	*Nine Inch Nails – I Do Not Want This*
T. S. Eliot – The Waste Land (text poem)	*The Cure – One Hundred Years*

Figure 7: *The results of associating songs in the database with other text and/or musical input. The input is clustered probabilistically and then associated with the existing song that has the least Euclidean distance in that cluster. The association of* The Waste Land *with The Cure's thematically similar* One Hundred Years *is likely due to the high co-occurrence of relatively uncommon words such as* water, death, *and* year(s).

composed of a random series of 1 to 5 note transitions, $\mathbf{Q}_M$ and 1 to 5 words, $\mathbf{Q}_T$. We then determine the actual number of matches n in the database, where a match is defined as a song X_k such that all elements of $\mathbf{Q}_M$ and $\mathbf{Q}_T$ have a frequency of 1 or greater. In order to avoid skewing the results by using unrealistically narrow or broad queries, we reject any query that has $n < 5$ or $n > 20$.

For the second experiment, we included a random image histogram representation, $\mathbf{Q}_G$. The nature of our model required us to arbitrarily determine a query generation method and matching criteria. To generate this histogram, we selected a random image from the set of X_k for which the corresponding T_k and M_k match the generated $\mathbf{Q}_T$ and $\mathbf{Q}_M$, and then sampled from a Gaussian in which the means were equal to the bins of the selected image and using the covariance matrix of all the images in the database. Two images are considered a match if the Euclidean distances between them is < 0.1.

We then sampled probabilistically from a set of clusters discovered with online EM, using equation 5 to find results. Based on previous experiments, we set the Dirichlet priors at $\beta_G = 1.1$, $\beta_T = 1.01$ and $\beta_M = 1.0001$. Once all the matches were returned, we computed the standard precision-recall curve (Baeza-Yates and Ribeiro-Neto 1999), as shown in figure 6. Our querying method enjoys a high precision until recall is approximately 80%, and experiences a relatively modest deterioration of precision thereafter.

4.4 ASSOCIATION

The probabilistic nature of our approach allows us the flexibility to use our techniques and database for tasks beyond traditional querying. One of the more promising avenues of exploration is associating documents with each other probabilistically. This could be used, for example, to find suitable songs for web sites or presentations (matching on text), or for recommending songs similar to one a user enjoys (matching on scores).

Given an entire input document as a query, $\mathbf{Q}$, we first cluster $\mathbf{Q}$ by finding the most likely cluster as determined by computing $\arg\max_c p(c|\mathbf{Q})$ (equation 3). Input documents containing text or music only can be clustered using only those components of the database. Input documents that combine text and music are clustered using all the data. Once the input document has been clustered, we can find its closest association by computing the distance from the input document to the other document vectors in the cluster. The distance can be defined in terms of matches, Euclidean measures, or cosine measures after carrying out latent semantic indexing (Deerwester, Dumais, Furnas, Landauer and Harshman 1990). A few selected examples of associations found in our database in this way are shown in figure 7. The results are often reasonable, though unexpected behavior occasionally occurs.

QUERY	**RETRIEVED SONGS**
$\{\mathbf{Q}_T\}$	*other – 'The Jetsons' theme song* *Nine Inch Nails – Burn* *R.E.M. – Man on the Moon*
$\{\mathbf{Q}_M\}$	*The Smiths – Girlfriend in a Coma* *Joy Division – Love Will Tear Us Apart* *The Beatles - I'm Only Sleeping*
$\{\mathbf{Q}_G\}$	*The Smiths – How Soon is Now?* *Moby – Alone* *The Smiths – Bigmouth Strikes Again*
$\{\mathbf{Q}_G, \mathbf{Q}_M, \mathbf{Q}_T\}$	*The Smiths – Girlfriend in a Coma* *The Smiths – Bigmouth Strikes Again* *The Smiths – How Soon is Now?*

Figure 8: *The results of the experiment described in the text. The song "The Boy With the Torn in His Side" by The Smiths was represented by text (lyrics), music (score), and image (album cover shown in figure 1), as a query* $\{\mathbf{Q}_G, \mathbf{Q}_M, \mathbf{Q}_T\}$. *Shown are the top three matches for each component separately, and for all three together. The best results are achieved when all three media are submitted in the query.*

As a demonstration of the power of this approach, we tested the retrieval algorithm using as input the song "The Boy With the Thorn in His Side" by The Smiths. The musical portion $\mathbf{Q}_M$ is extracted from a GUIDO file, which was generated from a MIDI file. The text portion $\mathbf{Q}_T$ is a based on the song's lyrics. The image histogram $\mathbf{Q}_G$ is derived from the cover of the album on which the song appears (figure 1). This song was intentionally chosen for this demonstration, as all the images associated with the songs in the

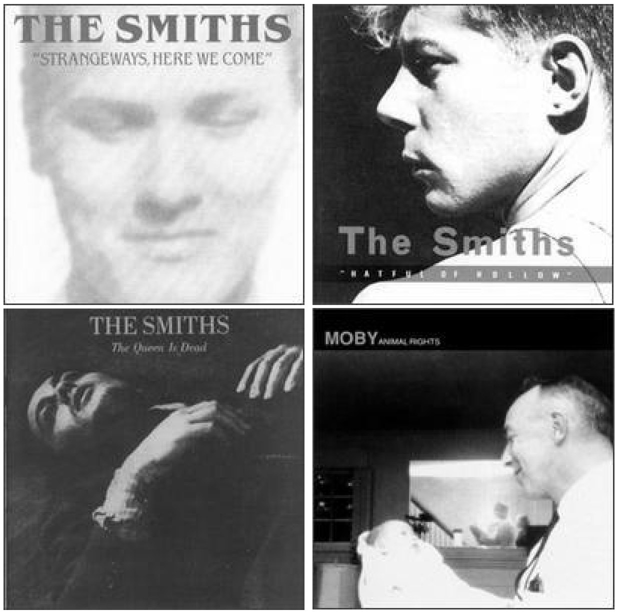


Figure 9: *CD cover art that matched the queries in figure 8. Clockwise from top left: "Girlfriend in a Coma," "How Soon is Now?," "Alone," "Bigmouth Strikes Again." The album by Moby is incorrectly returned when the query is based on the image in figure 1 alone. When music and text information is added, only the three albums by The Smiths are returned.*

database are also the CD cover art for the songs, and many albums from The Smiths feature high-contrast black-and-white cover photographs.

Using various elements of the set of variables $\{\mathbf{Q}_G, \mathbf{Q}_M, \mathbf{Q}_T\}$, we ran the query algorithm as presented in section 4.2. The top three ranked results of each trial are shown in figure 8. The various images that were matched in the trials are shown in figure 9.

5 CONCLUSIONS

We feel that the probabilistic approach to querying on music, text and images presented here is powerful, flexible, and novel, and suggests many interesting areas of future research. One immediate goal is to test this approach on larger databases. In the future, we should be able to incorporate audio by extracting suitable features from the signals. This will permit querying by singing, humming, or via recorded music. Segmentation and feature extraction can be used to model images in a more sophisticated manner, and other media, such as video.

Acknowledgments

We would like to thank Kobus Barnard, J. Stephen Downie, Holger Hoos and Peter Carbonetto for their advice and expertise in preparing this paper.

References

Baeza-Yates, R. and Ribeiro-Neto, B. (1999). *Modern Information Retrieval*, Addison-Wesley.

Barnard, K. and Forsyth, D. (2001). Learning the semantics of words and pictures, *International Conference on Computer Vision*, Vol. 2, pp. 408– 415.

Barnard, K., Duygulu, P. and Forsyth, D. (2001). Clustering art, *Computer Vision and Pattern Recognition*, Vol. 2, pp. 434–439.

Blei, D. M., Ng, A. Y. and Jordan, M. I. (2002). Latent dirichlet allocation, *in* T. G. Dietterich, S. Becker and Z. Ghahramani (eds), *Advances in Neural Information Processing Systems 14*, MIT Press, Cambridge, MA.

Brochu, E. and de Freitas, N. (2003). "Name that Song!": A probabilistic approach to querying on music and text, *Advances in Neural Information Processing Systems 15*, MIT Press, Cambridge, MA. To appear.

Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K. and Harshman, R. (1990). Indexing by latent semantic indexing, *Journal of the American Society for Information Science* **41**(6): 391– 407.

Deylon, B., Lavielle, M. and Moulines, E. (1999). Convergence of a stochastic approximation version of the EM algorithm, *The Annals of Statistics* **27**(1): 94–128.

Downie, J. S. (1999). *Evaluating a Simple Approach to Music Information Retrieval: Conceiving Melodic N-Grams as Text*, PhD thesis, University of Western Ontario.

Duygulu, P., Barnard, K., de Freitas, N. and Forsyth, D. (2002). Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary, *ECCV*.

Hofmann, T. (1999). Probabilistic latent semantic analysis, *Uncertainty in Artificial Intelligence*.

Hoos, H. H., Renz, K. and Gorg, M. (2001). GUIDO/MIR - an experimental musical information retrieval system based on GUIDO music notation, *International Symposium on Music Information Retrieval*.

Huron, D. and Aarden, B. (2002). Cognitive issues and approaches in music information retrieval, *in* S. Downie and D. Byrd (eds), *Music Information Retrieval*.

Pickens, J. (2000). A comparison of language modeling and probabilistic text information retrieval approaches to monophonic music retrieval, *International Symposium on Music Information Retrieval*.

Sato, M. A. (1999). Fast learning of on-line EM algorithm, *Technical report*, TR-H-281, ATR Human Information Processing Research Laboratories.

Sato, M. A. and Ishii, S. (1998). On-line EM algorithm for the normalized Gaussian network, *Neural Computation* **12**(2): 407–432.

Smith, A. F. M. and Makov, U. E. (1978). A quasi-Bayes sequential procedure for mixtures, *Journal of the Royal Statistical Society, Series B* **40**(1): 106–112.